CHARACTERIZING AND LEVERAGING PEOPLE MOVEMENT FOR CONTENT DISTRIBUTION IN MOBILE PEER-TO-PEER NETWORKS

BY

LONG HAI VU

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Professor Klara Nahrstedt, Chair
Professor Roy Campbell
Professor Nitin Vaidya
Associate Professor Indranil Gupta
Professor Andrew Campbell, Dartmouth College

ABSTRACT

In this thesis, we present a framework to characterize and leverage people movement for improvements of content distribution in mobile Peer-to-Peer (P2P) networks. Particularly, we study two typical classes of people movement including the Schelling behavior and repetitive behavior. The Schelling behavior exists in real-world scenarios where co-located people collaboratively share mutual content interest when they are moving towards the same Point of Interest such as shopping mall, football stadium, and outdoor concert. Meanwhile, the repetitive behavior of people movement can be found in numerous places where people visit regular locations and make regular social contacts for their daily routines such as university campuses and work places.

For the first part of the thesis, we start by analyzing the original segregation model proposed by Thomas Schelling, a Nobel prize winner in economics. We find that the properties of the segregation model exist in numerous real-world scenarios, in which the co-located people may form groups and collaboratively share data messages using their wireless devices when they are moving towards the same Point of Interest. This grouping behavior of people (or their mobile devices) is called the "Schelling behavior" of people movement. We find that when mobile nodes exhibit Schelling behavior, the network formed by these nodes has two important properties: (1) co-located mobile nodes form "moving" coalitions, and (2) the coalition size increases at the closer distance from Point of Interest. We then conduct a validation study on these properties by: (1) simulating people movement on real Google maps, and (2) modeling people movement in different street configurations by using the Mobius modeling tool. Our validation study confirms the two properties of the Schelling behavior of people movement. Then, we exploit these properties to design three protocols to improve content distribution in mobile P2P networks, including COADA, iShare, and DENTA. We evaluate our protocols via simulation and the evaluation results show that our protocols outperform other existing content distribution schemes significantly by improving message delivery and reducing message overhead.

For the second part of the thesis, we exploit the repetitive behavior of people movement for the design of content distribution protocols. Particularly, we propose a new methodology to collect people movement trace using mobile phones. We then apply this method to implement a trace collection system named UIM, which collects MAC addresses of Wifi access points and Bluetooth-enabled devices in the proximity of the experiment phones. The UIM system is deployed on Google Android phones carried by 123 faculties, staff, and students in University of Illinois campus from March 2010 to August 2010. The collected MAC addresses of Wifi access points are used to infer location information and the collected Bluetooth MACs are used to infer social contact. To the best of our knowledge, the UIM system is the first system to collect both location information and social contact of people movement. The inferred location information and social contact then are used in the characterization study, which shows that people movement exhibits a high degree of repetition. We then propose a novel method named Jyotish [1] to construct a predictive model of people movement from the joint Wifi/Bluetooth trace to predict future information of location, stay duration at the location, and social contact. Applying the Jyotish method, we construct a predictive model from the joint Wifi/Bluetooth trace collected by the UIM scanning system. To the best of our knowledge, Jyotish is the first method to construct the predictive model of people movement from the joint Wifi/Bluetooth trace and our constructed predictive model is the first to provide altogether predictions for location, stay duration, and social contact. Finally, we leverage the constructed predictive model to design a new content distribution protocol named COMFA, which exploits the regularity of social contact found in the Bluetooth trace collected by the UIM system to maximize the message delivery probability and preserve message delivery deadline. We compare the performance of COMFA with Prophet routing and Epidemic routing over the collected Bluetooth trace and the evaluation results show that COMFA outperforms other alternatives by reducing the message delivery delay and message overhead considerably.

[1] In Sanskrit, Jyotish (Ji-o-tish) is a person who predicts future events.

To my parents, Lac Vu and Ngung Doan,

my dear wife Mai Pham,

and my beloved daughter, Linh Vu.

ACKNOWLEDGMENTS

First, I would like to thank my PhD thesis advisor, Professor Klara Nahrstedt, for her great guidance throughout my PhD studies. She has led me on the transition from student to researcher during the past few years. Her insightful suggestions on my academic development were inspiring and will be invaluable in my future career. I will always be grateful for having her as my advisor.

I would like to thank the members of my thesis committee: Professor Roy Campbell, Professor Nitin Vaidya, Professor Andrew Campbell, and Professor Indranil Gupta. Professor Roy Campbell, Professor Nitin Vaidya, and Professor Andrew Campbell provided very helpful technical discussions on how to improve the research contained in my dissertation. In addition to providing useful feedback and insightful comments, Professor Gupta have collaborated with me on specific research projects and publications.

I would also like to extend my special thanks to my friend Quang Do for his collaboration with regard to Machine Learning techniques used in my dissertation.

Many thanks go to past and present Multimedia Operating Systems and Networking (MONET) research group members with whom I have interacted over the years. These members include the following: Won Jeon, Bin Yu, Jin Liang, Zhenyu Yang, Wenbo He, Hoang Nguyen, Ying Huang, Wanmin Wu, Muyuan Wang, Thadpong Pongthawornkamol, Rahul Malik, Raoul Vicente Rivas Toledano, Qiyan Wang, Ahsan Arefin, Shameem Ahmed, Debessay Fesehaye, Ravishankar Sathyam, Roger Cheng, Jigar Doshi, Zixia Huang, Pooja Agarwal, Anjali Sridha, Naveen Cherukuri, and Kurchi Subhra Hazra. Lynette Lubben deserves a special mention for unflagging efforts to ensure that all administrative procedures go through smoothly.

Last, but certainly not least, my parents deserve special recognitions for their everlasting support and care. I am grateful to my wife, Mai Pham, and my daughter, Linh Vu, for their endless love and support in my PhD study. Without them all that I have achieved would not have been possible. This dissertation is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

COADA	COalition-aware Adaptive content DownloAd protocol
COMFA	COMmunity-based data ForwArding protocol
DENTA	DENsity-aware daTa disseminAtion protocol
DTN	Delay Tolerant Network
GPS	Global Positioning System
iShare	Intra-coalition data Sharing protocol
MANET	Mobile Ad hoc Network
P2P	Peer-to-Peer
PoI	Point of Interest
UIM	University of Illinois Movement
WLAN	Wireless Local Area Network

CHAPTER 1

INTRODUCTION

1.1 Motivation

An extremely large percentage of personal devices (e.g., cell phones, PDAs, Zune, iPod, iPad) are now equipped with wireless network interfaces such as Wifi and Bluetooth. This opens the door to a wide range of decentralized and ubiquitous communications in which personal wireless devices can collaboratively create mobile peer-to-peer (P2P) networks of mobile nodes for data exchange. On the other hand, mobile users nowadays do not passively wait for the content to be pushed to their personal wireless devices as in the past. Instead, they actively use their personal wireless devices to create and share their own data. For example, mobile users can easily use their iPhones to take photos, record short video clips and share with their friends, who also use their cell phones to receive the content. As a result, content distribution for mobile users becomes increasingly important. Mobile P2P networks have drawn significant attention from the research and industry communities since they offer a low-cost and scalable solution for content distribution among personal wireless devices [1, 2, 3, 4, 5, 6, 7].

Mobile P2P networks can be classified into three main types of network: Hybrid mobile P2P Network, Mobile Ad hoc Network (MANET), and Delay Tolerant Network (DTN). Nodes in Hybrid mobile P2P Network have two main communication modes: infrastructure-based and P2P. Particularly, the infrastructure-based mode can be performed over the cellular base stations or Wifi access points. Meanwhile, the P2P communication mode is the ad hoc channel such as Wifi ad hoc and Bluetooth. Nodes in MANET only have the P2P commnication mode where they communicate with each other directly via the ad hoc channel. The communication between a pair of nodes in MANET is performed in a timely manner when the communication path between the two nodes is established and maintained for the entire communication period. Since the communication path must be maintained under network dynamics and node mobility, MANET seems to be only

applicable for the military context rather than the civilian context. Like MANET, nodes in Delay Tolerant Network only communicate via the ad hoc channel. However, unlike MANET, nodes in DTN perform the carry-and-forward paradigm where the communication between the sender and the receiver of the message is not required to be performed instantly. Instead, data forwarding is based on the opportunistic encounters of nodes in the network. Therefore, communication path between the sender and receiver is not required to be established and maintained for the entire communication period. As a result, data forwarding and content distribution in DTN suffer from a longer communication delay than those of MANET.

Data dissemination and content distribution in mobile P2P networks research literature can be divided into three major categories, in which each category is for one of the above networks. Content distribution in the Hybrid mobile P2P Network usually provides the most efficiency in terms of message delivery delay and message overhead since this network has the infrastructure to cache and distribute content messages to the recipients [8, 9, 10, 11]. However, the distribution efficiency comes from the cost of constructing and maintaining the wireless infrastructure of the network. Data dissemination in MANET, on the other hand, is based on the opportunistic ad hoc connection among mobile nodes. Moreover, in MANET the data dissemination between the sender and the receiver is successful only if there is a connected communication path between these two nodes [12, 5, 13, 14]. This requirement results in major burden on research and limits the applicability of data dissemination in MANET in civilian context where the communication path may get broken at any time due to the node mobility. Unlike MANET, data dissemination between a pair of nodes in DTN is not required to be performed in a timely manner [15, 16, 17, 18]. Instead, a long delay of data dissemination is accepted in DTN since data forwarding in DTN is performed by the intermittently connected path and the opportunistic encounters of mobile nodes. This opens a wider range of practical applications for data dissemination in civilian context. Since content distribution in Hybrid mobile P2P Network, MANET, and DTN depends on the P2P opportunistic communications among mobile nodes, the performance of these protocols depends significantly (if not fully) on the ad hoc P2P connectivity pattern created by the mobile nodes. Meanwhile, the P2P connectivity pattern of these networks depends on the node mobility, which is essentially the movement pattern of the people who carry the mobile devices. As a result, understanding people movement plays a pivotal role in the design of efficient content distribution protocols for mobile P2P networks.

Knowledge of people movement is not only crucial for content distribution in mobile P2P networks [19, 20, 21, 22, 23, 24] , but also fundamental for other research domains such as urban planning, traffic engineering, social science, and environmental science [25, 26, 27, 28, 29, 30, 31]. As a result, there has been a rich research literature on modeling and characterizing people movement from these research domains in order to understand people movement. Particularly, previous mobility models of people movement can be divided into two main categories: macroscopic-level mobility models and microscopic-level mobility models. Macroscopic-level mobility focuses on the aggregated effects of people mobility in a large scale for metropolitan, nation, and worldwide [25, 32]. The aggregated user mobility in urban context was also studied [28, 33], in which people were classified into different classes based on the socioeconomic characteristics and the aggregated spatiotemporal movement patterns of these classes were modeled. In contrast, microscopic-level mobility focuses on the movement of individuals ranging from Random Walk mobility [23] to Time-variant community mobility [34], from group mobility [19] to cluster-based mobility [22], from event-based mobility [35] to first-responder mobility [36], from Freeway mobility [37] to obstacle-based mobility [21, 38], from mathematical mobility based on social network theory [39, 40, 26] to trace-driven mobility based on real WLAN trace [41, 42]. It is understood that the movement of people is complex and depends on various environmental factors such as time, temperature, weather, etc., and socioeconomic factors such as age, gender, etc. For example, movement of firefighters in the first-responder scene differs significantly from that of shoppers in the shopping malls. Similarly, movement of students in university campuses differs from that of workers on the fields. As a result, on one hand, there has been no one-fit-all mobility model that represents all real-world scenarios. On the other hand, a deeper understanding of people movement remains challenging, especially in the level that can be useful for the efficient design of content distribution protocols in mobile P2P networks.

In this thesis, we present a framework to characterize and leverage people movement for content distribution in mobile P2P networks. Particularly, we characterize two typical classes of people movement found in numerous real-world scenarios: Schelling behavior and repetitive behavior. For each of these movement classes, we characterize people movement to draw the key properties, which are used to design new protocols for improvements of content distribution in mobile P2P networks.

We first observe that the grouping behavior of people, who share the mutual content interest and move towards the same targeted destination (i.e., Point of Interest), exists naturally in numerous real-world scenarios. We also observe that the grouping behavior of people movement is similar

to the segregation behavior (or Schelling behavior) presented by Thomas Schelling, a Nobel prize winner in economics. We then characterize the Schelling behavior by simulating people movement on real Google maps and by modeling people movement using the Mobius modeling tool [43]. The characterization study shows that when people movement exhibits the Schelling behavior, mobile nodes carried by these people create a dynamic coalition P2P network, in which the size of coalitions formed by co-located mobile nodes increases at the closer distance from the PoI. More importantly, the characterization study also shows that the coalition pattern of the dynamic coalition P2P network follows an exponential function with respect to the distance from the PoI. We then exploit the Schelling behavior of people movement in the design of three content distribution protocols: COADA [9, 43], iShare [10], and DENTA [11]. Particularly, COADA combines cellular and P2P interfaces of the mobile devices, and exploits the exponential-coalition-size function to adaptively tune the cellular download timer to reduce the cellular download and meet the file download deadline of the cellular users. iShare provides a lightweight solution so that the nodes in the same coalition can share downloaded data in a fair fashion. DENTA leverages the uneven network density of the dynamic coalition P2P network to disseminate data messages from the Wifi base stations at the Point of Interest to the coming mobile nodes. We evaluate these three protocols and compare them with other state-of-the art content distribution solutions. The evaluation results show that our protocols outperform other alternatives and provide improvements of content distribution for mobile P2P networks.

In the second part of thesis, we study the repetitive movement behavior of people and leverage the regularity of people movement for improvements of content distribution. It is believed that people exhibit regular movement pattern in their daily activities. For example, students take classes in certain weekdays, professors give lectures at certain time slots, and workers do their routines at work regularly. In order to understand the regularity of people movement, we opt for collecting the real movement trace of people at University of Illinois campus. To this end, we first propose a novel methodology to collect people movement trace using mobile phone. We then apply this methodology to implement a scanning system called UIM, which stands for University of Illinois Movement, to collect Wifi trace and Bluetooth trace [44]. The UIM system then is deployed on 123 Google Android phones carried by faculties, staff, and students in the University of Illinois campus from March 2010 to August 2010. The collected Wifi access point information is then used to infer location and the collected Bluetooth MACs are used to infer social contact. The

inferred information is used in the characterization study of people movement and the result from the characterization study confirms that people exhibit a high degree of regularity in their daily movements. Although the regularity of people movement can also be found by studying the daily calendars of people [45], we believe our collected data set provides a much finer granularity of people movement. Due to its finer grain, our collected movement trace can be used to learn other characteristics of people movement such as real social contact, accurate contact duration, and exact stay duration at the location, which can not be derived accurately from the calendars. We then propose a method named Jyotish that exploits the regularity of people movement found in the collected joint Wifi/Bluetooth trace to construct a predictive model to predict future location, stay duration at the location, and social contact [46]. Finally, we leverage the constructed predictive model to design a new content distribution protocol named COMFA to expedite data messages in Community-based Delay Tolerant Networks. We evaluate and compare the performance of COMFA with Epidemic routing [17] and Prophet routing [47] over the real trace collected by the UIM system. The evaluation shows that COMFA outperforms the other alternatives considerably.

1.2 Contributions and Dissertation Outline

1.2.1 Contributions

This thesis will show that the movement behavior of people can be characterized and leveraged for the design of efficient content distribution protocols in mobile P2P networks. In summary, the thesis has the following contributions:

1. We present a framework to characterize and leverage people movement for improvements of content distribution in mobile P2P networks.

2. We present the Schelling behavior of people movement. Schelling behavior essentially represents the instant grouping behavior of co-located people when they move towards the same Point of Interest. The Schelling behavior shows the impacts of individual movement (i.e., the microscopic level) on the aggregated network density (i.e., the macroscopic level), which has not been explored in existing mobility models.

3. We characterize Schelling behavior of people movement and validate its properties by simulating people movement on real Google maps and by modeling people movement in the Mobius

modeling tool.

4. We exploit the properties of the Schelling behavior to design three content distribution protocols for mobile P2P networks, including COADA, iShare, and DENTA.

5. We propose a novel trace collection methodology to collect people movement trace by using mobile phones. Then, we apply the methodology to implement the UIM scanning system to collect Wifi trace and Bluetooth trace. The joint Wifi/Bluetooth trace is used to infer location and social contact information. To the best of our knowledge, the UIM system is the first scanning system, which collects both location information and social contacts of people movement.

6. We deploy the UIM system on 123 Google Android phones carried by participants at the University of Illinois campus from March 2010 to August 2010. The collected joint Wifi/Bluetooth trace provides a rich data set of people movement.

7. To the best of our knowledge, our proposed Jyotish method is the first method to construct a predictive model of people movement from the joint Wifi/Bluetooth trace.

8. To the best of our knowledge, our constructed predictive model is the first method to provide predictions for location, stay duration, and social contact altogether.

9. We leverage the regularity of people movement to design COMFA, a new content distribution protocol for Community-based Delay Tolerant Networks. COMFA is evaluated over the real movement trace collected by the UIM system.

1.2.2 Dissertation outline

Figure 1.1 shows the content of our thesis, which is organized as follows. We first present the framework of characterizing and leveraging people movement for content distribution in mobile P2P networks in Chapter 2. Then, in Chapter 3 we present the Schelling's segregation model and characterize the Schelling behavior of people movement. After that, we present three content distribution protocols to exploit the properties of the Schelling behavior to improve content distribution, including COADA (Chapter 4), iShare (Chapter 5), and DENTA (Chapter 6).

Starting from Chapter 7, we characterize and leverage the repetitive behavior of people movement to improve content distribution in mobile P2P networks. Particularly, in Chapter 7 we present

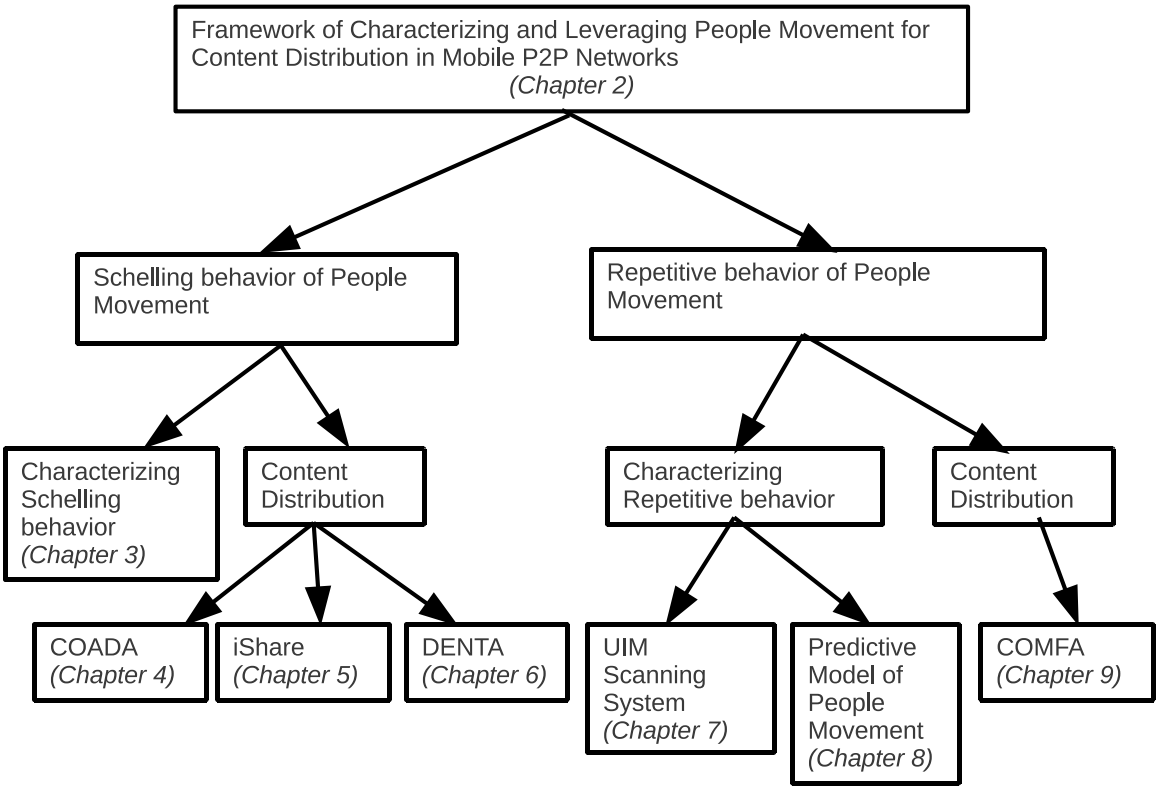


Figure 1.1: Thesis content and organization

the design, implementation, and deployment of UIM scanning system, which collects Wifi and Bluetooth traces. This joint Wifi/Bluetooth trace is used to infer the repetitive pattern and to construct the predictive model of people movement in Chapter 8. Then, we exploit the constructed predictive model to design a new forwarding scheme called COMFA for Community-based Delay Tolerant Networks in Chapter 9. We conclude the thesis and discuss the future work in Chapter 10.

In this thesis, we also include several appendix chapters. The first appendix in Chapter A presents operators in Relational Algebra, which are used in the thesis for the presentation clarification of the collected data set in Chapter 8. In the appedix Chapter B, we present the measurement results obtained from our UIM collected data set. Finally, we present a Hybrid Epidemic Routing protocol in the appendix Chapter C to evaluate the performance of a data dissemination scheme, which uses both Wifi and Bluetooth interfaces of the mobile nodes to forward data.

CHAPTER 2

FRAMEWORK OF CHARACTERIZING AND LEVERAGING PEOPLE MOVEMENT FOR CONTENT DISTRIBUTION IN MOBILE P2P NETWORKS

In this chapter, we first present the framework to characterize and leverage people movement for content distribution protocols in mobile P2P networks in Section 2.1. Then, we present the system model in Section 2.2. Finally, we present the design objectives in Section 2.3.

2.1 Framework Description

Figure 2.1 shows our framework of characterizing and leveraging people movement for content distribution in mobile P2P networks. Particularly, we study the Schelling behavior of people movement in the first part of the thesis and the repetitive behavior of people movement in the second part. In each part, we start by characterizing the movement behavior and then exploiting the lessons from the characterization study to design new content distribution protocols.

For the first part of the thesis (the left hand side of the Figure 2.1), we use the top-down approach by first obtaining the qualitative properties of the mobility model and then validating these properties from the extensive simulation. Particularly, we first analyze the original segregation model proposed by Thomas Schelling and find that its properties exist in numerous real-world

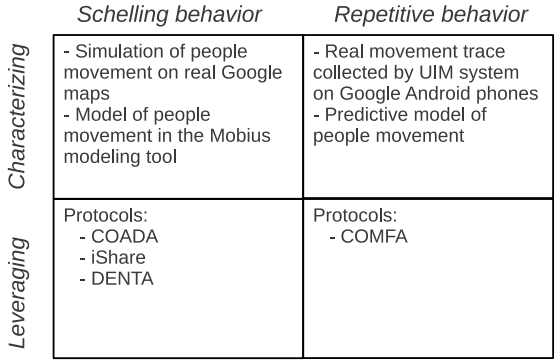


Figure 2.1: The framework of characterizing and leveraging people movement for content distribution in mobile P2P networks

scenarios, in which the co-located people may form groups and collaboratively share data messages using their wireless devices when they are moving towards the same targeted destination (i.e., Point of Interest). The grouping behavior of people (or their mobile devices) when they are moving towards the mutual Point of Interest is called the "Schelling behavior". We find that when mobile nodes exhibit Schelling behavior, the network has two important properties: (1) co-located mobile nodes form "moving" coalitions, and (2) the coalition size increases at the closer distance from the PoI. With this analysis, we have the mobility model of people and the network model of the devices carried by these people. We then validate the mobility model (i.e., Schelling behavior) by: (1) simulating the movement on real Google maps of mobile nodes along with the shortest paths returned by Google Map APIs, and (2) modeling the movement of mobile nodes along with the probabilistic paths returned by our probabilistic path selection model in the Mobius modeling tool [48]. Our validation study confirms the two above properties, which then are exploited in the design of three protocols to improve content distribution in mobile P2P networks, including COADA [9, 43] , iShare [10], and DENTA [11].

For the second part of the thesis (the right hand side of Figure 2.1), we use the bottom-up approach by first collecting real people movement trace and then extracting movement properties from the collected trace for content distribution. It is believed that people movement exhibits a high degree of repetition, in which people usually visit regular places and make regular social contacts for their daily activities [49]. However, there has been no fine-grained people movement trace to validate this observation. We opt for this bottom-up approach to collect a fine-grained people movement trace since we want not only to validate the regularity of people movement in their daily activities, but also to explore other information (e.g., social contacts, stay duration at the location, etc.) provided by the fine granularity of the collected trace. To this end, we propose a novel methodology to collect the fine-grained people movement trace using mobile phone and implement the UIM scanning system, which collects MAC addresses of Wifi access points and Bluetooth-enabled devices in the proximity of the experiment phones. The UIM system is deployed on Google Android phones carried by 123 faculties, staff, and students in University of Illinois campus from March 2010 to August 2010. The collected MAC addresses of Wifi access points are used to infer location information and the collected Bluetooth MACs are used to infer social contact. The inferred location information and social contacts then are used in the characterization study, which shows that people movement exhibits a high degree of repetition. Particularly, people

visit regular locations and have contacts with a regular set of other people for their daily schedules. This motivates us to construct a predictive model from the collected joint Wifi/Bluetooth trace to provide predictions for location, stay duration at the location, and social contact [46]. Finally, we leverage the predictive model to design a new data forwarding protocol named COMFA, which maximizes the message delivery probability and preserves message delivery deadline for content distribution in the Community-based Delay Tolerant Networks. We compare the performance of COMFA with Prophet routing and Epidemic routing over the real movement trace collected by the UIM system. The evaluation results show that COMFA outperforms other alternatives by considerably reducing the message delivery delay and message overhead.

Besides the two major parts of the thesis, we also present several appendix chapters. The first appendix in Chapter A presents operators in Relational Algebra, which are used in the thesis for the presentation clarification of the collected data set in Chapter 8. In the appedix Chapter B, we present the measurement results obtained from our UIM collected data set. Finally, we present a Hybrid Epidemic Routing protocol in the appendix Chapter C to evaluate the performance of a data dissemination scheme, which uses both Wifi and Bluetooth interfaces of the mobile nodes to forward data.

In this thesis, we present the COADA protocol for the content distribution in Dynamic Coalition P2P network and the COMFA protocol for the content distribution in Community-based Delay Tolerant network. In our context, coalition and community can be distinguished as follows. On one hand, a coalition or a community refers to a group of people who share mutual interests or socioeconomic characteristics. On the other hand, while coalition is formed by co-located people for a short period (i.e., in the order of hours or minutes), the community is formed by people in the same geographical area over the long period of time (i.e., in the order of weeks or months). In other words, the community is more stable than coalition. The coalition can be found in the shopping street scenario, where shoppers form coalition while they are moving towards the same shop. After the shoppers arrive at the shop, the coalition may not exist anymore. In contrast, the community can be found in the university campus environment, where students in the same class can form a community and they meet each other during the class period every week for the entire semester.

2.2 System Model

In this section, we present the network model and data model that are used in the design of content distribution protocols in the thesis.

2.2.1 Network model

We focus on a mobile P2P network of mobile nodes. Each mobile node is assumed to have one mandatory P2P interface such as Wifi or Bluetooth. This P2P interface is used in all of our content distribution protocols including COADA, iShare, DENTA, and COMFA. The mobile node is also assumed to have the cellular interface to work with COADA and iShare while mobile node is assumed to be able to communicate in infrastructure-mode to work with DENTA. For COMFA, the mobile node only needs the P2P interface.

Mobile nodes in our network exhibit different movement behaviors, depending on specific scenarios. For example, nodes may exhibit the Schelling behavior and perform the instant grouping with other co-located nodes when they are moving towards a Point of Interest. For the Schelling behavior, nodes can use COADA, iShare, and DENTA protocols to download and exchange data messages. On the other hand, if the node exhibits the repetitive movement behavior, it can use COMFA to forward data messages.

Since the mobile nodes are carried by people, we use the terms "mobile nodes" and "mobile users" interchangeably in this thesis.

2.2.2 Data model

The data exchanged among mobile nodes is in the format of text or video, depending on the specific protocol. For COADA and iShare, mobile nodes download the video file content from the content server via the cellular interface and exchange downloaded data via the P2P interface. For DENTA, mobile nodes exchange the text messages via the P2P interface and also download text messages from the Wifi access points at the Point of Interest. For COMFA, mobile nodes exchange the text/video messages by using only the P2P interface.

2.3 Design Objectives

The ultimate objective of this thesis is to improve content distribution in mobile P2P networks. Since the connectivity and topology of the networks depend on mobility of the mobile devices (or the movement behavior of the mobile device carrier), we characterize people movement and then apply the learned lessons from the characterization study for the design of content distribution protocols. As the result, the thesis has two objectives, including: (1) understanding the correct movement behavior of people, and (2) improving the content distribution in mobile P2P networks.

For the first objective, we use simulation, modeling technique, and real system deployment to characterize the Schelling behavior and the repetitive behavior of people movement. Since the Schelling behavior (or instant grouping behavior) of people movement exists when a large number of people is moving towards the same Point of Interest, we do not have enough resources and facilities to collect real trace of people movement to validate the Schelling behavior. Instead, we first simulate movement of multiple nodes on real Google maps where the nodes take the paths returned from Google Map APIs towards the Point of Interest. We further validate the Schelling behavior by modeling the movement of mobile nodes by using the Mobius modeling tool. For repetitive behavior of people movement, we propose a new methodology to collect people movement using mobile phone. Then, we apply the methodology to implement and deploy the UIM scanning system on 123 Google Android phones carried by students, staff, and faculties in University of Illinois campus to collect a real people movement trace from March to August 2010. The trace collected by the UIM system is used to characterize the regularity of people movement and construct the predictive model of people movement.

For the second objective, we leverage the lessons learned from the characterization study for the design of content distribution protocols in mobile P2P networks. For content distribution protocols, our objectives include improving message delivery probability, reducing message delivery delay, and limiting message overhead.

CHAPTER 3

CHARACTERIZING SCHELLING BEHAVIOR OF PEOPLE MOVEMENT

3.1 Introduction

Recently, a large percentage of wireless devices comes equipped with additional short-range ad hoc (or Peer-to-Peer) wireless communication interfaces such as Wifi and Bluetooth. P2P communication thus has been taken extensively into consideration in the design of content distribution protocols for mobile P2P networks [8, 16, 50, 51]. However, leveraging P2P communication remains challenging since the P2P communication is limited by a short transmission range and thus becomes broken under mobility of cellular users. More importantly, cellular users may not have the intermediate incentive to communicate in the P2P channel, which is highly energy-consuming.

Let us consider a shopping street scenario where customers walk to their interested shops and download the product preview video to their cell phones using the cellular connectivity, and at the same time they exchange the video via the Bluetooth or 802.11 wireless interfaces of the phones. Given two co-located customers A and B, according to previous protocols [8, 16, 50], A and B are required to collaboratively exchange/forward messages. However, A and B may have different targeted shops, so they may move towards different directions in very near future, causing their wireless connection to break. Further, if A is interested in jewelry and B is interested in digital cameras, what is the immediate incentive for A to disseminate the messages about digital cameras from B, and vice versa? We therefore believe that sharing mutual content interest is crucial to motivate people (with their smart phones) to collaboratively exchange content messages.

Interestingly, we observe numerous scenarios where co-located people motivate themselves to collaborate since they share mutual content interests. For example, smart phones of co-located audiences or co-located soccer fans may group to exchange data via the P2P channel, while these mobile users are heading towards the same Points of Interest (PoI) such as the concert theater or the soccer stadium. Also, audiences of an exhibition or students in a campus can group to exchange

messages while heading towards the same destinations such as exhibition halls or classrooms. More interestingly, the grouping behavior of people who share similarities was presented in the segregation model, one of the most cited models in economics, proposed by Thomas Schelling [52]. According to the segregation model, people move apart from each other if they have different economic/social interests; whereas, they group if they share mutual economic/social interests. We find that this grouping behavior of the Schelling model can be characterized and exploited to improve content distribution in mobile P2P networks.

In this chapter, we perform an analysis on the original Schelling model to highlight its important properties in Section 3.2. We present real world scenarios where these properties of the Schelling model exist and can be exploited for content distribution in Section 3.3. In Section 3.4, we present the Dynamic Coalition P2P network created by mobile nodes carried by mobile users whose movements exhibit the Schelling behavior. In order to understand the coalition pattern of this network, we study the relationship between the size of coalition formed by mobile nodes and the distance from the mobile nodes to the PoI on three real maps taken from Google Map [53] in Section 3.5. Our study shows that the coalition size distribution follows an exponential function with respect to the distance from the mobile nodes to the PoI. This result is further confirmed by our study of coalition pattern using Mobius modeling tool [48] in Section 3.6. In Section 3.7, we present the related work. Finally, in Section 3.8 we present briefly how the Dynamic Coalition Mobile P2P network is exploited to design content distribution protocols in Chapter 4, Chapter 5, and chapter 6 of this thesis.

3.2 Schelling Model

In our context, the term *similar individuals* or *similar people* is defined as follows: *individuals are considered similar if they share common similarity (race, education, community) or mutual interest on some topic (books, music, movies).*

3.2.1 Schelling's original model

In 1969, Thomas Schelling, a Nobel-prize winner in economics, proposed one of the most cited models named segregation model in economics to explain how similar people (i.e., people with same race, education, community) group in American neighborhoods [52]. According to Schelling's

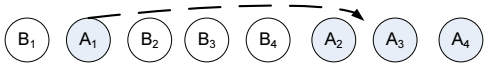


Figure 3.1: In Schelling's original model, people always move towards their similar neighborhoods

segregation model, the grouping is created by movements of individuals who want at least a certain portion of similar neighbors. In other words, when a person is unsatisfied with his neighborhood, he moves towards a place where he has more similar neighbors. Such movements eventually create clusters of similar individuals. Figure 3.1 shows the idea of Schelling's model where a circle denotes an individual and shading patterns represent different interests. In this example, A_1 moves towards its closest and similar neighbor, A_2. When everyone is satisfied with their neighborhoods, the clustering reaches the stable equilibrium. In what follows, we use the terms group and cluster interchangeably.

3.2.2 Analysis of Schelling's model

We introduce two important properties of the Schelling's model, which are later exploited by our protocols in Chapter 4, Chapter 5, and Chapter 6 for improving data accessibility in mobile P2P networks.

3.2.2.1 Property 1: Density of similar individuals increases in proximity of clusters

This property directly follows Schelling's original model since individuals move towards their desired neighborhoods and thus create clusters of similar individuals at these neighborhoods. As a result, the density of similar individuals increases significantly in proximity of these clusters.

3.2.2.2 Property 2: Similar individuals form "moving" clusters during their movements

According to Schelling, each individual always moves to his final cluster where he is satisfied with the neighborhood and stays. In Schelling's model, on the way to their final clusters, similar individuals form small clusters. However, individuals at the boundary of these small clusters may not be satisfied with their current mixed neighborhoods. Thus, they tend to move towards bigger clusters where they have better (similar) neighborhoods. When an individual at the boundary leaves, other inner individuals form the boundary; this again might cause them to leave. This process creates small "moving" clusters, which merge to bigger clusters.

3.3 Schelling Behavior and Content Distribution in Mobile P2P Networks

3.3.1 Schelling behavior

Schelling's original model focuses on economic and social phenomena where individuals gradually form groups on a very large timescale. For example, the formation of a China town in a city might take decades. However, in the context of wireless technologies, we observe numerous scenarios where mobile wireless devices carried by similar people (people share mutual interests on some topic such as books, music, movies) exhibit Schelling's model on a much smaller timescale. For instance, co-located customers can group for 20 minutes and exchange their mutually favorite product information via their wireless handheld devices, while heading towards the same shopping mall. Further, Schelling's model originally focuses only on the outcome of the grouping process (or the final clusters). Meanwhile, we observe that in the context of wireless technologies, not only the outcome but also the grouping process itself can be exploited to expedite data dissemination. This motivates us to study the analogies of Schelling's segregation model, where the two above properties exist in a much smaller timescale and the grouping occurs during the physical movements of people carrying wireless devices. In what follows, we use the term Schelling behavior to denote the analogies of the Schelling's segregation model. In the context of wireless technology, the Schelling behavior represents the "instant" grouping behavior of people as we present in the following sections.

3.3.2 Schelling behavior and wireless technology

There are many real world scenarios where mobile wireless technology and Schelling behavior co-exist. In the scenarios presented below, people exhibit an "instant" grouping behavior since they only group while they move towards the targeted destinations (or Point of Interest). After that, the group may not exist anymore.

Our first scenario can be found in the commercial sector. Let us consider a shopping street where customers cluster while arriving at their targeted shops. In this scenario, wireless base stations at shops can broadcast product advertisements, hot sales, discounts. Meanwhile, customers are individuals in the Schelling's model who walk to shops and can form groups to exchange their opinions, reviews, and comments about their mutually interested products via their wireless personal devices. The sizes of the groups formed by co-located customers grow at the closer distance

from the targeted shops (the second property) and the density of customers gets maximum at the shops (the first property). The Schelling behavior also exists when the customers drive towards the targeted shops and at the same time, the customers download the product review video from the cellular interface of their cell phones and exchange the downloaded content via the P2P interface such as Wifi and Bluetooth.

Our second scenario is a campus life where places such as book stores, libraries, and class rooms are visited frequently by university students. These places represent final clusters and students represent moving individuals in the Schelling's model. Similar to shopping street scenario, wireless base stations at these places broadcast announcements and advertisements to the coming students. Meanwhile, coming students can form groups due to their co-locations and similar targeted places to exchange their information during their movements. Again, at the closer distance from these places, the density of students reaches maximum (the first property) and the sizes of groups formed by these students increase (the second property).

Our third scenario is a social event such as an art exhibition or an outdoor music concert in the downtown area of a city. The event "attracts" interested audiences and plays the role of a final cluster in the Schelling's model. The wireless base station at the event can broadcast advertisements, content and showtime of the event to arriving audiences. These audiences can form groups and exchange their opinions and comments about the event via their personal wireless devices. The group size grows up and the node density increases at the closer distance from the PoI. Similarly, audiences who drive towards the location of the social event can group and download the event program video from the cellular interface to their cell phones, and then exchange the downloaded data via the P2P interface to expedite the downloading process.

In summary, we believe these above examples represent popular classes of scenarios where (1) the movement of mobile nodes (carried by mobile users) exhibits the Schelling behavior, and (2) these mobile nodes, which share the same content interest, form a mobile P2P network to exchange the mutual content. We name this network "Dynamic Coalition Peer-to-Peer Network" and present the network in the following section.

3.4 Dynamic Coalition P2P Network

In our context, the dynamic coalition P2P network consists of two concepts: Point of Interest and mobile node. Point of Interest (PoI) represents a fixed place destination such as soccer stadium,

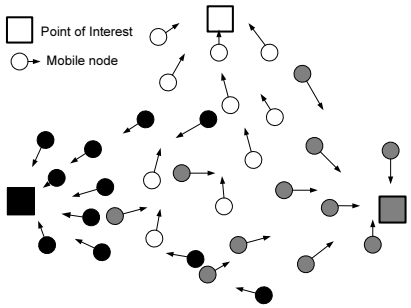


Figure 3.2: Dynamic Coalition P2P Network

concert theater, shopping mall, etc., where people move towards. The mobile node concept represents the cell phone of pedestrians or drivers, who are heading towards the PoI. When moving towards the PoI, co-located people form groups and the mobile nodes carried by these people form coalitions to exchange data messages. Henceforth, we use the terms "mobile nodes" and "mobile users" interchangeably. A coalition is considered as a "Communication Coalition" and is defined recursively as follows:

1. Nodes are connected in collaborative one-hop manner: If mobile nodes n_1 and n_2 are in the communication range of each other via the P2P channel, they are moving towards the same Point of Interest, and they share the same PoI mutual content interest, then n_1 and n_2 belong to the same communication coalition. In this case, n_1 and n_2 are one-hop neighbors of each other.

2. Nodes are in transitive relation: If nodes n_1 and n_2 belong to the communication coalition C, and n_2 and n_3 belong to the communication coalition C, then n_1 and n_3 belong to the communication coalition C. This transitive property means that communication coalition can be expanded to multiple hop networks.

3. The coalition is per topic content/interest, so one mobile node n may belong to multiple coalitions at the same time. For example, if n is downloading two content files in the topics of soccer and classical music at the same time, then n may belong to two different coalitions of "soccer fan" and "classical music fan". Also, one cluster of co-located people may form multiple coalitions if these people exchange content of different interests.

Henceforth, we use the terms *communication coalition* and *coalition* interchangeably. The dynamic coalition P2P network has two following important properties: (1) only mobile nodes in the same coalition collaborate to exchange data via the P2P channel, and (2) the size of coalitions

formed by mobile nodes becomes bigger at closer distance from the PoI. Figure 3.2 shows a dynamic coalition P2P network with three Points of Interest and mobile nodes in which the shading pattern represents the node's content interest. In this figure, there is a transition in coalition pattern in the dynamic coalition P2P network. Particularly, at the closer distance from the PoI, the density of nodes heading towards the PoI increases. In contrast, at the farther distance from the PoI, the density of nodes moving towards the PoI becomes sparse. For a further understanding of this coalition pattern, we study the coalition pattern on three real maps taken from Google Map and the coalition pattern modeled by the Mobius modeling tool [48].

3.5 Validation Study of Coalition Pattern on Real Google Maps

To understand the coalition pattern of mobile nodes in realistic scenarios, we perform the following experiment. First, we select a real map M_P from Google Map with a Point of Interest P at the center of the map. Second, we select N nodes at random locations on M_P in the surrounding areas of P. Let these selected locations of N nodes be their initial locations. We simulate two different scenarios for pedestrians and cars. If the map M_P is for the pedestrians, the distance from N nodes to P is shorter than that of the map M_P, which is used for cars. Third, we use Google Map APIs [53] to find the routes, which we call Google routes, for nodes from their initial locations to P. We observe that the routes returned by Google Map APIs usually represent the shortest routes. Fourth, we assume that all nodes arrive at P at approximately the same time and thus nodes closer to P will depart towards P later than nodes farther from P. In the simulation, N nodes will start at their initial locations, then they follows their Google routes to move towards P. When these nodes are moving towards P, we calculate the average coalition size at different distances from P. Here, the communication range of the nodes is 200 (m). In other words, if the distance between two nodes n_1 and n_2 is less than 200 (m) while they are moving towards P, n_1 and n_2 are in the same coalition and they are one-hop neighbors. The coalition can be expanded to multiple hops as presented in the definition of Coalition in Section 3.4.

We perform our simulation on three real maps for three realistic scenarios. The first map is for the Assembly Hall, Champaign, Illinois (i.e., longitude=40.096327, latitude=-88.238583) where the basketball matches or concert events are held at the University of Illinois. For this map, we select 320 pedestrians (e.g., basketball fans) at random locations as shown in Figure 3.3. When the simulation starts, these 300 nodes move from their initial locations to the Assembly Hall (i.e.,

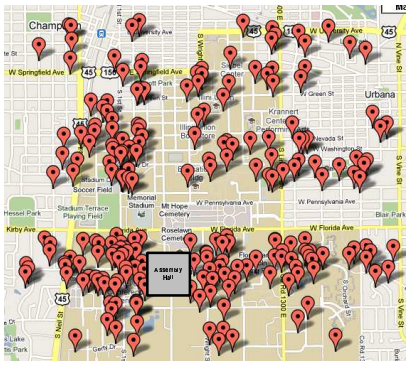


Figure 3.3: Initial positions of mobile nodes before simulation, with PoI at Assembly Hall.

Figure 3.4: Initial positions of mobile nodes before simulation, with PoI at Time Square area.

Figure 3.5: Initial positions of mobile nodes before simulation, with PoI at Market Place Mall.

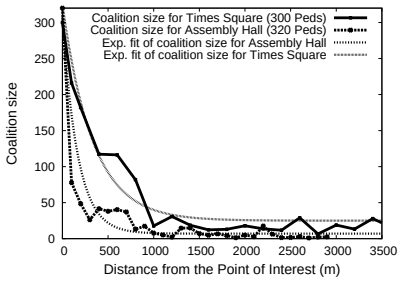


Figure 3.6: Coalition size distribution of pedestrians moving towards Assembly Hall or Times Square

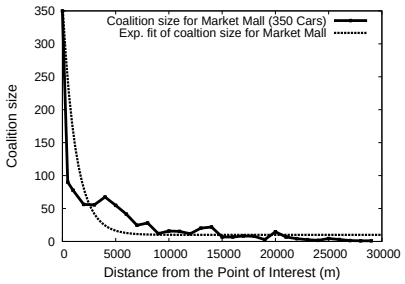


Figure 3.7: Coalition size distribution of cars moving towards Market Place Mall

PoI) along with their Google routes. Similarly, we select 300 pedestrians at random locations in the surrounding area of the Times Square (i.e., longitude=40.759903, latitude=-73.984294) in New York City as shown in Figure 3.4. When the simulation starts, these 300 nodes move from their initial locations to the Times Square (i.e., PoI) along with the Google routes. The third map is a the Market Shopping Mall in Champaign, Illinois (i.e., longitude=40.142927, latitude=-88.244419) as shown in Figure 3.5. Here, we select 350 cars at random locations with farther distances than those in Assembly Hall and Times Square simulations. When the simulation starts, these 350 nodes move from their initial locations to the Market Mall (i.e., PoI) along with their Google routes. For this simulation, the speed of cars (i.e., 8-10 (m/s)) is much higher than that of pedestrians (i.e., 1-2 (m/s)) in the two previous simulations for Assembly Hall and Times Square.

Figures 3.6 and 3.7 show that the coalition size increases significantly at the closer distance from the Point of Interest, although the actual value of the coalition size depends on the street configuration of the selected map. More interestingly, we found that the coalition size fits very well the exponential function. In particular, the exponential fitting functions of Assembly Hall, Times Square, Market Mall experiments are $y(d) = 300e^{-0.006d} + 7$, $y(d) = 300e^{-0.003d} + 25$, and $y(d) = 350e^{-0.0008d} + 10$, respectively. Here, d is the distance from the current location of mobile nodes to the PoI and $y(d)$ is the coalition size, which is an exponential function with respect to the

distance d.

In conclusion, the simulations of people/car movement on three real Google maps shows that the coalition size exhibits an exponential function in which the coalition size increases exponentially with respect to the distance from the PoI. We observe that the routes returned by Google Map APIs usually are the shortest routes. Meanwhile, recent studies stated that people with different socioeconomic characteristics may prefer different routes [54] when they move from the same starting location to the same Point of Interest. Therefore, in the next section, we investigate a further study to confirm whether the movement of people along with the shortest routes returned by Google Map APIs results in the exponential coalition size.

3.6 Modeling Coalition Pattern in Mobius Modeling Tool

3.6.1 Motivation

As we discussed in Section 3.5, the Google routes returned by the Google Map APIs are usually the shortest routes. However, the analysis of the pedestrian movement trace in Koblenz city shows that given the same pair of starting location and destination, different people may choose different routes when they move from the starting location towards the destination [31]. Particularly, people prefer the shorter route towards the destination, but they do not always select the shortest route. Another recent study claims that people with different socioeconomic characteristics may prefer different routes [54] when they move from the same starting location towards the same destination (e.g., Point of Interest). In other words, people usually select the shorter route towards the destination with a higher probability, but the final choice depends on the person's socioeconomic characteristics such as age, gender, and other environmental factors such as weather, social events, time of day.

In order to represent the aforementioned route selection behavior of people, we present a probabilistic model of route selection, in which a shorter route towards the PoI is selected with a higher probability. Then, we use this route selection model as the mobility model of mobile nodes in two different street configurations: Manhattan street and random street maps. For each configuration (or map), we have two Points of Interest. Then, the street configurations and the probabilistic route selection model are simulated by the Mobius modeling tool [48], which has been extensively used to model various stochastic systems. Finally, we study the coalition patterns resulted from the movement of mobile nodes towards their PoI in the Mobius modeling tool.

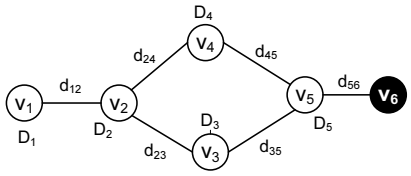


Figure 3.8: Sample street configuration with PoI at v_6

3.6.2 Probabilistic route selection

We present a probabilistic route selection model, which assigns a higher probability to the shorter route from the starting location to the PoI. We believe our probabilistic model represents the route selection behavior of people better than previous mobility models [23, 19, 55] since it introduces the flexibility into the route selection and represents the realistic movement behavior as studied in [54, 31]. Particularly, we use the probability to represent the uncertainty of the route selection and the impacts of socioeconomic factors in people's route selection. The probabilistic model of route selection is presented below.

We consider a mobile node n moving within a physical movement area, which consists of multiple street segments and intersections. Each street segment has two intersections at two ends. To simplify the model, we assume that the PoI is also an intersection. From its current location, node n follows consecutive street segments towards the PoI. We use an undirected graph $G =< V, E >$ to present the movement area as follows. The i^{th} intersection in the movement area is represented by a vertex $v_i \in V$. Meanwhile, a street segment connecting the i^{th} and j^{th} intersections is represented by an edge $(v_i, v_j) \in E$. An edge $(v_i, v_j) \in E$ has a weight d_{ij} representing the physical distance between v_i and v_j. So $d_{ij} = d(v_i, v_j)$ and we assume that the distance from v_i to v_j is equal to that from v_j to v_i (i.e., $d_{ij} = d_{ji}$ and G is an undirected graph). Further, each vertex $v_i \in V$ has a value D_i representing the shortest distance from v_i to the PoI. Figure 3.8 shows an example of the graph representation of a movement area. In this figure, v_6 is the PoI and mobile nodes from other vertices will move towards v_6, and each vertex maintains the shortest distance to v_6.

In our model, a route towards the PoI consists of vertexes in V. The intuition of selecting a route is that a person prefers a shorter route towards the PoI with a higher probability. So, we define the transition probability p_{ij} as the probability that the node n selects v_j as the next vertex in the route towards the PoI if n currently stays at vertex v_i. The next step is to calculate p_{ij}. Let A_i be the set of adjacent vertexes of v_i, and each vertex of A_i has a shorter distance to the PoI than v_i. In other words, if the node n is staying at the current intersection v_i, n will not select an intersection

v_j as its next intersection towards the PoI if $D_j \geq D_i$. Formally, $A_i = \{v_j : D_j < D_i, (v_i, v_j) \in E\}$, and we have:

$$p_{ij} = \begin{cases} \frac{d_{ij}+D_j}{\sum_{v_k \in A_i}(d_{ik}+D_k)} & ,\forall v_j \in A_i \\ 0 & ,\forall v_j \notin A_i \end{cases} \tag{3.1}$$

In Figure 3.8, if node n stays at v_2 and its PoI is v_6, then n selects v_3 as the next vertex with the probability $p_{23} = \frac{d_{23}+D_3}{d_{23}+D_3+d_{24}+D_4}$. Given a street configuration with intersections and a PoI, we can construct a graph G to represent this street configuration, in which each intersection is represented by one vertex in $v \in G$.

Notice that in previous section since we use Google Map APIs to find the route for mobile nodes, nodes have to follow the routes returned by Google Maps APIs, which is usually the shortest path. Here, we present the probabilistic route selection model, which allows us to vary and customize the route selection process and study people movement in different street configurations and different levels of detail. The next step is to study the coalition patterns formed by nodes moving in specific street configurations, in which each node uses the probabilistic model to select the route towards the PoI. For this study, we use the Mobibus modeling tool as presented in the following section.

3.6.3 Exponential coalition size resulted from probabilistic route selection

We use the Mobius modeling tool [48] to study the coalition patterns of two different movement areas (or street configurations): Manhattan street configuration and Random street configuration as shown in Figure 3.9 and Figure 3.10. Mobius modeling tool is the Stochastic Activity Network, which has been used extensively for modeling the behavior of distributed systems and networking systems. We are interested in the coalition size distribution at the steady state of the system since we believe the steady state provides insightful movement characteristics of the system. The Mobius modeling tool is used to model the street configuration and the route selection as follows.

Figure 3.9 shows the Manhattan street configuration, in which the distance between two adjacent vertexes is $l = 100(m)$ or l is the length of a street segment. Figure 3.10 shows the Random street configuration where we specify the distance with the street segment. For each movement area, we have 1000 mobile nodes moving towards the two PoIs v_1 and v_{25} with a random speed s in the rage [1.0, 2.0] (m/s). We select the movement speed in the range of [1.0,2.0] since we believe this is the normal walking speed of pedestrians in reality.

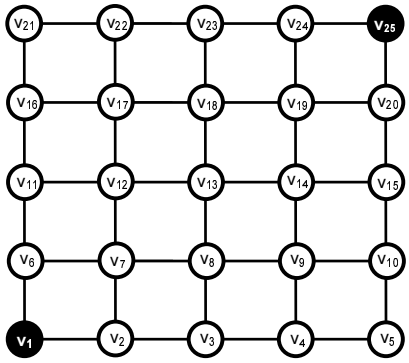


Figure 3.9: Manhattan street configuration with two PoIs at v_1 and v_{25}

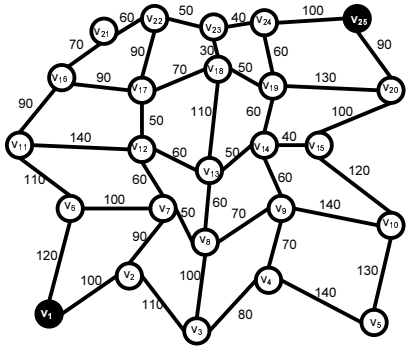


Figure 3.10: Random street configuration with two PoIs at v_1 and v_{25}

First, we use an undirected graph $G =< V, E >$ to represent the street configuration. Then, for each vertex $v_i \in V$, we create one atomic model in Mobius and specify the transition probability of the case activity in this atomic model as the probability calculated by the Equation (3.1). The duration a node n stays in the activity of the atomic model follows an exponential distribution with the mean $\lceil l/s \rceil$ seconds. Notice that $\lceil l/s \rceil$ is the traveling time node n spends to travel the street segment of length l. By using the exponential distribution with the mean $\lceil l/s \rceil$ as the traveling time of n on the street segment of length l, we use Mobius to model the movement of n. On arriving at one PoI, nodes stay for a random period, which follows an exponential distribution. For each street configuration, we study the impact of the staying period at the PoIs on the coalition size distribution. Particularly, for each street configuration, we have two different cases: the first case has staying period t_1 (i.e., nodes stay at PoI for t_1 period) and the second case has staying period t_2 (i.e., nodes stay at PoI for t_2 period), in which $t_1 > t_2$. After staying at the PoI, nodes start moving towards the other PoI, again by following the probabilistic route.

Figure 3.11 and Figure 3.12 show the coalition size distribution in two above movement areas in which $t_1 = 5t_2$. From these figures, we see that the coalition size gets maximum at the PoI

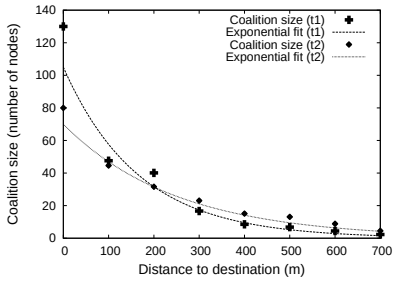


Figure 3.11: Manhattan street configuration (Here "destination" is PoI)

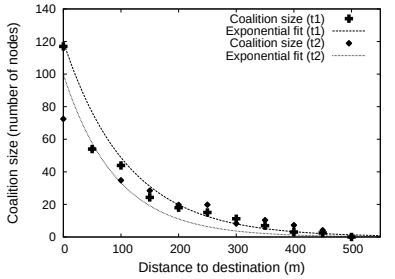


Figure 3.12: Random street configuration (Here "destination" is PoI)

and decreases when the distance to the PoI increases. Moreover, in the first case (with the staying period t_1), nodes are more clustered at the PoI since the staying period is longer than the second case (with the staying period t_2). We further fit the coalition size distribution to the exponential function in the form of $y(d) = ae^{-cd} + b$ with $d > 0, a > 0, c > 0, b \geq 0$. We see that the coalition size distributions of these two movement areas fit very well with the exponential function. Here, $y(d)$ is the coalition size, d is the distance from the PoI, and a, b, c are coefficients.

In conclusion, when the route to PoI is selected such that the shorter route has a higher probability to be selected, the nodes form coalitions in which the coalition size follows an exponential function with respect to the distance from the PoI. This result confirms the results obtained in Section 3.5 from the simulation of people movement on three real Google maps. In the next section, we briefly present how this exponential coalition size is exploited to improve content distribution.

3.7 Related Work

Since the Schelling behavior can serve as the mobility for mobile nodes in mobile P2P network, we present existing mobility models as the related work in this section. Particularly, we classify existing mobility models into two main categories: microscopic mobility models and macroscopic

mobility models. The former is about the movement behavior of individual mobile node in the network while the later focuses on the aggregated effects of individual node mobility on the entire network.

3.7.1 Microscopic mobility models

The first class of microscopic mobility model comes from random models including Random Walk mobility [23], Hybrid Random Walk mobility [56], and Random Waypoint mobility [57]. For these random mobility models, a node selects a random destination in the network space and moves from the current location to that random destiantion with a random speed. These models are obvious unrealistic since in reality people do not move between random locations. However, these models are used mostly in all current simulations due to their simplicity.

The second class of microscopic mobility model is group mobility models [19, 58, 59, 60, 61] where each group moves from one random location to another random location and the group size remains unchanged over time. This static grouping behavior is suitable to the military scenarios where group members are required to stay in a proximity to accomplish their cooperative tasks. Relying on this grouping behavior, there have been several mobility models for first responder and disaster-recovery scenarios [62, 36, 35]. In these models, nodes are assigned a certain roles such as police, fire fighters, victims, etc., where nodes with the same role move in groups and each group has one group leader. In general, the static grouping behavior of these models may work for mission-based scenarios but may not work for civilian scenarios where people move in much more free fashion. Recently, the Heterogeneous Random Walk model is presented to capture the notion of groups observed in civilian scenarios [22]. However, the model assumes an extreme clustering behavior where nodes in the big clusters group together and nodes move between big clusters do not have any neighbors. This extreme clustering behavior may not be realistic since clusters may be created at different locations of the networks.

There have been several mobility models, which aim to capture the movement of people in urban space with geographic restriction such as City Section Mobility Model [63], Manhattan, Freeway[37], Pathway [20]. However, these models do not capture the spatiotemporal movement behavior of pedestrians. There also exists mobility model for modeling obstacles in the network area and how nodes move with the existence of these obstacle [21, 38]. For this model, nodes take the obstacles into consideration and try to find a path to avoid these obstacles.

In 2003, Ray presented the Generic Mobility Model (GEMM), a tool to generate mobility traces [64], which introduces and uses four concepts: Attraction Point, Activity, Role, and Group Behavior. However, GEMM can not be used to model the pedestrian movement in urban area since GEMM assigns nodes fixed roles and nodes stay with fixed schedules of movement. Meanwhile, pedestrian movement is more dynamic and the trip plan of pedestrians may change over time.

Besides the above synthetic mobility models, researchers recently spend a considerable effort in collecting real traces of human movement to validate existing mobility models and derive new models based on the real traces [41, 42]. The first trace-based mobility model is derived from a WLAN campus at ETH [65]. In this model, the network area is divided into equal-sized squares and authors study the probability that nodes transits among these squares. Another mobility model for PDA users named Campus Way Point Mobility Model is studied at UCSD where users' appearances are associated with access points in campus buildings [66]. Hsu et al. present several trace-based mobility models at USC and University of Florida [42], including Weighted Way Point [67] and Time-variant community mobility model [34]. In Weighted Way Point model, the authors use the Markov model to represent the probability of user movements between different locations in the USC campus [67]. In Time-variant community mobility model, the authors study the repetitive movement patterns of people from the real mobility traces of campuses and vehicular networks. Ekman et al. present the Working Day Movement model to present the daily movement of average people who go to work in the morning, spend their day at work, and commute to their homes in the evening [68].

Another class of mobility models comes from social network theory [39, 40, 26]. For these mobility models, the social relationships among nodes are captured and represented where nodes tend to move toward nodes with which they have strong social connections. These models fit very well with the Delay Tolerant Network (DTN) paradigm and they can be used for DTN-based communication protocols.

3.7.2 Macroscopic mobility models

Macroscopic models present the aggregated effects of user mobility in a large scale. The first class of macroscopic models comes from traffic engineering and transportation where the teletraffic models for metropolitan, national and worldwide scale are studied for location traffic management [25, 32]. The aggregated mobility of users in urban context are also studied [28, 33, 24]. Particularly, authors

classify people into different classes based on their socioeconomic characteristics and model their aggregated spatiotemporal movement patterns. The models can be used to generate the aggregated workload of mobile users to validate communication protocols for urban areas.

In 2009, Maeda et al. present a macroscopic mobility model to represent the movement flow of pedestrians in urban area [69]. In particular, the average density of pedestrians at different locations in Osaka downtown is captured. The model represents changes of pedestrian density over time at different places of the Osaka downtown. Moreover, a large body of macroscopic mobility models comes from urban planning and transportation including flow based [70] and cellular automata [71]. Particularly, flow-based methods model the density of nodes in continuous flows. Cellular automata divides the space into cells and model the density of nodes in each cell. Recently, Ahas et al. model the short-term mobility of people in Estonia using mobile positioning data [72]. Given the location of cell phones at different times, authors confirm that the developed model describes the geography of population well.

In conclusion, the Schelling behavior of people movement lies in the middle of microscopic and macroscopic mobilities since the grouping behavior of individuals creates the uneven network density in the macroscopic level. In other words, the movement of people towards their Points of interest creates: (1) coalitions of peole sharing mutual content interest and (2) the transition of network density from sparse to dense. Schelling behavior thus can be used to study the effects of individual node's mobility on the aggregated effects of the network density. As we show in the later chapters of this thesis, Schelling behavior essentially provides a novel opportunity to improve content distribution.

3.8 Exploiting Dynamic Coalition P2P Network for Content Distribution

Nowadays, there have been numerous P2P systems that allow people to share file content (e.g., eMule, Bittorrent) or video streaming content (PPLive, Sopcast, TVUPlayer) in the Internet. These systems work beautifully by leveraging the mutual content interest of the users as the intermedia incentive for the users to collaborate and share their mutual content. For example, a person in England can share his network bandwidth and streaming content with a person in China if their computers are streaming the same football match from Sopcast. For these above P2P systems, the geographical location of users is important but not the crucial factor since users from different continents can still collaboratively share their bandwidth and mutual content. In contrast, the

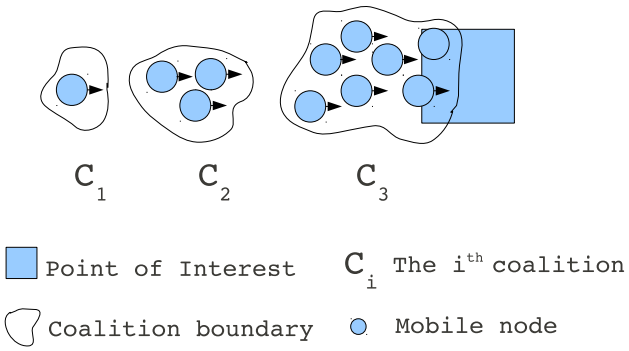


Figure 3.13: Dynamic Coalition P2P Network for one Point of Interest

geographical location of mobile users is the deciding factor for content distribution in mobile P2P networks since the network topology depends on P2P connectivity, which has a limited transmission range. In other words, in order to make the P2P sharing happen in mobile P2P networks, besides mutual content interest, the co-location of mobile users is necessary to enable the P2P connectivity. We find that the Schelling behavior represents the movement behavior of co-located mobile users who potentially share mutual content interest since they are moving towards the same PoI. The Schelling behavior thus one on hand can be used to identify the scenarios where the mobile P2P networks becomes practical, and on the other hand it can be used model the movement of mobile users for the design of content distribution protocols for these practical scenarios.

In its general form, the dynamic coalition P2P network consists of multiple Points of Interest as shown in Figure 3.2. However, in this thesis we only focus on the dynamic coalition P2P network, which has one Point of Interest as shown in Figure 3.13. In this figure, the mobile nodes form coalitions C_1, C_2, and C_3 when they are moving towards the PoI. Also, the coalition size increases at the closer distance from the PoI. For example, $|C_1| = 1$, $|C_2| = 3$ and $|C_3| = 6$ and thus $|C_1| < |C_2| < |C_3|$. As presented in Section 3.5 and Section 3.6, the coalition size increases exponentially with respect to the distance from the PoI. The exponential coalition pattern and the transition of network density in the dynamic coalition P2P network are exploited to improve content distribution as follows.

In Chapter 4 we leverage the exponential coalition size function to design COADA, a COalition-aware Adaptive content DownloAd for cellular users. With COADA, mobile nodes predict the exponential coalition pattern of the network and tune the cellular download timer adaptively to meet the file download deadline. In Chapter 5, we design iShare, an Intra-coalition data Sharing protocol to provide a lightweight and fair data exchange solution for nodes within the same coalition.

With iShare, mobile nodes download content from the cellular channel and at the same time exchange downloaded via the P2P channel. For example, nodes in the coalition C_2 of the Figure 3.13 use iShare to exchange their downloaded content. Finally, in Chapter 6, we present DENTA, a DENsity-aware daTa disseminAtion, which exploits the uneven node densities at the PoI and other areas of the dynamic coalition P2P network to efficiently disseminate data messages.

CHAPTER 4

COADA: COALITION-AWARE ADAPTIVE CONTENT DOWNLOAD PROTOCOL FOR CELLULAR USERS

4.1 Introduction

As presented in Chapter 3, the movement of mobile nodes in the dynamic coalition P2P network exhibits the Schelling behavior. As shown in Figure 4.1 , while mobile nodes move towards the same Point of Interest they form coalitions and at the closer distance from the PoI, there are more nodes moving towards the POI and thus the coalition size increases. In constrast, at the further distance there are less mobile nodes moving towards the PoI and thus the coalition size decreases. In Figure 4.1, we have $|C_1| < |C_2| < |C_3|$. Validation study in Chapter 3 shows that the coalition size formed by co-located mobile nodes follows an exponential function with respect to the distance from the Point of Interest. In this chapter, we exploit the exponential coalition size function to design an adaptive protocol named COADA, which stands for COalition-aware Adaptive content DownloAd protocol, to improve content file download of cellular users. The COADA protocol was presented in our previous papers [9, 43].

This Chapter is organized as follows. We first present the system model in Section 4.2 . Then, we present the detail of COADA design in Section 4.3. After that, we evaluate and compare the performance of COADA with an non-adaptive protocol in Section 4.4. Finally, we present the

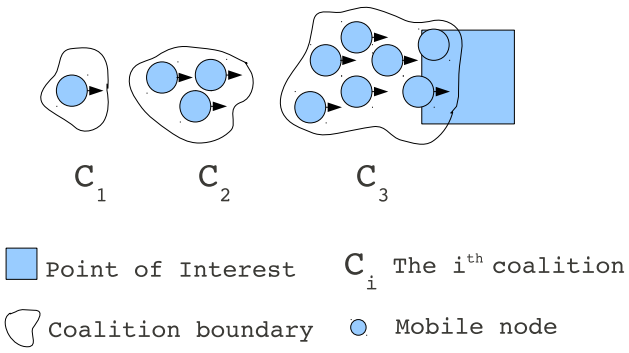


Figure 4.1: Dynamic Coalition P2P Network for one Point of Interest (Revisit)

related work in Section 4.5 and conclude the Chapter in Section 4.6.

4.2 System Model

4.2.1 Network model

We focus on a dynamic coalition P2P network as shown in Figure 4.1, where the coalition size follows the exponential function. In our network, we assume that each mobile node has a long-range connectivity interface (e.g., cellular) and short-range connectivity interface (e.g., Bluetooth, Wifi). We consider the scenario where a mobile user moves towards a PoI, requests download of files from the content server, sets the deadlines for these downloads, and uses multiple connectivity interfaces to download the file. For each file download, the user may need to set a different deadline depending on her needs. The mobile node (i.e., cell phone) then downloads some data blocks from the file server via the cellular communication, sends its own and receives other downloaded data blocks via the P2P communication to/from other nodes in the same coalition to expedite the download and meet the file download deadline. In this chapter, we use "mobile users" and "mobile node" interchangeably.

4.2.2 Data model and online codes

We use Online Codes [73] in our protocol design in which the file server divides the original file into B equal-sized *message blocks*. The content file server usually uses a big value of B so that duplication in block generation can be avoided[1]. For example, a 16MB file can be divided into 2^{13} message blocks and each is 2KB.

Given B message blocks, the server performs the following encoding procedure. The server will first create $Bk\delta$ auxiliary blocks from the B message blocks. To create these auxiliary blocks, each message block will be added by k distinct randomly-chosen auxiliary blocks and each auxiliary block is the sum of $1/\delta$ message blocks on average, where δ is a tunable parameter. In practice, the typical value of δ is 0.005 and value of k is 3 as presented in [73]. These auxiliary blocks are used to facilitate the decoding procedure in the client side. Finally, the server has a set F, in which $|F| = B + Bk\delta$ blocks.

[1]The big value of B is also important in our design of COADA since we can estimate the data availability in the P2P channel more accurately. See Section 4.3.5 for detail

Upon receiving a request for a block from a mobile node n, the server will create a check-block using the F blocks and send the check-block to n. Notice that the check-block is the transmitted data unit between mobile nodes and the server via the cellular channel, and among nodes in the P2P channel. To create a check-block, server picks a random number p (i.e., $p \leq |F|$) and selects p blocks at random from F. Then, the server calculates the sum of these p blocks (e.g., server performs XOR operations on p blocks) to obtain the check-block q. The server then sends the packet in the form of (x, q) to n, where x is a vector (or meta-data) of all indexes of p selected blocks in F.

To decode the original file, node n collects all received check-blocks (x_i, q_i), where the index i denotes the i^{th} check-block received by n. Node n then can iteratively recover original blocks from these received check-blocks until all B message blocks are decoded. As shown in [73], the B message blocks will be decoded in linear-time as long as n receives at least $(1 - \delta)F$ blocks (either from the server via cellular channel or from other nodes via P2P channel). In the next section, we use the terms "check-block" and "block" interchangeably.

4.3 COADA: Coalition-aware Adaptive Content Download Protocol

4.3.1 Design objective and COADA protocol overview

The objective of COADA is to minimize the amount of downloaded data blocks from the cellular channel and meet the file download deadline. In our protocol, each mobile node n has a cellular download timer T, which specifies how often n downloads a content block from the content server via the cellular interface (i.e., at each T interrupt, a file data block is downloaded via cellular network from the file server). Our objective is to tune T adaptively to the coalition pattern of the network so that n can minimize the cellular download while meeting the file download deadline.

In order to minimize the cellular download and meet the file download deadline, n needs to predict the coalition pattern of the network and tunes the download timer T adaptively. Since nodes choose different routes to the PoI, coalition sizes observed by different nodes may be different, and thus the actual shapes of their exponential-coalition-size functions are different. Therefore, each mobile node n will periodically obtain and keep the coalition sizes in a coalition size list. Then, n predicts the future coalition size function by fitting the coalition size list to an exponential function (i.e., curve fitting) and uses this function to estimate the data availability in P2P channel based on

Online Codes [73] (See Section 4.3.5 for calculation of data availability). Given the estimation of data availability in P2P channel, n tunes the cellular download timer T to meet the download deadline. For example, if it is estimated that there are a lot of available downloaded data in the P2P channel, n should make T longer so that it can reduce the download from the cellular channel. In contrast, if it is estimated that there are not enough available downloaded data in the P2P channel, n should make T shorter to download more check-blocks from the content server via the cellular channel to meet the file download deadline. The protocol is adaptive since n predicts the coalition size function and tunes the cellular download timer T on the move. Notice that one download timer T is for one file download. If the user has two concurrent downloads for different files, we use two different timers T. Since the number of people downloading each of the two files is different or the coalition pattern for each file is different (See the definition of coalition in Section 3.4), the download timer for each file should be adaptively tuned separately. Table 4.1 shows notations used in our protocol design, which will be presented in detail in the following sections.

Name	**Description**
F	Number check-blocks n must download to decode original file, $F = (1-\delta) \cdot B'$
T_D	Deadline at which n must finish downloading the file
T_C	Current time
T	Cellular download timer. After each T, n downloads a check-block from the content server
$g(t)$	Predicted exponential-coalition-size function, $g(t) = a \cdot e^{\lambda \cdot t} + b$
$G(t_k)$	Set of nodes in the same coalition with n at time t_k, including n. Size of $G(t_k)$ is $g(t_k)$
M	Number of check-blocks n is carrying throughout current time T_C
Δ	The protocol time period. For each period Δ, n samples one coalition size
B_p	Estimated # of check-blocks n may obtain from the P2P channel during period $[T_C, T_D]$
B_c	Estimated # of check-blocks n downloads from server during period $[T_C, T_D]$

Table 4.1: Notations used for design of COADA

4.3.2 Bootstrapping

In the COADA protocol, we divide time into equal-sized periods Δ. The length of the period Δ depends on node's speed. For example, for the network formed by cellular phones of pedestrians, Δ is longer than that of the network formed by mobile phones of drivers. This is intuitive since when nodes move faster, the coalition size changes faster and thus Δ should be smaller. Since when n moves closer to the PoI, the coalition size might get bigger, n needs to update its coalition size every period Δ. So, Δ specifies how frequently n needs to sample the coalition size.

When the mobile user starts her file download, node n is moving towards the PoI, n sets a deadline for the file download and switches both cellular and P2P channels on. Initially, n has a default cellular download timer T (with $T < \Delta$) and when T expires, n requests a check-block of the file from the content server via the cellular channel. At the same time, n exchanges downloaded data with other peers within n's P2P one-hop communication range.

To avoid wrong prediction of its coalition size function, the node n first samples several coalition sizes, one sample per period Δ, and put these coalition sizes into the list of coalition sizes. Then, for each period Δ, n performs 4 following actions: (1) predicts the future coalition size function using the list of coalition sizes, (2) downloads data from cellular channel when T expires, (3) tunes the download timer T to meet the file download deadline, and (4) exchanges data via the P2P channel. Next, we present in detail these actions of n.

4.3.3 Predicting coalition size function

As shown in Section 3.5 and Section 3.6, in our network the coalition size follows an exponential function with respect to the distance d from the node's current location to the PoI as follows:

$$y(d) = a_1 \cdot e^{-c_1 \cdot d} + b_1 \tag{4.1}$$

In Equation 4.1, $a_1 > 0, b_1 > 0, c_1 > 0$ and $y(d)$ decreases when d increases. Let t_0 be the time at which the node n starts downloading the file content from the content server via the cellular channel. Let d_0 be the distance from the location of n to the PoI at time t_0. Since the node n is moving towards the PoI, the distance d is the decreasing function of time. Here, we use $d(t)$ to denote the distance from the location of n to the PoI at time t after the time t_0 (i.e., $t > t_0$). Assuming that the movement speed of n is v (i.e., $v > 0$), for a time $t > t_0$, we have: $d(t) = d_0 - v \cdot (t - t_0)$, here $d(t) > 0$, so we have:

$$d(t) = d_0 + v \cdot t_0 - v \cdot t \tag{4.2}$$

By replacing d in Equation 4.1 with $d(t)$ in Equation 4.2, we have: $y(d(t)) = a_1 \cdot e^{-c_1 \cdot (d_0 + v \cdot t_0 - v \cdot t)} + b_1$. Or, $y(d(t)) = (a_1 \cdot e^{-c_1 \cdot (d_0 + v \cdot t_0)}) \cdot e^{(c_1 \cdot v) \cdot t} + b_1$. Let $g(t) = y(d(t)) = a \cdot e^{\lambda \cdot t} + b$. Then, we have: $a = (a_1 \cdot e^{-c_1 \cdot (d_0 + v \cdot t_0)})$, $\lambda = (c_1 \cdot v)$, and $b = b_1$.

The intuition of the above analysis is as follows. Since the file download deadline is T_D, the file downloading period is $[t_0, T_D]$. At time t, where $t_0 < t < T_D$, the node n is downloading the file

and n is also moving towards the PoI. We see that the distance d decreases if the time t increases since when t increases, n moves closer to the PoI (i.e., d decreases). Therefore, while $y(d)$ is a decreasing function with respect to d, $g(t)$ is an increasing function with respect to t. Intuitively, when the node n gets closer to the PoI, there are more other mobile nodes moving towards the same PoI, so the coalition size increases. As a result, we have $g(t) = a \cdot e^{\lambda \cdot t} + b$, in which $g(t) > 0$, and $\lambda > 0$, a, b are coefficients. The next step is to obtain the coefficients a, b, λ for the function $g(t)$ of the mobile node n.

For each period Δ, the node n obtains a new (current) coalition size and puts this new coalition size into n's list of coalition sizes. To obtain the coalition size, n broadcasts a membership message, and nodes in n's coalition respond to this membership message. n can obtain the coalition size based on the responsive messages. Then, n fits the list of coalition sizes to an exponential function in the form of $a \cdot e^{\lambda \cdot t} + b$ to obtain coefficients a, b, λ of $g(t)$. Since nodes choose different routes to the destination, coalition sizes observed by nodes may be different. However, notice that at a specific time t, nodes in the same coalition will observe the same coalition size. Node n will use its predicted exponential-coalition-size function $g(t)$ to estimate the amount of data node n can obtain from the P2P channel (to be shown in Section 4.3.5).

4.3.4 Downloading from cellular channel

The node n has a cellular download timer T. When this timer T expires, n requests new data from the server. Upon receiving the request from n, the server creates a check-block[2] and sends it to n [73]. Since Online Codes is applied at the server side and the number of message blocks B is large, the server will not create duplicate check-blocks with high probability (the detailed analysis can be found in [73]). Therefore, any check-block returned by the server is useful for n (and its peers if n sends the downloaded blocks to them) to decode the original file.

4.3.5 Tuning cellular download timer

Given the predicted coalition size function $g(t)$, the next step is to calculate the data availability in P2P channel (i.e., B_p in Table 4.1) and tune the cellular download timer T adaptively. In our protocol, when the two nodes n_1, n_2 first meet at time t, they exchange all new check-blocks via the P2P channel. For the node n_2, the blocks of n_1 are new if the blocks are not carried by n_2 at

[2] A check-block is an encoded block from the original blocks (see Section 4.2.2).

time t. After that, when they stay in one coalition, n_1 and n_2 only exchange check-blocks, which are newly downloaded during the last period Δ. Then, B_p is estimated as follows.

Since n only knows its current cellular download timer T, n assumes all nodes in n's current coalition have the similar cellular download timer T. The reason for the similar timers T is as follows: intuitively, nodes in one coalition carry the same set of check-blocks and observe similar coalition size, so their cellular download timers should be similar or at least close.

For a future time $t_k > T_C$, the coalition size $g(t_k) = a \cdot e^{\lambda \cdot t_k} + b$ can be predicted by using the function $g(t)$ of n. Further, the number of check-blocks a node in $G(t_k)$ can download via the cellular channel for the period $[t_k, t_k + \Delta]$ is $\lfloor \frac{\Delta}{T} \rfloor$. The total number of check-blocks downloaded via the cellular channel by all nodes in $G(t_k)$, including n, for the period $[t_k, t_k + \Delta]$ is $g(t_k) \cdot \lfloor \frac{\Delta}{T} \rfloor$. Notice that this calculation holds since the server will not create duplicate check-blocks with high probability by using Online Codes as presented in [73]. As a result, all new check-blocks downloaded by other nodes in $G(t_k)$ are useful for n to decode the original file. The number of periods Δ between the current time T_C and the file download deadline T_D is $\sigma = \lfloor \frac{T_D - T_C}{\Delta} \rfloor$. So, we have:

$$B_p = \sum_{i=0}^{\sigma} g(T_C + i \cdot \Delta) \lfloor \frac{\Delta}{T} \rfloor \tag{4.3}$$

Let M be the number of check-blocks carried by node n since the download starts until the current time T_C. Therefore, $M + B_p$ is the number of check-blocks n can potentially obtain by the file download deadline T_D, using the unchanged cellular download timer T for the duration $[T_C, T_D]$. To make the protocol adaptive, n updates the cellular download timer T as follows:

1. If $M + B_p < F$, then $B_c = F - (M + B_p)$. The number of check-blocks that n needs to download from the cellular channel for each period Δ is $\lceil \frac{B_c}{\sigma} \rceil$. As a result, the new cellular download timer is $T = \lfloor \frac{\sigma \cdot \Delta}{B_c} \rfloor$. If $T > \Delta$, then n sets $T = \Delta$.

2. If $M + B_p > F$, then that means n can potentially download all needed check-blocks from the P2P channel to decode the original file. However, n conservatively sets $T = T + \lfloor \frac{T}{2} \rfloor$ to prepare for a sudden change of coalition pattern in the future. If $T > \Delta$, then n sets $T = \Delta$.

Given the new value of the cellular download timer T obtained by these above steps, the node n uses the new T to download check-blocks via the cellular channel from the next Δ onward.

4.3.6 Exchanging data via P2P communication

Since the neighbor list may change over time, node n uses timestamp to avoid duplicate data exchange with its neighbors as follows. Any time n receives a block from the content server or the peer, if the block is new to n, n marks the block with n's current time. Any time n wants to send a block to n_1 via the P2P channel, n sends the oldest block b that n has not sent to n_1, based on the timestamp of b marked by n. Upon sending b to n_1, n records the timestamp of b as the oldest time n sends a block to n_1. As a result, n can use the timestamp to avoid duplicate data exchange.

4.3.7 Protocol summary and discussion

In summary, the COADA protocol divides time into equal-sized period Δ. A mobile node n periodically (for each period Δ) samples the coalition size and adds it into the list of coalition sizes. Then, n fits the list of coalition sizes to an exponential function in the form of $g(t) = ae^{\lambda t} + b$. Then, n uses $g(t)$ function to predict the future coalition sizes for the duration $[T_C, T_D]$ and calculates the number of check-blocks (i.e., B_p) n can potentially obtain from the P2P channel during this duration $[T_C, T_D]$. Next, n calculates the number of check-blocks (i.e., B_c) n needs to download from the cellular channel and tunes the cellular download timer T accordingly to meet the file download deadline. When n gets closer to the PoI, the coalition size list represents the coalition pattern of the network better. As a result, the prediction of coalition size function $g(t)$ becomes more accurate and thus the cellular download timer is estimated more accurately.

When n receives F check-blocks, n can decode the original file. Since by using Online Codes, the server does not create duplicate check-blocks with high probability, all the check-blocks that n receives from the P2P channel is useful to decode the file. Notice that since the fitting function of the coalition size is an exponential function, which is more expressive than a linear function, we believe our scheme works even better for other scenarios (with different street configurations and node densities) where the coalition size distribution exhibits a linearly increasing function. Finally, COADA is not limited to the combination of cellular and Wifi/Bluetooth communications. We believe that COADA is applicable for other communication combinations such as cellular and ZigBee.

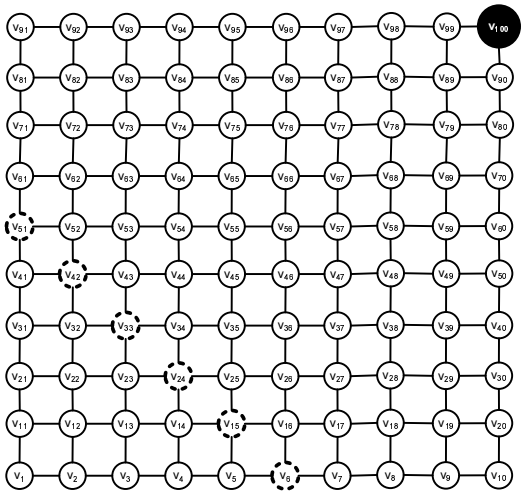


Figure 4.2: Simulation area to evaluate COADA

4.4 Evaluation

4.4.1 Settings

Parameter	**Description**
Number of nodes	[75,100,150,200]
Street seg. length (v_1, v_2)	Pedestrian:100, Car:500 (m)
Destination (i.e., PoI)	v_{100}
Download deadline	1200 (s)
Start locations	$v_6, v_{15}, v_{24}, v_{33}, v_{42}, v_{51}$
File length	[5000,10000,15000,20000](block)
Block size	2KB
Cellular download rate	2 (Mbps) [74]
P2P transmission range	Pedestrian:10, Car:75 (m)
Node speed	Pedestrian:[1.0,2.0], Car:[8.0,10.0] (m/s)
Δ	Pedestrian:50(s), Car: 40(s)

Table 4.2: Simulation settings for COADA's evaluation

We write our own simulator in C++ to evaluate our protocol. Nodes in our simulation move in a Manhattan street area with 100 intersections as shown in Figure 4.2, in which V_{100} represents the destination of the node movement (i.e., the PoI). Table 4.2 shows the details of simulation settings. For the cellular channel, we use 1xEV-DO (Evolution-Data Only) with a peak data rate of 2.4Mbps and implement the Proportional Fair Scheduler of the cellular network as presented in [74]. Initially, node n is placed at one of intersections $v_6, v_{15}, v_{24}, v_{33}, v_{42}, v_{51}$, at random. The reason we start nodes at these intersections is to give them the similar distance from v_{100}. Then, n moves towards the destination v_{100}. To avoid the wrong coalition size sampling, initially n has

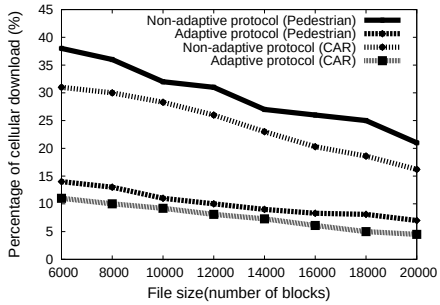


Figure 4.3: Variation of file size and its impact on % of file download from cellular network. Adaptive protocol is COADA.

the initial coalition size 1 and after the first 100(s), n starts sampling the coalition size. n starts to predict the coalition size function at 200(s). At each intersection, n selects the next intersection towards v_{100} by following the probabilistic model of route selection presented in Section 3.6.

In our simulation, we have two different types of dynamic coalition P2P networks named Pedestrian and Car networks. The former is used to evaluate the performance of our protocol for the pedestrian network with a smaller transmission range and a slower speed. This type of network corresponds to the scenario where football fans are walking towards the football stadium to attend a football match, where they download the video of match preview via the cellular channel and exchange data via the Bluetooth channel with transmission range = 10 (m). Meanwhile, nodes in Car network have a longer transmission range and a faster speed, which corresponds to the scenarios where shoppers drive towards the shopping malls and download videos of product preview via the cellular channel and exchange downloaded data via the Wifi channel. For the Car network, we use 75 (m) for the transmission range since the transmission range in practice is much shorter than the theoretical range (i.e., 250 m) due to the fading of the Wifi channel and the obstacles on streets. The deadline of the file download is set to 1200 (s) since a node n is about to arrive at v_{100} with this deadline, which is set based on the movement speed and the distance from the initial location of n towards v_{100} in Table 4.2. Notice that different mobile nodes have different movement speeds. However, they download the same file in our simulation. We run the experiments 10 times and plot the average of the two metrics: (1) the percentage of downloaded blocks via the cellular link, and (2) the peer-to-peer message overhead. These two metrics are chosen to evaluate the performance of COADA since we want to reduce the expense from cellular download and save the node energy.

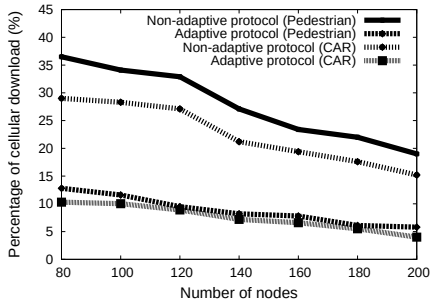


Figure 4.4: Variation of number of nodes its impact on % of file download from cellular network. Adaptive protocol is COADA.

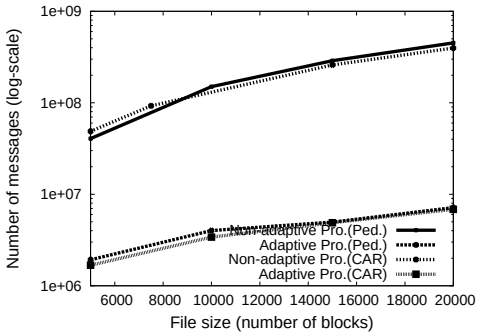


Figure 4.5: Total message overhead. Adaptive protocol is COADA.

4.4.2 Evaluation result

4.4.2.1 Comparison of COADA with Non-adaptive Protocol

We first describe the non-adaptive protocol and then compare the performance of COADA protocol with the non-adaptive protocol.

Non-adaptive protocol: The non-adaptive protocol is described as follows. A mobile node uses the cellular channel to download random message blocks from the content server and at the same time uses the P2P channel to exchange downloaded message blocks. Here, no networking coding is used to encode the file at the content server and no coalition pattern of the network is considered. Notice that the non-adaptive protocol is the current state of the art protocol, which is used to combine cellular and P2P channel to improve data download on the cell phones. Previous works in combining ad hoc and cellular interfaces to improve content distribution fall into this non-adaptive category, which does not take the exponential coalition size into account [75, 8, 50, 76, 77, 51, 78, 79, 80].

Comparison result: Figures 4.3, 4.4, and 4.5 compare the performance of COADA with the

above non-adaptive protocol. In our simulation, using the COADA protocol, all nodes meet the file download deadline. These figures show that the adaptive protocol outperforms the non-adaptive protocol in both Pedestrian and Car networks. Particularly, COADA reduces the cellular download from 20% to 25% compared to the non-adaptive scheme. In Figure 4.3, when the file size increases, COADA works better for Car network since when the network is more dynamic, nodes may meet more peers, create coalitions, and have a higher chance to obtain more missing check-blocks. Figure 4.4 shows that when the number of nodes in the network increases, both COADA and non-adaptive protocol perform better since the nodes have more peers to exchange data. Again, COADA works better for the Car network due to the increases of network dynamics. Figure 4.5 shows that compared to the non-adaptive protocol, the use of timestamp in peer to peer data exchange significantly saves the message overhead of COADA. Notice that the y-axis of this figure is in log-scale.

In conclusion, our COADA protocol reduces the cellular download and reduces message overhead significantly.

4.4.2.2 Fitting Error

Figure 4.6 shows that when the nodes get closer from the destination, COADA can predict the exponential coalition function $g(t)$ more accurately. In this figure the average normalized error is calculated as follows.

The analysis in this paragraph is for a mobile node n. Let Y denote the set of all predicted coalition size functions $g(t)$ of n for the entire simulation. Notice that for each Δ period, we have one predicted function $g(t)$ for a mobile node n. For the i^{th} Δ period, let $g_i(t)$ be the exponential coalition function predicted for the i^{th} Δ period. So, we have $Y = \{g_i(t)\}$, which is the set of all these $g(t)$ functions obtained since the starting time of the download t_0 until the file download deadline T_D. At the time of file download deadline T_D, the obtained function $g(t)$ should be the most accurate coalition size function since in our simulation the node is very close to or stays at PoI at the file download deadline. Let α be the last Δ period before the file download deadline T_D. Let $g_\alpha(t)$ be the predicted function obtained during the period α, we have $g_\alpha(t) \in Y$. For one function $g_i(t) \in Y$, we calculate the sum of absolute difference, E_i, between $g_i(t)$ and $g_\alpha(t)$. Let $E_{\max}$ denote the maximum value of E_i for all functions $g_i(t) \in Y$, or $E_{\max} = \arg\max_i(E_i)$. Then, we normalize E_i (i.e., $E_i = \frac{E_i}{E_{\max}}$), hence $0 \leq E_i \leq 1$ for all functions $g_i(t) \in Y$. As a result, for each node n we have a set of normalized sum errors of all functions $g_i(t) \in Y$.

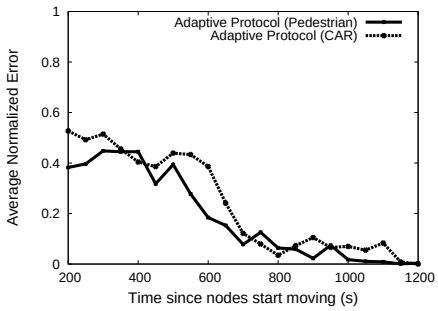


Figure 4.6: Fitting Error. The adaptive protocol is COADA.

The plot in Figure 4.6 is obtained by averaging the normalized sum errors E_i of all 1000 nodes in the simulation (for Pedestrian network or Car network). In Figure 4.6, the average normalized error decreases when time is closer to the file download deadline or node is closer to the PoI. That means, the coalition size function is predicted more accurately when node is closer to the PoI since the node obtains more "knowledge" about the coalition pattern of the network. In other words, mobile nodes obtain a more accurate exponential coalition size function when they get closer to the PoI.

4.5 Related Work

There have been previous projects, which combined cellular and peer-to-peer channels to improve downloading bandwidth of mobile users [8, 78, 16, 50, 75, 51]. These approaches are not adaptive to the network condition to reduce the download load on cellular channel, which is one of the most critical issues in mobile networking today, since the deployment of cellular infrastructure is costly. Our approach is novel since (1) we capture the change of coalition sizes from the sparse areas to the dense areas of the network, and (2) we predict the P2P data availability on the fly so that we can adapt the download plan efficiently to reduce the load on cellular infrastructure.

Network Coding becomes more and more popular in both wired and wireless P2P content distribution networks [73, 81, 82, 83]. Due to the random nature of check-blocks, network coding simplifies the protocol design and improves the performance significantly under network dynamics. In this chapter, we use Online Codes at the content server, which provides low redundant check-blocks and simplifies the estimation of data availability in the P2P channel.

Previous studies in urban planing and traffic modeling have shown that human movement depends on various socioeconomic factors such as weather [84], religious affiliation [85], and social

characteristics [86]. Recent measurement studies in [31, 54] show that people do not always choose the shortest distance route to the destinations. This motivates us to study the probabilistic model of route selection and use it to model the coalition pattern of dynamic coalition P2P network in Mobius modeling tool.

4.6 Conclusion

We exploit the transition of coalition size when the nodes move from the sparse areas to the dense areas of the dynamic coalition P2P network to design an adaptive content distribution protocol named COADA. Particularly, COADA blends cellular and P2P communications of the mobile devices and leverages the exponential-coalition-size function to improve file content download. The simulation results show that COADA considerably outperforms non-adaptive protocol and adapts well to network dynamics.

CHAPTER 5

ISHARE: INTRA-COALITION DATA SHARING PROTOCOL

5.1 Introduction

As presented in Chapter 3, the movement of mobile nodes in the dynamic coalition P2P network exhibits the Schelling behavior. As shown in Figure 5.1 , while mobile nodes move towards the same Point of Interest they form coalitions.

In this chapter, we present the iShare protocol that provides fairness in data sharing among mobile nodes of the same coalition. For example, in Figure 5.1, nodes in the coalition C_2 can use iShare protocol to download data via the cellular interface and exchange downloaded data via the P2P interface in an efficient and fair fashion. Similarly, nodes in the coalition C_3 also can use iShare protocol to expedite their data downloading process. iShare protocol was presented in our previous paper [10].

The Chapter is organized as follows. We first present the system model in Section 5.2. We then present the design of the iShare protocol in Section 5.3. In Section 5.4, we evaluate iShare protocol and compare it with other alternative protocols by mean of simulation in Network Simulator 2. Finally, we present the related work in Section 5.5 and conclude the paper in Section 5.6.

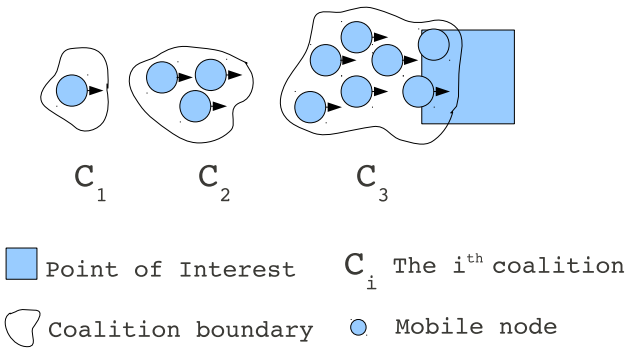


Figure 5.1: Dynamic Coalition P2P Network for one Point of Interest (Revisit)

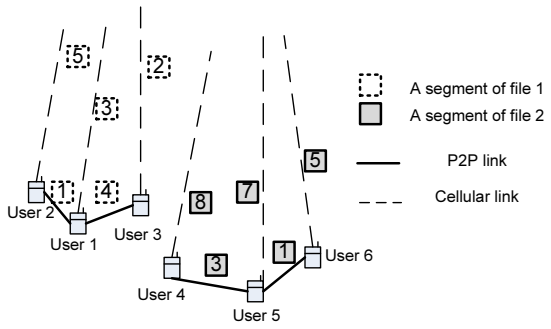


Figure 5.2: Nodes in the same coalition use iShare protocol for their downloads

5.2 System Model

We focus on a dynamic coalition P2P network, in which mobile nodes form coalition when they are moving towards the PoI as shown in Figure 5.1. We consider the scenario where mobile users download a file from the content server via the cellular interface of their cell phones. We assume that the file is similar to the Bittorrent file, which has a unique *file id* and consists of multiple equal-sized segments. Each segment also has a unique *segment id* to distinguish it from other segments of the same file.

Figure 5.2 shows the scenario where co-located mobile users (or mobile nodes in the same coalition) download the same file. We assume that each mobile node has two communication links: cellular link and P2P link (e.g., Wifi). In Figure 5.2, users 1,2,3 download file 1 from server 1 via their cellular links (likewise, users 4,5,6 download file 2 from server 2). Co-located nodes download the same file and form a mesh structure, in which each mesh member may download different segments of the file in parallel and exchange downloaded segments via P2P links. For example, node n_1 may download segment 3, node n_2 may download segment 5, and they exchange segment 1. In this paper, the terms P2P link, ad hoc link, and Wifi connection are used interchangeably.

5.3 iShare: Intra-coalition Data Sharing Protocol

5.3.1 Overview of iShare

Figure 5.3 shows the protocol state machine of an iShare node n. When the user starts his file download, his wireless device n stays in state 1 and n downloads the list of segment ids of the desired file from the cellular link. Then, n moves to state 2 and downloads one random data segment from

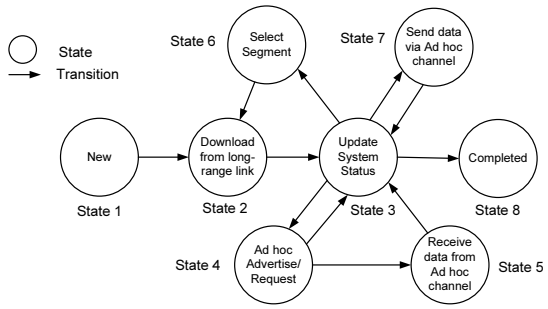


Figure 5.3: iShare node's protocol state machine

the cellular link again. After that, n stays in states 2,3,6 and continuously downloads its missing segments via the cellular link, one random missing segment at a time. At the same time, n attends an ad hoc mesh formed by nodes in the same coalition , advertises its available segments, requests missing segments, and receives segments from nodes in the same coalition via the ad hoc link (states 3,4,5). For a fair collaboration with other nodes, n applies the "tit-for-tat" incentive mechanism to send segments to neighbors via the unicast ad hoc link (state 7). Whenever a missing segment is obtained by either link, n switches to state 3 and updates the system status. When the desired file is obtained entirely, n switches to the completed state.

5.3.2 Bootstrapping iShare

This section focuses on states 1,2,3 in Figure 5.3. When a mobile user starts requesting a file, his device (i.e., the iShare node n) is in the New state. n first obtains the metadata of the file such as *file id* and the list of segment ids from the content server and downloads a random segment s of the file via the cellular link. Receiving s, n stays in state 3, where n puts s into its memory and updates its currently missing/available segments. The next question is whether n turns its ad hoc interface on to find iShare neighbors since n needs to save energy. There are two solutions. First, we can use a server from the infrastructure to track the location of nodes using GPS or Wifi access points [87] and return the closest iShare neighbor n_1 to n if n_1 is downloading the same content as n (by asking the file server). If n_1 is inside n's ad hoc range, they can collaborate via the ad hoc connection. This method is feasible because mobile devices today are equipped GPS devices and localization using Wifi access points can be applicable for indoor environment. For the second approach, after n starts its download, n can periodically check the neighborhood for iShare neighbors who are downloading the same content. This method is energy-consuming. In the

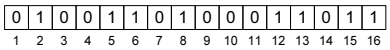


Figure 5.4: HELLO message format

future, mobile phones can use GPS IIF for an accurate localization both indoor and outdoor [88].

5.3.3 Ad hoc data exchange

After downloading the first segment from the cellular link, the iShare node n starts using both cellular and P2P links. This section focuses on states 3,4,5 in Figure 5.3.

5.3.3.1 Content advertisement and request

For an iShare node n, the ad hoc channel is used to advertise its available segments and request its missing segments. Particularly, n periodically broadcasts a HELLO message, which is in bit vector format as shown in Figure 5.4. In this figure, the index of the bit, starting from 1 to 16, is the *segment id*; for example, the 12^{th} index denotes the 12^{th} segment. The HELLO message represents the latest downloaded segments of n. Notice that the length of a HELLO message is the number of segments of the file. The HELLO message can be used as both segment advertisement and segment request, where 1 represents one downloaded segment and 0 represents a missing segment in n's memory. Thus, the HELLO message efficiently reduces ad hoc network contention.

5.3.3.2 Mesh structure and data exchange

When the node n keeps its ad hoc interface on, n attends an ad hoc mesh formed by nodes in n's coalition . The mesh has following characteristics. First, the mesh structure is formed automatically since the co-located nodes in the same coalition are within the ad hoc communication range. This incurs little construction/maintenance cost since nodes only need to keep the one-hop neighbor list [1]. Second, any two one-hop mesh neighbors can exchange data whenever they are within the communication range. This one-hop communication adapts the network dynamics and fits the tit-for-tat incentive mechanism very well (See Section 5.3.5).

[1]HELLO message is the overhead of data advertisement, not the mesh structure.

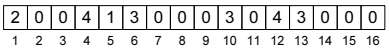


Figure 5.5: Aggregated HELLO message format

5.3.4 Downloading data from cellular link

The iShare nodes exchange downloaded data via ad hoc connections. At the same time, they continuously download segments from the cellular link. This section presents how iShare nodes download segments from cellular link and focuses on states 2,3,6 in Figure 5.3.

To reduce the download from the cellular link, iShare nodes utilize HELLO messages to download the best segments. In particular, node n decides to download its missing segment s if s is the least available segment in n's neighborhood. To do so, n aggregates all the latest HELLO messages received from its one-hop neighbors to create an aggregated HELLO message as shown in Figure 5.5. In this figure, each square can be a byte instead of a bit like the HELLO message. Notice that for each neighbor n_1 of n, n only keeps the latest HELLO message received from n_1 for the most updated available segments of n_1. In Figure 5.5, a square represents a segment s with the number of available copies of s in n's neighborhood. Node n downloads a missing segment whose number of copies is least founded in the aggregated HELLO message. If there exist more than one missing segments with equal number of available copies, n downloads one at random. For example, if n's sent HELLO message is in Figure 5.4 and n's aggregated HELLO message is in Figure 5.5, then n may download segment 3 from the cellular link since segment 3 is missing at n and n's neighbors. Whenever n finishes downloading a segment s, n is in the state 3 in Figure 5.3. Here, n inserts s into its memory and continues downloading its missing segments from the cellular link.

The random segment selection presented above parallelizes the download among mesh members. Particularly, mesh members concurrently download different segments and exchange these segments via ad hoc connections as shown in Figure 5.2. The aggregated HELLO message thus minimizes redundant downloads from the content server and reduces the load on the cellular link.

5.3.5 Tit-for-tat incentive mechanism

The mesh ad hoc structure efficiently parallelizes the download and reduces the load on the content server. However, for an efficient iShare protocol, we need to design an incentive mechanism to motivate the collaboration of nodes in the same coalition . Particularly, we focus on states 3 and 7 in Figure 5.3.

5.3.5.1 Tit-for-tat period

For two iShare nodes n_1 and n_2, tit-for-tat means if n_1 gives c segments (needed by n_2) to n_2 then n_2 will give c segments (needed by n_1) to n_1. Applying tit-for-tat, iShare nodes divide time into equal-sized periods, called tit-for-tat period (TTP). The TTP is then used as follows. Given two one-hop mesh neighbors n_1 and n_2, n_1 uses the current TTP to receive segments from n_2 so that n_1 can send n_1's segments back to n_2 in the next TTP. Also, n_1 sends segments to n_2 in the current TTP so that n_2 can send segments to n_1 in the next TTP. Notice that the length of a TTP is longer than that of the HELLO message broadcast period since nodes need to update available segments to perform tit-for-tat.

During a TTP, a node n counts the number of segments received from its neighbors. Given two neighbors n_1 and n_2, during a TTP, if n_2 sends 15 segments (needed by n_1) to n_1, then n_1 has counter $c_2 = 15$, corresponding to n_2. At the end of the TTP, n_1 is at the state 7, if n_1 has more than 15 segments that n_2 needs (known from n_2's HELLO message), n_1 only sends 15 random segments to n_2 via *unicast ad hoc connection*. If n_1 has less than 15 segments needed by n_2, n_1 sends them all to n_2. Here, the unicast connection is used to obtain the fair collaboration between neighbors and ensure a reliable ad hoc data exchange. In the next section, we present how to bootstrap and adapt the tit-for-tat under network dynamics.

5.3.5.2 Applying tit-for-tat

The tit-for-tat mechanism presented above encourages nodes in the same coalition to collaborate. However, it may not work effectively if the node neighborhood changes frequently , since a new pair of one-hop neighbors needs to start tit-for-tat from scratch. Thus, we present two techniques to bootstrap and adapt the tit-for-tat under network dynamics.

First, we turn on the promiscuous mode of the Wifi interface so that the iShare node n can potentially overhear messages, which are destined to n's neighbors in the same coalition by the above unicast communication of the tit-for-tat. By doing so, n opportunistically receives more data from the ad hoc channel. Of course, when the network is dense or congested, the overheard messages might be dropped and n misses the chance. Second, during a TTP, n broadcasts in the ad hoc channel a small number of its segments whose available copies are least in n's neighborhood.

Using the promiscuous mode and broadcast mechanism, iShare nodes in the same coalition improve the "tit" step of the tit-for-tat so that they exchange more segments in the "tat" step.

These two techniques allow the new neighbors to exchange data effectively under network dynamics, without restarting the tit-for-tat from scratch. These techniques also enable new nodes to join the coalition smoothly since they are given several segments for free. However, to exchange data with the old nodes effectively, new nodes need to download new segments via the cellular link. Otherwise, they become "iShare selfish nodes" and their downloading times might be longer as shown in Figure 5.8(a).

5.3.6 Completed state

When n finishes downloading the entire file, n switches to the completed state. Here, there are two options. If the iShare node n is rational, n leaves the mesh ad hoc network and stops all ad hoc communications. If the iShare node n is collaborative, n may stay for a certain period to support other downloading nodes.

5.4 Evaluation

5.4.1 Settings

We evaluate the performance of iShare where cellular nodes in the same coalition download a file via cellular link and at the same time they exchange downloaded data via the Wifi interface. We use Network Simulator 2 (NS2) to simulate cellular cells and mobile nodes with the settings in Table 5.1. Here, the segment size is 4KB since from our simulation we observe that smaller segment incurs longer HELLO message while bigger segment causes more ad hoc collision. A node has two interfaces: cellular link and IEEE 802.11b ad hoc link. We use RTS/CTS for unicast ad hoc communication. For the cellular technology, we use 1xEV-DO (Evolution-Data Only) with a peak data rate of 2.4Mbps. We implement the Proportional Fair Scheduler of the cellular network [74] in NS2.

We evaluate two metrics: "average downloading time" and "average number of downloaded segments". The former is the average (AVG) period for an iShare node to finish downloading a file. The latter is the average number of segments a non-iShare node (or background node) can download via *only* the cellular unicast link for a given period. In our context, the background node is the node which downloads the content from the cellular link but it does not use iShare protocol.

Field	Value/Unit
Segment size	4KB
File size	[1000...6000] segments
Node ad hoc transmission range	125(m)
Base station radius	750(m)
Mobility model	Random Way Point
Node speed (Mobility-NS2)	[1,3,5,7,11] (m/s)
Pause time (Mobility-NS2)	5 (seconds)

Table 5.1: Simulation settings for iShare's evaluation

In this section, we use the terms "coalition" and "group" interchangeably. Also, we use "iShare node" to denote the mobile nodes, which participate in a coalition and use iShare to download file content.

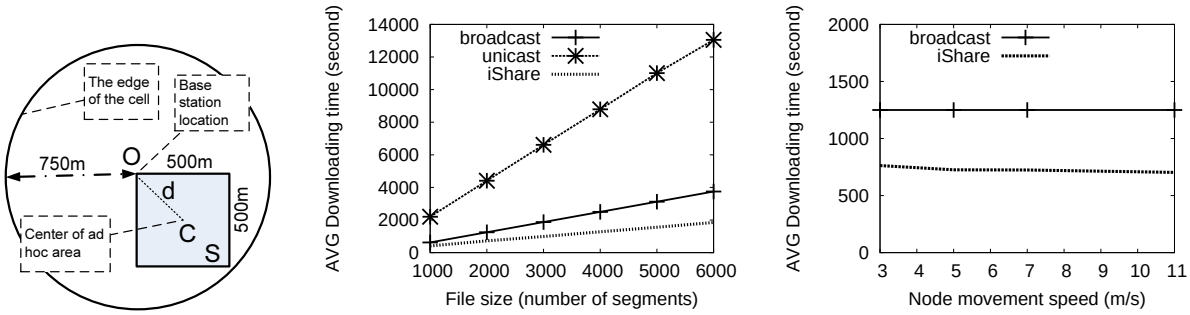
In our plots, broadcast channel means the base station broadcasts the file to all downloading nodes with a fixed rate of 208.4 Kbps and the broadcast channel takes 25% of cell bandwidth. Cellular unicast channel means downloading nodes only use cellular unicast link (without ad hoc link) to download. In our simulation, we set the HELLO broadcast period 2 seconds and TTP 7 seconds. The number of broadcast (tit) segments is 1% of the file size. The default configurations of our plots are: one coalition with 15 iShare nodes, node speed is 5 (m/s), 30 background nodes, file size is 3000 4KB-segments. Node transmission range is 125 (m) since in reality the transmission range of Wifi interface is much less than its theoretical range (e.g., 250 m). We run each simulation 10 times and plot the mean. More detail of the experiment results can be found in our technical report [89].

5.4.2 Single downloading coalition

Here, the base station is at O and nodes are generated at random in the square S as shown in Figure 5.6(a).

5.4.2.1 Impact of file size and network dynamics

Figure 5.6(b) shows that the average downloading time linearly increases when the file size increases. iShare outperforms both broadcast and unicast channels since ad hoc connection accelerates the downloads of iShare nodes. Figure 5.6(c) shows when node speed varies, iShare remains quite stable (even slightly better for more dynamic networks as nodes can exchange more data with more neighbors) and always better than broadcast channel. This confirms the robustness of the



(a) Experiment Area (b) Impact of File size (c) Network Dynamics

Figure 5.6: Performance of iShare nodes in one cellular cell

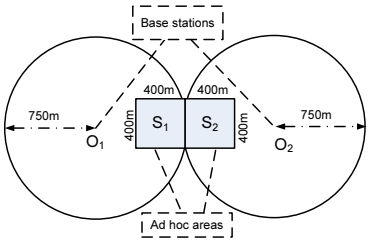


Figure 5.7: Experiment area of one coalition spanning over two cells

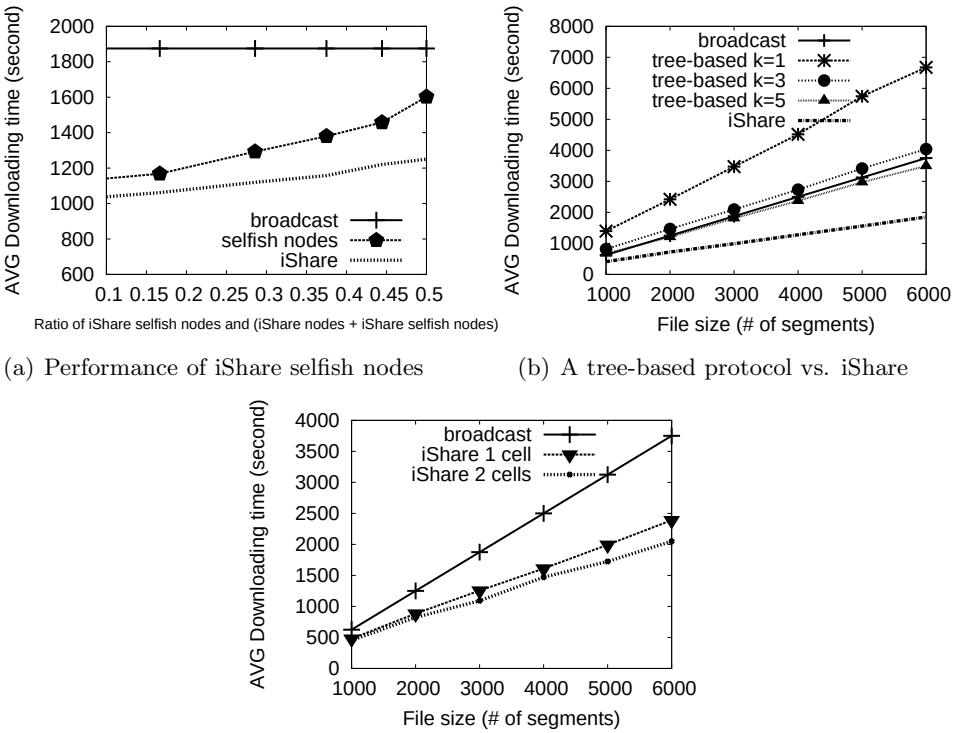
mesh structure created by iShare to network dynamics.

5.4.2.2 Performance of iShare selfish nodes

In this simulation, we have 15 iShare nodes and we vary the number of iShare selfish nodes from 1 to 15. In our context, iShare selfish nodes download data via unicast cellular links and at the same time only overhears (due to tit-for-tat) data from Wifi channel without sharing their data. Figure 5.8(a) shows that when more iShare selfish nodes exist, their downloading time increases noticeably. Meanwhile, the downloading time of iShare nodes only increases slightly due to the higher load on the cell tower resulting from higher number of selfish nodes in the cellular cell. In other words, iShare selfish nodes suffer from their own existences or iShare limits the selfishness.

5.4.2.3 Tree-based protocol vs. iShare

We compare the performance of iShare and a tree-based protocol of cooperative downloading nodes [8], which is implemented as follows. Downloading nodes elect the proxies whose distances to the base station are shortest (implying the best cellular channel downloading rate). These proxies download segments from the base station and broadcast the segments to tree members, which



(a) Performance of iShare selfish nodes

(b) A tree-based protocol vs. iShare

(c) One coalition spanning two adjacent cells

Figure 5.8: Performance of selfish nodes, comparison of iShare and tree-based protocols, and performance of a spanning coalition

replay the segments through the tree of ad hoc nodes. We make sure the tree is always connected. Here, iShare and the tree-based protocol both use the same simulation settings. Figure 5.8(b) shows that iShare protocol consistently outperforms the tree-based protocol. In this figure, k denotes number of concurrent proxies of the tree. When $k = 1$, the tree-based protocol performs much worse than cellular broadcast channel. When k increases, the tree of k roots performs noticeably better, although always worse than iShare.

5.4.2.4 Spanning Coalition

Here, we first create a group (or coalition) g_1 of 10 iShare nodes within the square S_1 in Figure 5.7. Then, we create a group g_2 of 20 iShare nodes within two squares S_1 and S_2 spanning two adjacent cells. We call g_2 a spanning group, which has a significant number of low-downloading-rate nodes (nodes are close and at the edge the cell). Nodes in g_1 and g_2 download segments from their current base stations and exchange segments via the ad hoc channel. Here, node speed is 7 m/s and each cell has 30 background users. Figure 5.8(c) shows that g_2 consistently outperforms

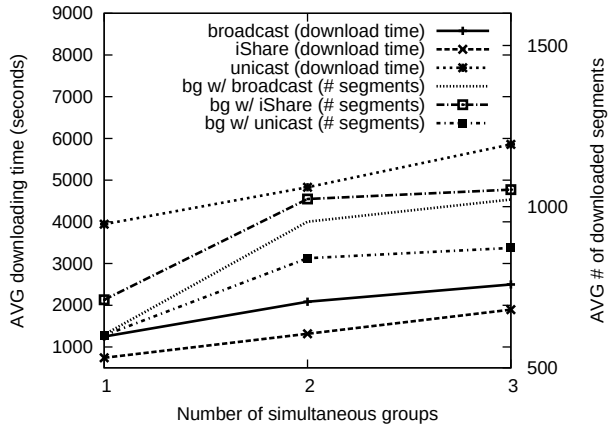


Figure 5.9: Performance of multiple coalitions in one cell

g_1 and broadcast channel. This result has several implications. First, iShare provides an efficient method to reduce downloading time, especially for low-data-rate nodes at the boundary of the cell. Second, iShare nodes can continuously obtain data via the ad hoc channel during their cellular handoff periods. Finally, iShare offers the multi-homing download for a spanning group, where group members download content from different/adjacent base stations and exchange segments via ad hoc connections to improve downloading throughput.

5.4.3 Multiple downloading coalitions

We use the settings in Table 5.1 and Figure 5.6(a). We have 3 groups (or coalitions) within S, each group has 10 nodes and downloads a different file. Thus, there is no ad hoc inter-group communication. We assume cell bandwidth for the broadcast channel is 25%, 15%, 12.5% corresponding to 1, 2, 3 simultaneous groups; or 25%, 30%, 37.5% aggregated bandwidth for the broadcast channel. Here, the file size is 3000 segments, node speed is 5 (m/s), number of background users is 30. Figure 5.9 shows that the downloading time of iShare nodes increases when more groups exists since iShare nodes suffer from a higher contention in the ad hoc channel. Moreover, due to the promiscuous mode, iShare nodes receive redundant messages from nodes in other groups, which may collide with the desired overheard messages from nodes in the same group. As a result, the performance of tit-for-tat degrades. However, iShare performs consistently better than broadcast channel. Figure 5.9 also shows background nodes download more data (the right y-axis) from cellular link if mobile nodes use iShare protocol (i.e., bg w/ iShare). This result suggests the use of multiple channels for multiple ad hoc groups in the same cellular cell.

5.5 Related Work

There have been previous works on the combination of cellular and ad hoc networks to improve downloading bandwidth of mobile users [76, 77, 16, 79, 8, 80, 90, 91, 78]. These approaches put new functionalities on the cellular telephony infrastructure such as a new scheduler, membership management, credit verification; and thus require a high cost of deployment. In particular, these approaches select the high-data-rate nodes such as proxies or super nodes to connect to the base station. They next construct and maintain trees rooted at proxies. Then, the packets are sent from the base station to the proxies and forwarded to the receivers, assuming that nodes are collaborative. Under network dynamics, maintaining these trees incurs high overhead. If the proxies leave the cell, the trees need to be reorganized. In contrast, iShare requires no changes in the cellular telephony infrastructure since network functionalities are performed by ad hoc nodes.

Another possibility to provide content to simultaneous users is using the cellular broadcast/multicast channel [92, 93, 94, 95, 96]. However, multicast/broadcast services [93, 94] are to support content to a large number of users simultaneously; thus, the number of multicast/broadcast channels is usually limited [94]. Meanwhile, in reality, people can instantly form small-scaled groups to exchange content; broadcast/multicast services thus become inefficient. Moreover, multicast/broadcast services have no feedback channel, the data is delivered with no guarantee. There also exists work combining cellular unicast with ad hoc links to improve the downloading bandwidth of multicast users by creating a tree rooted at a proxy to relay packets [97]. However, this approach suffers a high cost of tree maintenance under network dynamics.

Previous studies on wireless networks presented numerous incentives mechanisms to motivate the collaboration of mobile users such as market sharing, credit accounting, and rewarding [98, 99, 100, 101, 102]. However, these mechanisms add significant complexity to the system and applying them requires changes in the business policy of service providers. Moreover, due to the scarcity of battery, there is no immediate incentive for a user to turn on his ad hoc channel just for forwarding others' data. Our tit-for-tat mechanism is simple yet practical because they reflect the rationale of human beings: co-located mobile users are willing to collaborate if they share the mutual content interest.

5.6 Conclusion

We have developed an efficient sharing protocol (iShare) that combines different wireless interfaces on mobile nodes in the same coalition to improve content dissemination services. The results from simulation in NS2 show that iShare significantly outperforms alternative schemes based on cellular broadcast channels, cellular unicast channels, or tree-based protocols. Furthermore, the results confirmed that "tit-for-tat" mechanism succeeds in countering selfishness user behavior, adapting very well to network dynamics, and improving performance of nodes in the same coalition. Finally, the obtained results showed the multi-homing download feature for coalitions spanning over adjacent cellular cells.

CHAPTER 6

DENTA: DENSITY-AWARE DATA DISSEMINATION PROTOCOL

6.1 Introduction

As presented in Chapter 3, the movement of mobile nodes in the dynamic coalition P2P network exhibits the Schelling behavior. As shown in Figure 6.1 , while mobile nodes move towards the same Point of Interest they form coalitions. At the closer distance from the PoI, there are more nodes moving towards the PoI and thus the coalition size increases. Meanwhile, at the further distance from the PoI there are less mobile nodes moving towards the PoI and thus the coalition size decreases.

In other words, there exists a transition in network density, in which at the surrounding area of the PoI, the node density increases significantly while at the further distance, the node density decreases gradually. In this chapter, we exploit this transition of network density to improve data dissemination. Particularly, we present a protocol named DENTA, which stands for DENsity-aware daTa disseminAtion protocol, to disseminate data messages from the Wifi Access Points at the Point of Interest to the coming mobile nodes. DENTA differs from COADA and iShare since we focus on the combination of Wifi infrastucture-based connectivity and Wifi ad hoc connectivity, rather than the combination of cellular connectivity and ad hoc connectivity. DENTA protocol

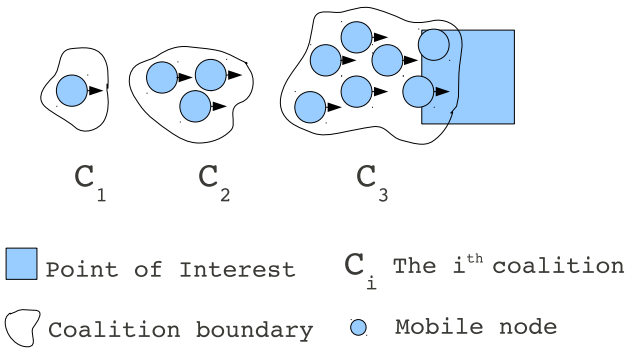


Figure 6.1: Dynamic Coalition P2P Network for one Point of Interest (Revisit)

was presented in our previous paper [11].

The Chapter is organized as follows. In Section 6.2, we present the system model. Then, we present the design of DENTA in Section 6.3. In Section 6.4 we evaluate the performance of DENTA. Finally, we present the related work in Section 6.5 and conclude the Chapter in Section 6.6.

6.2 System Model

6.2.1 Network model

We focus on a dynamic coalition P2P network where each *Point of Interest* (PoI) has a wireless base station and a server to serve requests from the mobile nodes, which are moving towards the PoI. The base station periodically broadcasts messages from the server to the surrounding area of the PoI. For example, the wireless base station at the shops can broadcast product reviews to coming customers. Also, the wireless base station at the outdoor concert can broadcast concert schedule to coming audiences. We assume that all wireless devices communicate via a common channel using IEEE 802.11 or Bluetooth. All mobile nodes have the same transmission range and distance between the two nodes within the transmission range can be estimated by various techniques [103] or using GPS. Each mobile node n can communicate with the server via the base station in the infrastructure mode and with other nodes in the same coalition via the ad hoc (or peer to peer) mode. In other words, mobile nodes moving towards the shop can exchange the product reviews via the ad hoc connectivity. Similarly, mobile nodes moving towards the outdoor concert can exchange the concert information via the ad hoc connectivity. Each PoI is characterized by its content interest. For example, the shop has the product information while the concert has the music/song information. At one time, we assume that a mobile node n moves towards only one PoI and has one content interest. although n may change its content interest and start moving towards the PoI of that interest at any time. The mobile node n has a limited amount of cache (memory) to store messages.

6.2.2 Data model

In this chapter, we consider messages including text, images, short video clips created by the PoIs. For example, messages can be advertisements and discount in commercial applications, or lecture announcements in the campus life scenario. The broadcast frequency of messages depends on their

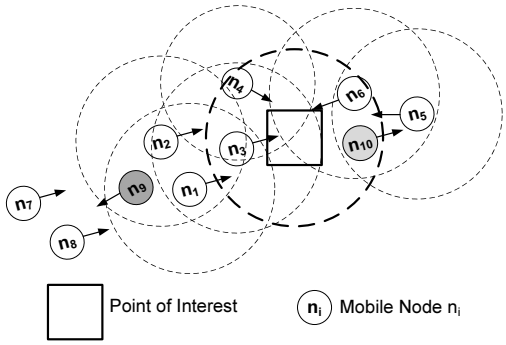


Figure 6.2: Overview of DENTA

popularity, which is determined by the PoIs. For example, for a hot sale with big discount, the shop will advertise/broadcast more frequently than other sales. Without loss of generality, we assume the popularity of messages follows a *Zipf like* distribution:

$$f(r;\theta;N) = \frac{\frac{1}{r^{\theta}}}{\sum_{i=1}^{N} \frac{1}{i^{\theta}}} \tag{6.1}$$

In Equation 6.1, N is the total number of messages created by one PoI and r is the rank of a message. When θ is equal to one, the *Zipf like* distribution becomes the classic *Zipf* distribution. We also assume the query/request of nodes follows the above *Zipf like* distribution since in reality people usually request information from more popular items [104].

6.3 DENTA: Density-aware Data Dissemination Protocol

Name	**Description**
n	Mobile node
$d(n_1, n_2)$	Distance between mobile node n_1 and mobile node n_2
m	Data message
q	Query for data message
T_m	Broadcast timer of message m
TX	Wireless transmission range of node n
β	Time-To-Live (TTL) value of message m
Γ	Message Reachability Zone
B_p	Estimated # of check-blocks n may obtain from P2P channel during period $[T_C, T_D]$
B_c	Estimated # of check-blocks n downloads from server during period $[T_C, T_D]$

Table 6.1: Notations used for design of DENTA

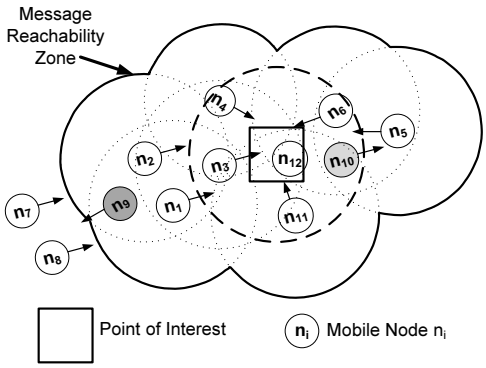


Figure 6.3: *Message Reachability Zone* of a PoI

6.3.1 Design objectives and overview of DENTA

Our design objective is to design a data dissemination protocol that (i) allows a PoI to efficiently spread its advertisements, sales, and other information to as many coming nodes of its interest as possible, and (ii) allows coming nodes to query for their interested information as soon as they comes closer to the PoI. In particular, the data dissemination scheme should maximize data access of nodes following Schelling behavior, especially for ones in close proximity of the PoI. This is intuitive because when the mobile users arrive closer to a PoI, they expect to have a more timely access to the data of their targeted PoI. Moreover, the scheme should reduce redundant transmissions to save node energy, avoid broadcast storms, and minimize transmission collisions. Notice that Table 6.1 shows the notations used for DENTA protocol.

Figure 6.2 shows the overview of our network in which a square represents a PoI and circles represent mobile nodes. The dotted circles denote wireless broadcast (transmission) ranges. In our network, a mobile node moves toward its targeted PoI (e.g. nodes n_7 and n_8) and after it arrives at its targeted PoI, it stays (nodes n_3, n_4, n_6), and then leaves (nodes n_9, n_{10}). To cover the high density of nodes surrounding a PoI, we use a push model where the PoI creates a *Message Reachability Zone* Γ by assigning each message m a Time-To-Live (β) value. This β specifies how many forwarding actions nodes in *Message Reachability Zone* perform on m. On receiving m, a node n computes a broadcast timer for m. Later, when m's timer expires and n did not hear any nodes broadcasting m, n re-broadcasts m. The timer is used to avoid broadcast storms and it is re-estimated whenever n overhears m being sent by other nodes. Outside Γ, due to the sparse network, mobile nodes follow a pull model by sending queries to their neighbors in the same coalition to save network bandwidth and energy. In the following sections, we present in detail design of DENTA.

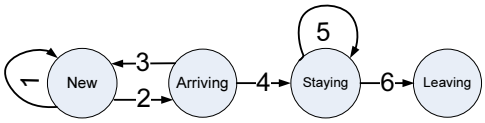


Figure 6.4: Mobile nodes can be in one of four states: *New*, *Arriving*, *Staying*, and *Leaving*.

6.3.2 Message reachability zone Γ

According to the first property of Schelling behavior in Section 3.2, density of similar nodes in the proximity of a PoI increases significantly. Thus, we use a push model to disseminate messages to this dense area. Particularly, the Wifi base station at the PoI periodically broadcasts messages and assigns a β value for each message m. Receiving m, nodes in proximity of the PoI cooperatively rebroadcast m to create a *Message Reachability Zone*. To be precise, "a *Message Reachability Zone (*Γ*) of a PoI is an area covered by broadcasts of mobile nodes arriving and staying at the PoI.*" According to this definition, the size of a Γ is not fixed. Instead, it depends on β values of the messages, transmission ranges and speeds of relaying nodes (including *Arriving* and *Staying* nodes in Section 6.3.3). Particularly, if m has a larger β or relaying nodes have larger transmission ranges, the size of Γ is bigger. In contrast, if the relaying nodes have higher speed, the size of the Γ is smaller. Notice that n only rebroadcasts m if m's β is positive and anytime m is broadcast or overheard, its β decreases by 1. Figure 6.3 shows a Γ with the solid curve boundary.

The above push model expedites messages for nodes inside the Γ. In the next section, we present how mobile nodes in the entire network cooperatively disseminate messages.

6.3.3 Mobile node states

Figure 6.4 shows a state machine where each circle is a state of a mobile node n in our network. In following sections, we discuss characteristics, the transitions, and corresponding protocols of mobile nodes at each state.

6.3.3.1 New node

A mobile node n is in the New state if n is outside the Γ of its targeted PoI. For example, in Figure 6.3, nodes n_7, n_8 are in New state since they are outside the Γ. A node n can detect the existence of the Γ by overhearing messages broadcast by nodes at the boundary of the Γ.

The main communication mode of New nodes is ad hoc because they can not directly reach

base stations. Whenever a *New* node n has a query q, n follows a pull model by broadcasting q to its neighbors in the same coalition and waiting for the answer from them. If n participates in a big coalition, q may be relayed by nodes in this coalition to nodes inside the Γ. Thus, q can be answered by nodes inside the Γ or the PoI (see Section 6.3.4). The pull model is used because according to Schelling behavior, outside the proximity of the PoI, the density of similar nodes is very low. The pull model thus can save node energy, network bandwidth, and reduce interference. A *New* node n has two transitions: 1 and 2. The former occurs when n switches to a new interest and starts moving towards the new PoI. The latter occurs when n enters the Γ of its current PoI and changes state to *Arriving*.

6.3.3.2 Arriving node

At *Arriving* state, a mobile node is inside the Γ. For instance, node n_1 in Figure 6.3 is an *Arriving* node. The role of *Arriving* nodes is to rebroadcast and relay messages to create the Γ. All *Arriving* nodes receive a rich set of information via broadcasts of similar neighborhoods. To help *New* nodes detect the boundary of Γ, in its broadcast messages, the *Arriving* node n adds a flag to mark its *Arriving* state. Using this flag, *New* nodes can distinguish the broadcast messages from other query/response messages and thus can detect the boundary of Γ.

Communication among *Arriving* nodes is ad hoc. Whenever an *Arriving* node n has a query q, n broadcasts q to its similar one-hop neighbors, which in turn can relay q to the PoI. Eventually, PoI will answer q if no nodes inside Γ can answer. This means all queries of *Arriving* nodes have a very high chance to be answered. From *Arriving* state, n can switch to *New* or *Staying* state (transitions 3 and 4 in Figure 6.4). Switching to *New* state means the mobile node changes interest while switching to *Staying* state means the mobile node enters the transmission range of the base station of its targeted PoI.

6.3.3.3 Staying node

At *Staying* state, a node n is inside the transmission range of the PoI's base station, its queries will be answered instantly by its local cache or the PoI. For example, node n_3 in Figure 6.3 is a *Staying* node. A *Staying* node has two communication modes: infrastructure and ad hoc. The former is used to communicate with the PoI and other *Staying* nodes while the latter is for communication with *Arriving* nodes. *Staying* nodes can relay queries of *Arriving* or *New* nodes to the PoI and

responses/answers from the PoI back to these nodes. *Staying* nodes also take part in creating Γ by rebroadcasting messages. Similar to *Arriving* nodes, *Staying* nodes learn a rich set of information from the base station via periodic broadcasts. In Figure 6.4, a *Staying* node can switch to *Leaving* state by transition 6. This occurs when a node changes interest or leaves the network.

6.3.3.4 Leaving node

When a *Staying* node switches interest or leaves the network (e.g. node n_9 in Figure 6.3), its state becomes *Leaving*. In the first case, a *Leaving* node of one interest becomes a *New* node of another interest. If n switches from the *Staying* state to the *Leaving* state, it resets all β values of messages in its cache. At the same time, it stops relaying queries/responses for nodes of its old interest. However, a *Leaving* node n can answer queries (via ad hoc mode) for coming nodes of its old interest.

6.3.4 Limiting query scope and cache management

When a mobile node n_1 has a query q for a message m, n_1 broadcasts q to its one hop neighbors. Whenever a node n_2 receives q, if n_1 and n_2 are in different coalitions, n_2 will not forward q. Thus, the query flooding is limited within the coalition of n_1. To avoid query broadcast storms, node n_2 only forwards the query q from n_1 once if n_2 and n_1 are in the same coalition.

Due to its limited cache size, when a node n receives message m, n keeps most popular messages in its cache and discards the overflowed messages.

6.3.5 Improving data accessibility

To improve the efficiency DENTA, we use two following techniques: broadcast storm avoidance and context-switching.

6.3.5.1 Broadcast storm avoidance

When receiving a message m, the mobile node n assigns m a broadcast timer T_m. n re-estimates T_m whenever n overhears m from nodes in the same coalition.

Broadcast Timer Estimation: When an *Arriving* or *Staying* node n receives a message m

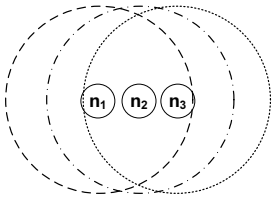


Figure 6.5: n_1 is broadcaster, n_2, n_3 are receivers. Most distant node, n_3, will be next broadcaster

from a node s, T_m is estimated by n as follows:

$$T_m = \frac{TX}{d(s,n)} \tag{6.2}$$

In Equation 6.2, TX is n's transmission range. The distance $d(s,n)$ between s and n can be estimated by various techniques [103] or using the GPS.

In Equation 6.2, if n_1 is the sender of m and n_2, n_3 are receivers, and if $d(n_1,n_2) < d(n_1,n_3)$, n_3 will assign a shorter broadcast timer to m. Then, n_3 will broadcast m prior to n_2, resulting in a larger region covered by m in the network. Figure 6.5 shows an example where n_1 is the original broadcaster of m. n_3 will be the next broadcaster because it is farther from n_1 than n_2.

Overhearing Mechanism: Besides broadcast timer, we use the overhearing mechanism to avoid broadcast storms, reduce transmission collisions and interference. In particular, whenever a node n_3 overhears or receives a message m, n_3 re-estimates T_m and decreases the value of m's β by 1. Notice that m has a β value as presented in Section 6.3.2. By re-estimating T_m, n_3 can minimize redundant broadcasts, save node energy, and reduce transmission collisions. By decreasing the value of β, node n can delete m after a certain period of save memory for other messages.

6.3.5.2 Context-switching

Essentially, when leaving the old PoI, a node n can be a New node of another interest or n leaves the network. At the moment, n has a rich set of information about its old PoI, which can be used to improve query hit for coming nodes to n's old PoI. Let us consider the first case, n switches from its old interest PoI_1 to its new interest PoI_2. When n gets closer to PoI_2, n learns more about PoI_2 through its queries. Thus, n's cache content changes gradually, with more and more messages of PoI_2 replacing messages of PoI_1. This is where the context-switching occurs. In our scheme, when n switches interest to PoI_2, although n stops broadcasting messages of PoI_1 in its cache, n still answers queries from coming nodes to PoI_1 whenever its cache has the answers. The

context-switching therefore depends on n's query for PoI_2 and n's cache replacement policy. The longer n keeps data of PoI_1, the better n can support coming nodes to PoI_1. For instance, in Figure 6.3, PoI_1 is the square, when n_9 is leaving PoI_1 for PoI_2, it meets n_7. If n_7 has a query q for a message m of PoI_1 and n_9 receives q, n_9 can answer if m still exists in n_9's cache. Likewise, n_9 may learn about its new interest PoI_2 from nodes leaving PoI_2 before n_9 enters the Γ of PoI_2. Notice that in context-switching, leaving nodes and coming nodes are considered to share partially mutual interest. Similarly, if nodes leave the network, they also can help coming nodes to improve data accessibility.

6.4 Evaluation

We implement a Java-based simulation in middleware layer to evaluate the performance of DENTA. In this section, we first describe simulation settings and how we simulate the Schelling behavior. Then, we rely on the Schelling behavior to evaluate our data dissemination scheme.

6.4.1 Simulating Schelling behavior

Table 6.2 presents the settings we use to simulate Schelling behavior. We implement a mobility model operating on a Manhattan grid model where PoIs are at the intersections of streets. A mobile node n in our simulation works as follows. Initially, n obtains a position, a speed, and a PoI, all at random. Then, n starts moving along with streets and towards its PoI. During its movement, n also can switch interest with probability p (in our simulation p=0.05 means 50 nodes out of 1000 nodes change interests per second). When arriving at its PoI, n stays for a random period from 10 to 50 seconds (notice that if this period is longer, nodes cluster more at PoI and our scheme works better). After staying at the PoI, n changes interest to a new PoI, and repeats the entire process. In this chapter, we study the *steady state* of our simulation. We expect that if we use settings in Table 6.2, at *steady state*, the simulation results in Schelling behavior. To obtain the *steady state* in our simulation, we let the entire simulation runs for 3000 steps. After that, the simulation is considered in the *steady state*.

Figure 6.6 shows node distribution in *steady state*. Each shape in this figure denotes one interest, a big shape represents a PoI and a small shape is one mobile node. When a node changes to a new interest, its shape changes accordingly. Figure 6.7 shows node distribution of one particular

Field	Value/Unit
Number of nodes	1000
Number of PoIs	6
Node speed	random [1.0,2.0](m/s)
Area of simulation	1000x1000(m^2)
Node and PoI tran. range (TX)	[50,75,100](m)
Staying period at a PoI	random [10,..,50](s)
Probability of changing interest (p)	random [0.01,0.08]

Table 6.2: Network settings for DENTA's evaluation

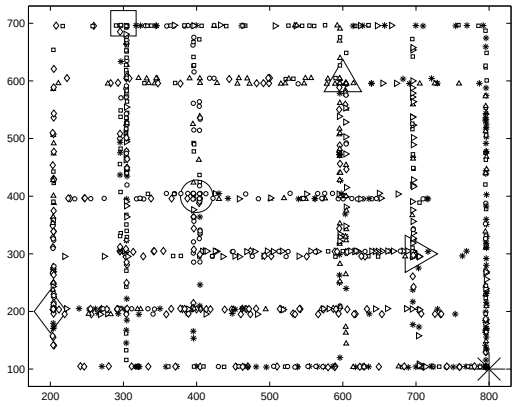


Figure 6.6: Schelling behavior exists in steady state. Big shapes are PoIs and small shapes are mobile users

interest. These two figures confirm that at *steady state* the Schelling behavior exists because (1) closer to a PoI, the density of nodes interested in this PoI increases and (2) similar nodes can group into clusters[1] due to their close proximities on the ways towards their mutual targeted PoIs. Figure 6.7 also shows that density of similar nodes gets maximized at the PoI and decreases gradually at farther distance.

Figure 6.8 shows that when a node n is closer to its PoI, its group size (or coalition size) increases. This confirms the first property of Schelling behavior. At farther distances, group size varies from 5 to 20. This confirms that mobile nodes can group into small clusters on the ways to their targeted PoIs; thus, the second property of Schelling behavior holds. Given the Schelling behavior, we simulate our data dissemination scheme with the settings in Table 6.3. Then, we use the metrics defined in Table 6.4 to evaluate our presented data dissemination scheme.

[1]In this section, we use "group", "cluster", and "coalition" interchangeably.

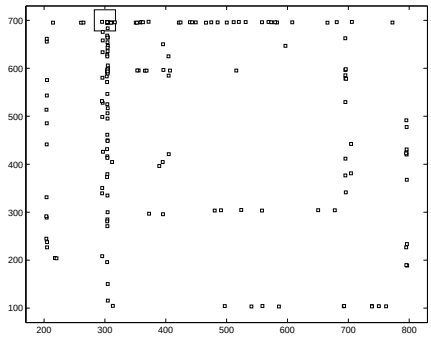


Figure 6.7: Schelling behavior for one interest

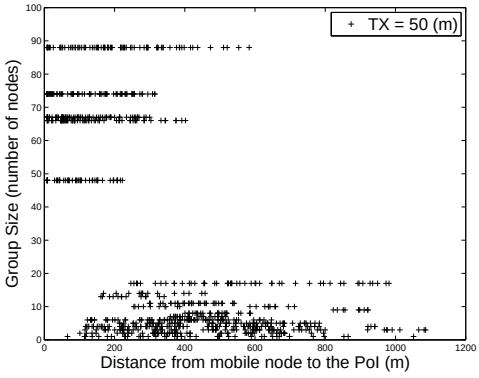
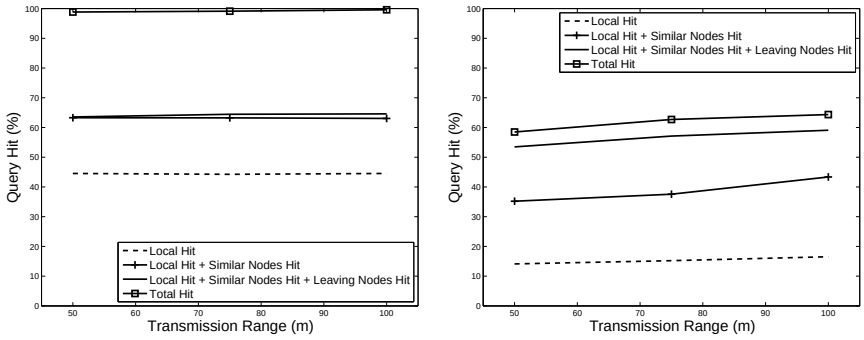
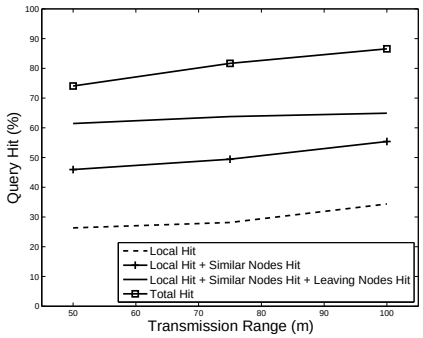


Figure 6.8: Distance to the PoI and group size. Each plus sign (+) represents a mobile node



(a) For nodes inside the Γ

(b) For nodes outside the Γ



(c) For the entire network

Figure 6.9: Message Reachability Zone and Context-switching improve "Total Hit" significantly

Field	**Value/Unit**
Number of messages created by a PoI	500
Number of messages in one broadcast of PoI	50,75,100
Node memory size M (all nodes are equal)	50,75,100
β	1,2,3,4
θ	[0.6,..,1.0]

Table 6.3: Data settings for DENTA's evaluation

6.4.2 Evaluation of data dissemination

In our simulation, we vary θ (see Equation 6.1), node transmission range TX, node memory size M, number of messages in one broadcast of a PoI, β, and p to evaluate our proposed data dissemination scheme. Notice that we only consider messages created by the PoIs.

6.4.2.1 Γ and Context-switching

In Figure 6.9, the *Message Reachability Zone* (Γ) and context-switching improve the "Total Hit". Particularly, Figure 6.9(a) shows that nodes inside the Γ have a better "Local Hit" than that in Figure 6.9(b) because they are more informed by the similar neighborhoods. Meanwhile, "Local

Name	Description/Unit(%)
Local Hit (L_1)	Query answered by local memory
Similar Nodes Hit (S_1)	Query answered by similar nodes in the neighborhoods
Leaving Nodes Hit (L_2)	Query answered by leaving nodes during their context-switchings
Server Hit (S_2)	is contributed by (i) "Staying" nodes, who directly access the PoI and (ii) "Arriving" and "New" nodes through multi-hop relays
Total Hit	$L_1 + S_1 + L_2 + S_2$
Query Miss	100 - Query Hit

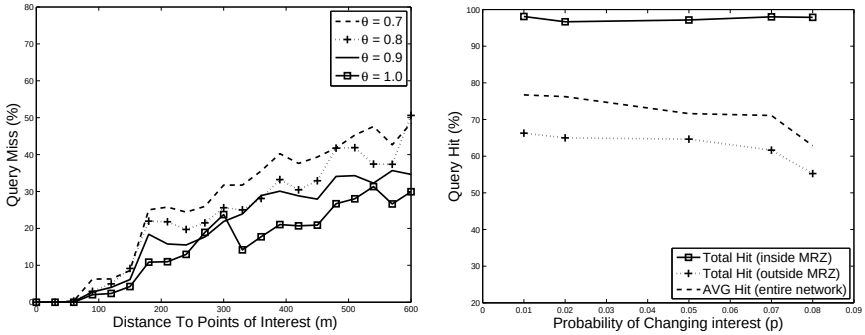
Table 6.4: Definitions of metrics for DENTA's evaluation

Hit + Similar Nodes Hit + Leaving Nodes Hit" of nodes outside the Γ is slightly less than that of nodes inside the Γ. This is because inside the Γ, nodes have similar cache content as they tend to store most popular messages. Thus, when a node fails to answer a request, it is likely its neighboring nodes will fail. In contrast, nodes outside the Γ have more diverse cache content; thus leaving nodes can contribute more to the query hit of coming nodes. In Figure 6.9(a), the two curves "Local Hit + Similar Nodes Hit + Leaving Nodes Hit" and "Local Hit + Similar Nodes Hit" look similar. This implies the context-switching is not very effective for nodes inside the Γ because they can obtain information from their neighbors or the PoIs.

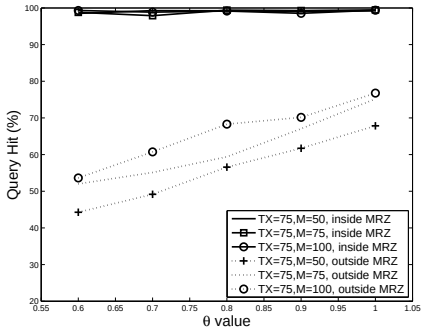
Figures 6.9(a) and 6.9(b) show the "Total Hit" of nodes inside the Γ increases up to 100% while that of nodes outside the Γ is 64%. This is because nodes inside the Γ can get answers from the dense neighborhoods and the PoIs. Meanwhile, communication of nodes outside the Γ is ad hoc in sparse areas. This also explains when TX increases, the "Total Hit" of nodes inside the Γ varies slightly. Similarly, due to the sparse network, when TX increases the "Total Hit" of nodes outside the Γ doesn't change significantly. However, for the average query hit of the entire network in Figure 6.9(c), the "Total Hit" increases about 15% (to $\sim$ 87%). This is because when TX increases, the Γ becomes larger and thus more nodes are inside the Γ, which improves average query hit ratio. In conclusion, for nodes inside the Γ, our scheme is robust with respect to TX. The context-switching concept contributes more for nodes outside the Γ and for smaller TX.

6.4.2.2 Impact of Distance and Changing Interest

Figure 6.10(a) (with TX=75,M=75,β=2) shows that at further distance to PoIs, query miss increases. This is expected because when a node n comes closer to its PoI, n observes higher data



(a) Distance sensitivity (b) Changing interest



(c) θ sensitivity ($\beta = 2$)

Figure 6.10: Sensitivity of DENTA's performance

availability provided by similar nodes in its neighborhood. Thus, n should have higher query hit ratio.

Figure 6.10(b) (with TX=75,M=75,θ = 0.8,β=2) shows when the probability of changing interest increases, query hit ratio of nodes inside the Γ is stable due to the rich set of information within the Γ. Meanwhile, the query hit ratio of nodes outside the Γ decreases gradually due to the sparse network density. This confirms the robustness of our scheme to p for nodes inside the Γ.

6.4.2.3 Impact of θ and Node Memory Size

Figure 6.10(c) shows that inside the Γ, increasing node memory size (M) does not improve the query hit much due to the high availability of data within this area. In contrast, outside the Γ, increasing M improves query hit considerably, from 5% to 10%. Figure 6.10(c) also presents impact of θ on data accessibility. In particular, when θ increases to 1, the broadcasts of PoIs and queries of nodes follow a "more" $Zipf$ distribution (Equation 6.1). Thus, they have a better "match" and provide a higher query hit ratio. Again, impact of θ on nodes inside the Γ is less significant than that on nodes outside the Γ because nodes inside the Γ are well informed by their neighborhoods.

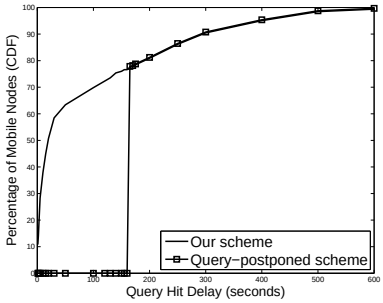


Figure 6.11: Comparison of Query Hit Delay between DENTA and Query-postponed scheme.

Meanwhile, nodes outside the Γ communicate in ad hoc mode with limited memories. This result together with results in Sections 6.4.2.1 and 6.4.2.2 confirms that inside the Γ, our presented scheme is robust to TX, θ, p and M.

6.4.2.4 Query Hit Delay and Context-switching

To further evaluate context-switching concept and automatic grouping of mobile nodes, we define the "Query Hit Delay" metric of our scheme for a node n as follows:

$$QueryHitDelay(n) = t_2 - t_1 \tag{6.3}$$

In Equation 6.3, t_1 is the time at which n switches to a new interest (and thus n starts moving towards the corresponding PoI). t_2 is the time when n's first query for the new interest gets a hit. Figure 6.11 (with TX=75,M=75,θ = 0.8,β=2) compares our scheme with a query-postponed scheme where node n holds queries since n switches to a new interest until n arrives at the new PoI. At the PoI, n obtains 100% of query hit. In our simulation, "Query Hit Delay" metric of query-postponed scheme can be estimated using distance from n's current position to its new PoI, and the speed of n. Figure 6.11 shows that our scheme obtains shorter delay (less than 170 (s)) for about 78% of nodes. Meanwhile, the query-postponed scheme has a sudden change at 170 (s) since many nodes arrive at their new PoIs after 170 (s). Particularly, in 170(s) a node can travel from 170(m) to 340(m) and it gets into transmission range of the new PoIs because a few PoI pairs in our simulation are 400(m) and one pair is 300(m) apart. This concludes that our dissemination scheme obtains better access time for all nodes, regardless their distances to their targeted PoIs.

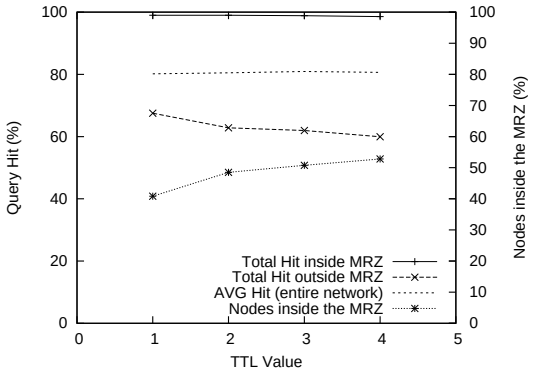


Figure 6.12: β sensitivity and number of nodes inside the Message Reachability Zone

6.4.2.5 β sensitivity and Message Overhead

In Figure 6.12, when β increases, the "Total Hit" of nodes inside the Γ is stable. In contrast, "Total Hit" of nodes outside the Γ decreases because when β increases, the radius of the Γ also increases (i.e. radius of $\Gamma \sim TX \cdot (\beta + 1)$). Meanwhile, the node density is independent of the Γ and decreases at further distance to the PoI. Thus, for a large Γ, the node density at its edge is much lower than that of a small Γ, resulting in lower query hit of nodes outside this large Γ. However, for the entire network, the average (AVG) query hit is stable ($\sim$82%) because larger β provides more nodes inside the Γ ($\sim$52% of nodes when $\beta = 4$). These nodes have better data accessibility to their targeted PoIs. This makes the AVG query hit for the entire network stable.

To further evaluate our scheme (TX=75,M=75,θ=0.8), we compare it with two other schemes: *PureFlooding* and *LimitedFlooding*. These two schemes also rely on Schelling behavior of the network but they have different data dissemination strategies. *PureFlooding* simply floods messages to the network. Queries are answered by node n itself and nodes within n's transmission range. *LimitedFlooding* uses β to limit flooding. However, it does not have broadcast timers and context-switching concepts. Similar to *PureFlooding*, queries are answered by node n and nodes within n's transmission range. All other simulation settings of three schemes are exactly the same. Figure 6.13 presents the average (AVG) query hit ratio over the entire networks and average number of messages broadcast (overhead) by one node during the simulation. Particularly, our schemes improves the query hit ratio more than 20% because mobile nodes collaboratively relay and answer queries. Our scheme also outperforms *PureFlooding* and *LimitedFlooding* in terms of average message overhead. In particular, when β increases up to 4, each mobile nodes in our scheme sends less than 200 messages while the overhead of *LimitedFlooding* and *PureFlooding*

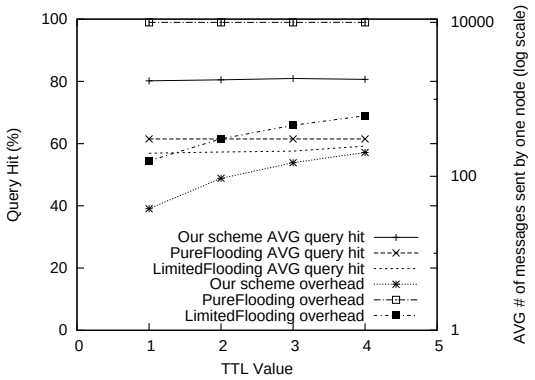


Figure 6.13: DENTA improves significantly average query hit ratio while minimizing message overhead

are about 600 and 10000, respectively. In conclusion, our proposed scheme improves significantly average query hit ratio while minimizing message overhead.

6.5 Related Work

There have been numerous studies on data dissemination in mobile P2P networks [1, 2, 4, 105, 5, 3, 6, 7]. In this section, we present representative data dissemination schemes to highlight the difference between our proposed data dissemination scheme and the existing works.

The first approach is broadcast-based data dissemination [105, 3, 12, 5, 6], which tries to adapt the dynamic and unstable nature of wireless networks. The broadcast, therefore, is the intuitive way to disseminate messages. However, blind broadcast causes broadcast storms and hurts network bandwidth. To avoid this, numerous methods have been proposed [106, 107, 108], in which nodes broadcast messages and tune broadcast rate adaptively according to network condition. Although the broadcast is controlled, these schemes may still create broadcast storms in dense networks.

The second approach is for intermittently connected wireless networks [15, 18]. These schemes leverage the store-carry-forward paradigm to improve data delivery. They essentially work for sparse networks but may not work for denser networks where the quality of dissemination is expected. For example, deadline or coverage of data delivery may not be guaranteed with these schemes.

The third approach is topic-based (interest-based) data dissemination [1, 109, 18], where co-located mobile users only exchange information if they share mutual interest on some topic. However, these schemes are only for ad hoc networks and don't exploit the big picture of the entire network where network density changes according to distance to PoIs as shown in Schelling behav-

ior. Understanding the big picture of the network and the grouping behavior of users is the key to design an efficient data dissemination scheme.

Finally, geocasting is also a related data dissemination technique, where the broadcast is determined by the physical location of mobile nodes [110, 111]. Whereas our scheme uses the similar interests to expedite data dissemination.

6.6 Conclusion

In this chapter, we present an efficient and lightweight data dissemination scheme, which exploits Schelling behavior to provide timely data accessibility, especially for nodes inside the *Message Reachability Zone* (Γ). Relying on the first property of Schelling's model, we design a push model by allowing nodes inside the Γ to collaboratively rebroadcast messages. In order to avoid broadcast storms, save node energy and reduce transmission collisions, we assign each message m a broadcast timer and apply overhearing mechanism to re-estimate this timer. To exploit the second property, we apply a pull model, in which nodes outside the Γ broadcast queries to their similar neighbors and queries/responses are automatically limited within the similar nodes, with no group management protocol requirement. To further improve data accessibility, leaving and coming nodes collaborate to answer queries. Our simulation results show that the proposed data dissemination scheme improves data accessibility significantly while minimizing network overhead.

CHAPTER 7

UIM: A JOINT WIFI/BLUETOOTH SCANNING SYSTEM FOR CHARACTERIZING REPETITIVE BEHAVIOR OF PEOPLE MOVEMENT

7.1 Introduction

It is believed that people movement exhibits a high degree of repetition, in which people usually visit regular places and make regular social contacts for their daily activities [49]. The repetitive behavior of people movement can be found in numerous scenarios such as university campuses, working places, etc. Particularly, in the university campuses, students have classes in certain weekdays and professors have fixed meeting schedules. Similiarly, workers perform their daily routines during the day time at their organizations before coming back home every evening. In this part of the thesis, we focus on the design of efficient content distribution protocols, which exploit the repetitive pattern of people movement to improve content distribution in mobile P2P networks.

Understanding the correct movement of mobile users is crucial to the design of efficient data dissemination protocols and to the network resource planning for Infrastructure-based wireless networks, Mobile Ad hoc Networks (MANET), and Delay Tolerant Networks (DTN). As a result, collecting the real movement trace of mobile users has drawn significant attention and effort from research community. However, obtaining an accurate human movement trace has remained challenging due to the lack of (1) the portable device that the experiment participants can carry for a prolonged experiment period, (2) a light-weight, power-efficient scanning protocol that can capture the movement trace and conserve the battery, (3) a device that can be programmed and debugged to capture both location information and social contacts.

In this chapter, we first present a new methodology of collecting people movement by using the cellular phone. Then, we apply our methodology to implement a scanning system named UIM[1] that collects MAC addresses of Wifi access points and Bluetooth-enabled devices in the proximity of the experiment phones. The UIM system is deployed on 123 Google Android phones carried by

[1] UIM stands for University of Illinois Movement

students, staff, and faculties in the University of Illinois campus from March to August 2010. The collected Wifi MACs are used to infer locations while the collected Blueooth MACs are used to infer social contact. To the best of our knowledge, the UIM scanning system is the first to collect both social contact and location information in one people movement trace. The UIM system was presented in our previous paper [44]. In summary, this Chapter has the following contributions:

1. We present a novel methodology to collect both social contact and location information of the people movement by using cellular phones.

2. We present the detailed design and implementation of the UIM scanning system, which collects both MAC addresses of Wifi access point and Bluetooth devices in the proximity of the experiment phone. We discuss the design decision on how to conserve phone battery and how the system components interact with each other to obtain the performance reliability.

3. We characterize the contact patterns, regularity of contacts, and regularity of location visits found in the joint Wifi/Bluetooth trace collected by the UIM system. We find that people movement exhibits a high degree of regularity. These regular patterns will be used in Chapter 8 and in Chapter 9 to predict people movement and design a new content distribution scheme.

In this chapter, we first present a new methodology of collecting the movement trace on the Google Android phone and the design of the UIM system in Section 7.2. Then, we present the detail of our collected data set in Section 7.3 and the sensitivity analysis of scanning frequency on the collected data in Section 7.4. In Section 7.5, we characterize the repetitive pattern of people movement found in the collected joint Wifi/Bluetooth trace. Finally, we present the related work in Section 7.6 and conclude the Chapter in Section 7.7.

Notice that, we present other measurement results derived from our collected data set in the appendix Chapter B. We also present the Hybrid Epidemic routing protocol, which is evaluted over the real collected Wifi and Bluetooth traces in the appendix Chapter C.

7.2 UIM: Joint Bluetooth/Wifi Scanning System

In this section, we introduce a new methodology of collecting people movement trace. Then, we present the implementation of the UIM system and discuss the UIM system design decision.

7.2.1 New methodology of collecting people movement trace

In designing a movement trace scanning system, we must ensure: (1) the experiment devices must be always carried by the experiment participants when they are moving, and (2) the experiment devices must have a long enough battery life for the prolonged experiment period. Except the Reality data set [112], which was collected by cell phones, all previous works [113, 114, 115, 116, 117, 118] used either iMote device or PMTR platform-based device to collect the movement trace of people. Due to the limited battery of the iMote device and PMTR platform-based device, these above collected traces included no more than 20 days of movement trace. We thus believe using the cellular phones to collect people movement is the most suitable solution since: (1) people always carry their cellular phones with them, and (2) people recharge the phone for their uses when needed. However, deploying a movement trace scanning system on cellular phone is challenging since the scanning system must be transparent to the cellular users and robust to the energy-limited environment of the cellular phone. In this section, we present the methodology of collecting people movement trace by using mobile phone. Particularly, we present the objectives that the movement trace scanning system on the cell phone must obtain:

1. The scanning system should ensure that user movement is scanned accurately in which both location information and social contact information of people movement are collected. These two pieces of imformation are important since they can be used to infer the movement pattern of the user (location) and the social interaction among people

2. The scanning system should conserve battery usage for a prolonged experiment period. Normally, the user uses the cell phone for phone calls and message exchange. So, if the system scans the movement trace so frequently and that makes the user charge the phone so frequently, she will not use the experiment phone (and not carry it with her) and thus no trace is collected. Currently, an iPhone or a Google Android phone can only last for about 10 hours if the Wifi interface is on. Therefore, designing the scanning system in an energy-efficient manner so that the phone can both scan movement trace and satisfy other usages of the user becomes critical.

3. The scanning system should incur no interference to the other applications running on the experiment phone. Currently, the user can install and run different apps on the phone. If we want the user to use the experiment phone as her daily phone (and thus she will carry the

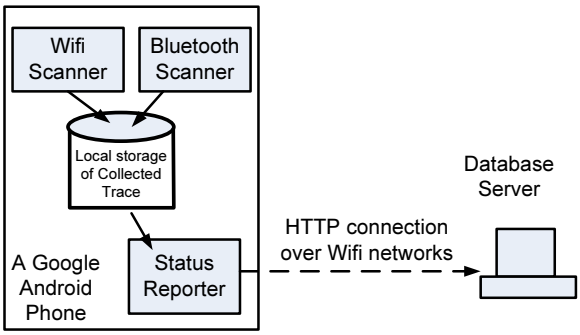


Figure 7.1: UIM System Architecture

phone, recharge it, and we have a good collected movement trace), the scanning system must: (1) start it by itself when the phone starts, (2) run in the background and does not display any messages on the GUI interface to interrupt the user (transparent), (3) keep running even if other applications halt or crash (robust).

In the next section, we apply this methodology to design and implement a new movement trace scanning system on Google Android phones.

7.2.2 UIM system: design and implementation

As shown in Figure 7.1, UIM has two main components: the database server and the Google Android phone. The former hosts a relational database management system, which accepts and stores the scanning status updates from the experiment phones. The latter has three subcomponents: the Status Reporter, the Bluetooth scanner, and the Wifi scanner. The Bluetooth scanner collects the Bluetooth MAC addresses of Bluetooth-enabled devices in the experiment phone's proximity while the Wifi scanner collects the MAC addresses of Wifi Access Points in the proximity of the experiment phones. The collected movement trace, including ad hoc trace and Wifi trace, is stored at the local disk of the phone. The *Status Reporter* updates the scanning status of the phone (e.g., how the scanning works, how many trace files have been created) to the server via the HTTP connection when the Wifi connectivity is available. Due to the battery constraint, *Status Reporter* is set as an optional feature of UIM. We find that *Status Reporter* works smoothly if enabled. In the next sections, we present in detailed the design and implementation of Bluetooth scanner and Wifi scanner.

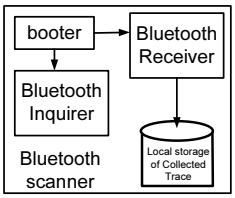


Figure 7.2: Bluetooth scanner

7.2.2.1 Bluetooth scanner

The *Bluetooth scanner* periodically (e.g., every 60 seconds) scans the Bluetooth-enabled devices in the phone's proximity [2]. The scanned results include the MAC addresses of the Bluetooth-enabled devices and the corresponding scanning time stamps. In this chapter, we use δ_B to denote the scanning period of the Bluetooth scanner (e.g., $\delta_B = 60(s)$) and "ad hoc MAC" to denote the scanned MAC addresses of the scanned devices. Notice that the ad hoc MAC can be an experiment phone or a scanned device, which is not in the set of experiment phones. So, we use "external ad hoc MAC" to denote a scanned device, which is not in the set of experiment phones. The trace collected by the Bluetooth scanner is called the "ad hoc trace" or "BT trace". We set the Bluetooth scanning period $\delta_B = 60(s)$ to conserve the phone battery. Notice that UIM makes the experiment phones discoverable in the Bluetooth channel so that an experiment phone can scan other experiment phones in its proximity.

The Bluetooth scanner is shown in Figure 7.2 with three sub components: booter, Bluetooth inquirer, and Bluetooth receiver. We implement Bluetooth scanner as a background service, anytime the phone restarts, the phone operating system will trigger the booter, which starts the Bluetooth inquirer and Bluetooth receiver. With the booter, UIM obtains the robustness and reliability to run for a prolonged experiment since anytime the phone restarts, the scanner can start its scanning work automatically. The inquirer and receiver work in an asynchronous fashion in which the inquirer uses a request timer to periodically (i.e., every 60(s)) generate a Bluetooth scanning request and send to the system. After sending the request, the inquirer makes the phone discoverable by other experiment phones, goes to sleep, and wakes up for the next request when the timer expires. To conserve battery, we configure the inquirer so that it only generates scanning request from 7AM of a day to 1AM of the next day. During the period [1AM,7AM] of a particular day, the inquirer sleeps. As a result, we can collect most of people movement while saving phone battery. The receiver, on the other hand, is only triggered to work whenever a Bluetooth scanned result is

[2] In this Chapter we use "participant", "phone", "user", and "experiment phone" interchangeably.

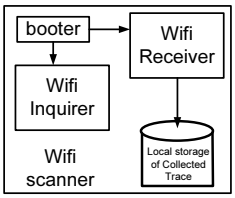


Figure 7.3: Wifi scanner

returned from the phone operating system. Upon receiving the scanned result, the receiver writes the result to the log file with the timestamp and then goes to sleep. Moreover, whenever the phone user triggers the Bluetooth scanning from the phone's GUI, the phone operating system performs a Bluetooth scan and returns result to the Bluetooth receiver. As a result, the Bluetooth receiver *opportunistically* receives Bluetooth scanned results even when the inquirer does not trigger the Bluetooth scan. With this design methodology, the Bluetooth scanner obtains robustness and conserve phone battery.

7.2.2.2 Wifi scanner

The *Wifi scanner* periodically (e.g., every 30 minutes) scans the Wifi access points in the phone's proximity. The scanned results include the MAC addresses of the Wifi access points and the corresponding scanning time stamps. In this chapter, we use δ_W to denote the scanning period of the Wifi scanner (e.g., $\delta_W = 30(minutes)$) and "Wifi MAC" to denote the scanned MAC addresses of the Wifi access points. The trace collected by the Wifi scanner is called the "Wifi trace". There are two reasons we set $\delta_W = 30(min)$. First, in the campus environment, people usually do not move too far and stay in the offices or buildings for a long time period (e.g., a class session is usually 50 minutes). Second, performing Wifi scan on the cell phone is energy-consuming. Notice that UIM does not scan for absolute positions of experiment phones using GPS since (1) GPS scanning is highly energy-consuming and (2) people in university campus usually stay indoor and the phones usually obtain inaccurate coordinates when it is indoor.

The Wifi scanner is shown in Figure 7.3 with three components: booter, Wifi inquirer, and Wifi receiver. The design of Wifi scanner is similar to the Bluetooth scanner in which the Wifi inquirer and Wifi receiver are decoupled. The Wifi scanner automatically boots itself and starts Wifi scanning when the phone boots. The Wifi inquirer periodically (and automatically) triggers the Wifi scan request so that the Wifi receiver starts the scan in response. If the phone user uses

the Wifi connectivity, the Android OS will issue a Wifi scan. In this case, our Wifi receiver can *opportunistically* receive more Wifi scanned results. However, since the Wifi scan is highly energy consuming, the period between two consecutive inquires of our Wifi inquirer is set to 30 minutes (or the Wifi scan period is 30 minutes).

There are two reasons the Wifi scanning period is set to $30(min)$. First, in the campus environment, people usually stay in the offices or buildings for a long time period (e.g., a class session is usually 50 minutes). Second, performing Wifi scan on the cell phone is energy-consuming. The scanned results of the Wifi scanner include the MAC addresses of the Wifi access points and the corresponding scanning time stamps. In this chapter, we use "Wifi MAC" to denote the scanned MAC addresses of the Wifi access points. The trace collected by the Wifi scanner is called the "Wifi trace".

7.2.3 Discussion of UIM design decision

UIM system achieves the *design objectives* in Section 7.2.1 as follows. For the *first objective*, UIM provides more accurate movement trace than previously collected traces [113, 112, 114, 115, 116, 117, 118]. Particularly, the Bluetooth scanner of UIM scans every 60 (s), which is the highest scanning frequency compared to previous works in the University Campus category as presented in Table 7.3. So, UIM can obtain more Bluetooth MACs. Moreover, since UIM has the Wifi scanner, the Wifi MACs obtained by the Wifi scanner can be used to represent location (see Section 8.3) infer the movement patterns of the participants. For the *second objective*, since we tune the scanning frequency of Bluetooth and Wifi scanners carefully, the phones can be used by participants as daily cell phones for about 2 days before running out of battery. Moreover, since most of our participants put their cell phone simcards into our experiment phones, participants recharge the phones and keep the phones on. Furthermore, we configure our scanners to run from 06:00 to 23:59 everyday, instead of running the entire day. After 23:59, the scanners pause to conserve battery and wake up at the 06:00 the following day. Using this scanning configuration, we obtain most of the movement activity of the phone carriers and save the battery for the phone usage. For the *third objective*, we implement the UIM system so that UIM runs as a background process of the phone and is transparent to the phone user. We carefully implement and configure UIM so that it does not interfere with other services and applications of the phone users. Moreover, anytime the user turns on the phone, the scanners start and scan the devices in proximity periodically. This ensures that

UIM remains robust to the usage of the phone and can starts running itself whenever the phone is on.

Another possibility to improve our scanners is to use the motion sensor equipped with the experiment phones to detect the movement of participants and activate the scan only when people is moving. This requires an accurate motion detection technique [119], which is not the focus of this thesis. On the other hand, if the movement detection is not accurate enough, we may have false positives that result in unnecessary scans and consumes more phone energy.

7.3 Collected Data Set

Overall Characteristics			
Name of Data Set	D_1	D_2	D_3
Number of Internal Devices (participants)	28	79	16
Experiment Period	03/01-03/20	04/08-05/15	05/24-08/16
Number of Days	19	38	85
Bluetooth Scanning Period (sec, δ_B)	60	60	60
Wifi Scanning Period (min, δ_W)	30	20	30
Number of External Scanned BT MACs	8508	17080	7360
Number of Scanned Wifi AP MACs	7004	29324	6822
Participant Information			
Number of CS faculties	2	2	0
Number of CS staff	1	1	0
Number of CS grads	14	30	12
Number of CS undergrads	8	43	1
Number of ECE grads	2	2	2
Number of ABE grad	1	1	1

Table 7.1: Overall characteristics of UIM trace

Table 7.1 shows the overall statistics of the UIM trace. Basically, from March 2010 to August 2010, we have conducted three rounds of experiment with 102 unique participants in total. The total number of participations is 123; however, some participants participated in more than one round of experiment. Notice that in Table 7.2 and Table 7.3, the numbers of UIM trace are taken only from the data set D_1 in Table 7.1.

The participants include faculties, staff, grads, and undergrads as shown in Table 7.1. The CS faculties, staff, grads usually work inside our department building named Siebel Center. Meanwhile, CS undergrads may take classes in different buildings throughout the university campus. ECE and

ABE (e.g., Department of Agricultural and Biological Engineering) grads stay in different buildings from Siebel Center. In this table, for two phones p_1 and p_2, we say that p_1 and p_2 have an "internal contact" if p_1 sees p_2 in its Bluetooth scanned results or vice versa. For a phone p and an external ad hoc MAC address e, we say that p and e have an "external contact" if p sees e in its Bluetooth scanned results.

	PMTR[118]	**Intel**[114]	**Cam. City**[116]	**Infocom**[115]	**UIM**
Environment	**Workplace**	**Corp.**	**City**	**Conf.**	**UIUC**
Duration (day)	19	3	10	3	19
# of Devices	49	8	36	41	28
δ_B(second)	1	120	600	120	60
Ad hoc Trace	Yes	Yes	Yes	Yes	Yes
Location Trace	No	No	No	No	Yes
Device Type	PMTR	iMote	iMote	iMote	Phone
# of In- contact	11895	1091	8545	22459	30385
# of Ex- device	N/A	92	3586	197	8508
# of Ex- contact	N/A	1173	10469	5791	82091

Table 7.2: Comparison of UIM trace with other traces collected in City, Workplace, Corporation

	Cam. U.[114]	**MIT**[112]	**Toron.**[117]	**UCSD**[120]	**Dart.**[121]	**UIM**
Environment	**University Campus**					
Duration (day)	5	246	16	77	114	19
# of Devices	12	97	23	273	6648	28
δ_B(second)	120	300	120	N/A	N/A	60
Ad hoc Trace	Yes	Yes	Yes	No	No	Yes
Location Trace	No	CellID	No	AP	AP	AP
Device Type	iMote	Phone	PDA	PDA	Laptop	Phone
# of In- contact	4229	54667	2802	195364	4058284	30385
# of Ex- device	159	N/A	N/A	N/A	N/A	8508
# of Ex- contact	2507	N/A	N/A	N/A	N/A	82091

Table 7.3: Comparison of UIM trace with traces collected in other University campuses

Table 7.2 and Table 7.3 compare the overall characteristics of UIM trace and other previously collected Bluetooth/Wifi traces. To the best of our knowledge, we are the first to collect the social contact and location information in a comprehensive movement trace. Moreover, UIM trace falls into the University Campus category trace and has the highest scanning frequency of the Bluetooth scanner compared to other traces collected in University campuses. Thus, we obtain

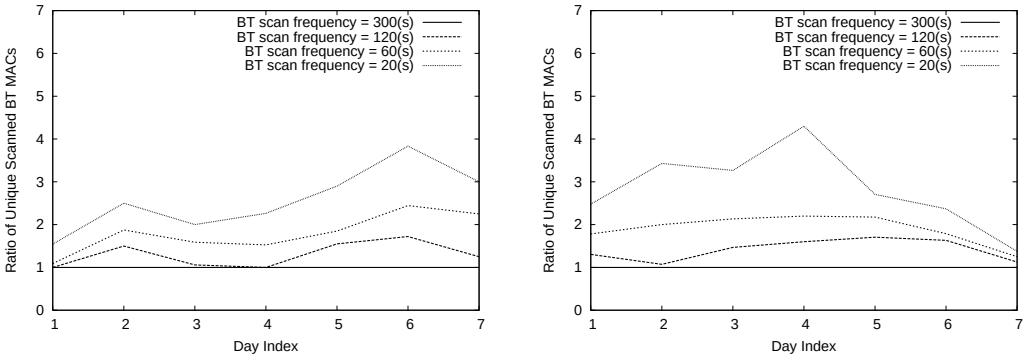
more detailed and accurate social contacts of people movement (see Section B.2). For all traces, only MIT/Reality [112] can provide the location information by inferring the cellular base station ID associated with the experiment phones. However, the cellular base station transmission range varies from meters (e.g., 500 (m)) to kilometers (e.g., 30 (km)) and thus can not be used to represent the fine granularity of the physical location. With the UIM system, we collect the MAC addresses of the Wifi access points in the proximity of the phone and use the Wifi MACs to infer the physical location. This inferred location information provides a finer granularity than that of cellular base station ID.

Since we have three data sets D_1, D_2, D_3, in our later sections in this thesis, we will refer to the specific data set in our analysis and protocol evaluation.

7.4 Sensitivity Analysis of Scanning Frequency

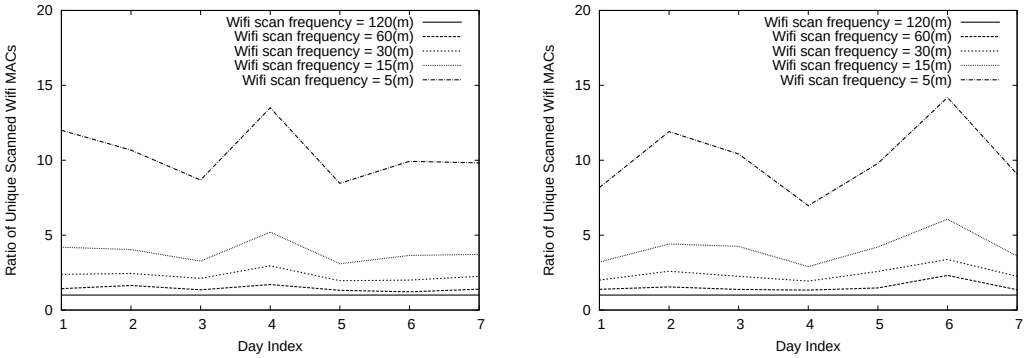
This section present the sensitivity analysis of scanning frequency on the collected data set. Our goal is to understand the tradeoff between scanning frequency (or energy consumption for the scan) and the amount of collected data. To this end, we first have two participants carrying two experiment phones for a week, in which we set the scanning frequency of Blueotooth scanner to 20 seconds and frequency of Wifi scanner to 5 (minutes). Let D_B and D_W denote the collected Bluetooth and Wifi data from these two participants, respectively. Notice that as of October 2010, the highest scanning frequency of the Bluetooth scanner is 12 seconds due to the hardware constraint of the Google Android phones. Then, we use D_B and D_W to create data sets for lower scanning frequency as follows. For example, we can create a Bluetooth data set D_{B1} with the scan frequency of 40 seconds by taking every other Bluetooth scanned record of the data set D_B. Similar technique can be applied for Wifi data set.

Figure 7.4 shows the impact of scanning frequency on Bluetooth collected data set. In order to obtaint this figure, for each user, we create the Bluetooth data sets for the scanning frequencies of 20(s), 60(s), 120(s), and 300(s). Then, we use the scan frequency 300(s) as the base line. For each frequency, we count the number of unique BT scanned devices for each day and take the ratio of this number over the corresponding number obtained by the scan frequency 300(s) of the same day. The figure shows that with the highest frequency of 20(s), the BT scanner can obtain up to 4 times the number of scanned devices compared to the base line. With our current scanning frequency of 60(s) (used for data in this thesis), we obtain 2 times the number of scanned devices compared to



(a) User 1: Bluetooth Data Set (b) User 2: Bluetooth Data Set

Figure 7.4: Impact of Scanning Frequency on Bluetooth Collected Data Set



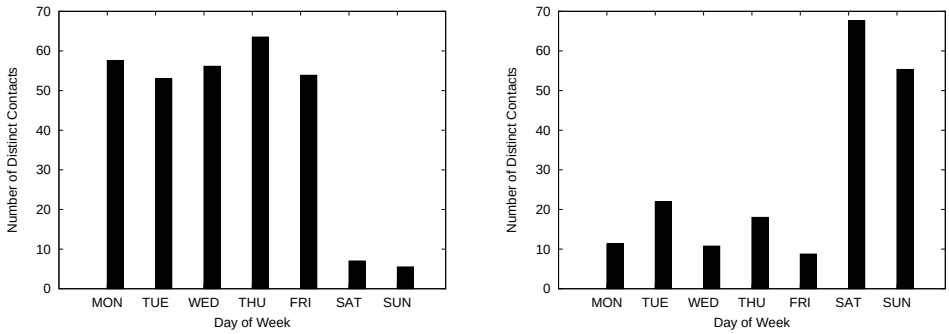
(a) User 1: Wifi Data Set (b) User 2: Wifi Data Set

Figure 7.5: Impact of Scanning Frequency on Wifi Collected Data Set

the base line.

Figure 7.5 shows the impact of scanning frequency on Wifi collected data set. In order to obtaint this figure, for each user, we create the Wifi data sets for the scanning frequencies of 5(m),15(m), 30(m),60(m), and 120(m). Then, we use the collected data with the scan frequency 120(m) as the base line. For each frequency, we count the number of unique Wifi scanned MACs for each day and take the ratio of this number over the corresponding number obtained by the scan frequency 120(m) of the same day. The figure shows that with the highest frequency of 5(m), the Wifi scanner can obtain up to 14 times the number of scanned devices compared to the base line. With our current scanning frequency of 60(s) (used for data in this thesis), we obtain 2.5 times the number of scanned devices compared to the base line.

The analysis shows a clear tradeoff between the scanning frequency and the amount of collected data. With a higher scanning frequency, it is obvious that we obtain more detailed movement



(a) Contact number decreases at the weekend (b) Contact number increases at the weekend

Figure 7.6: Contact number decreases/increases at weekend

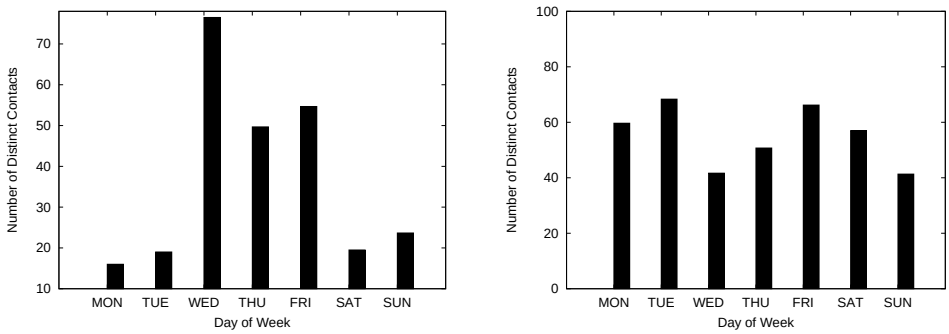
trace. However, we need to configure the experiment phones so that the participants can use the phones as their daily phones and thus they recharge the phones for the prolonged experiment. From our real deployment of the experiment, we learned that participation of the experiment participants is the most crucial factor in collecting movement trace. On the other hand, by setting current scanning frequencies, we can keep the experiment phones up for about 2 days to motivate participants to use the phones as their daily phones. We believe under the battery constraint of the current Google Android phone, our design and configuration of UIM system provides a robust and practical solution to obtain people movement trace.

7.5 Regularity of People Movement Found in Joint Wifi/Bluetooth Trace

We select a set of 50 phones from three data sets D_1, D_2, D_3 in Table 7.1. Let D be the set of 50 selected phones. In this set D, a participant collects from 19 days to 50 days of joint Wifi/Bluetooth trace. In this section, we first study different contact patterns and then present the regularity of contacts found in the Bluetooth trace. Finally, we present regularity of location visit found in the Wifi trace.

7.5.1 Classifying contact pattern

From our analysis on the collected Bluetooth trace, we find that there are four different types of contact patterns as shown in Figures 7.6 and 7.7. In order to obtain these figures, for each experiment phone p, we calculate the average number of unique BT MACs scanned by the Bluetooth scanner of p for a particular day of week such as Monday, Tuesday, ..., Sunday.



(a) Contact number grows high in midweek (b) Contact number stays in a small range

Figure 7.7: Contact number grows high in midweek or stays in a small range

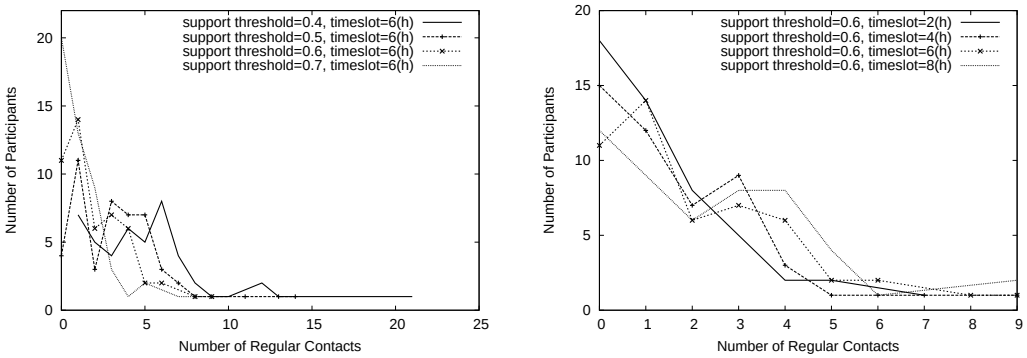
The first contact pattern in Figure 7.6(a) shows that people belong to this class usually have a considerably higher number of contacts during the weekdays and then the number of contacts drops significantly during the weekend. This is the most common contact pattern found in our data set since most people perform the casual routines at work for the weekdays. Then, they take a break during the weekend and stay at home or meet less number of people during the weekend. In constrast, some people may have a totally opposite contact pattern where they meet many more people during the weekends than that of weekdays as shown in Figure Figure 7.6(b).

The third type of contact patterns exists for people who have busiest schedule in the middle of the week as shown in Figure 7.7(a). The contact pattern in this figure looks similar to the bell shape and people with this pattern usually meet many people at midweek only. The last type of contact patterns is the most steady pattern as shown in Figure 7.7(b). People belong to this contact pattern meet a similar number of other people for all days, regardless of weekdays or weekends. These people may be very active and travel a lot, even during the weekend.

7.5.2 Regularity of social contact

In this section, we study the regularity of social contacts between experiment phones and the collected Bluetooth devices. We define a contact as follows: **for an experiment phone p, if a Bluetooth MAC u is found in one scan of p, then p and u have a contact**. Notice that in this definition of contact, we do not consider the duration of contact.

So, given the Bluetooth trace collected by a phone p in the set D of 50 experiment phones, we first find the set U_p of all unique Bluetooth MAC u scanned by the Bluetooth scanner of p. Then, we divide a day time into time slots of size τ. For example, with $\tau = 6(hours)$, we have following



(a) Threhold ϵ_c Sensitivity (b) Time Slot τ Sensitivity

Figure 7.8: Regularity of Social Contact of 50 participants

time slots ([00:00:06:00), [06:00:12:00), [12:00:18:00), [18:00:24:00]).

For each phone p, let D_p be the number of days in which p collected the Bluetooth trace. For a scanned Bluetooth MAC $u \in U_p$, let D_{up} be the number of days p and u have contacts, in which these contacts happen at the same time slot of these days. For example, p and u may have contact at every weekday at 8AM. In our context, **a contact between the phone p and a scanned Bluetooth MAC $u \in U_p$ is the "regular contact" of p if $D_{up} \geq D_p \cdot \epsilon_c$.** Here, ϵ_c is the threshold or the support value. This definition borrows the notion of "regular pattern" in Frequent Pattern Mining [122]. In other words, a contact is considered a regular contact if the phone p and the Bluetooth MAC u meet at the same time slot for at least $D_p \cdot \epsilon_c$ days during the experiment period of D_p days. In order to evaluate the regularity of social contact, we vary the size of time slot τ and the value of threshold (or support value) ϵ_c.

In Figure 7.8(a) we fix the time slot $\tau = 6(hours)$ and vary the value of ϵ_c from 0.4 to 0.7. We find that when ϵ_c increases the number of participants with smaller number of regular contacts increases. For $\epsilon_c = 0.5$ we have 45 participants with at least 1 regular contacts.

In Figure 7.8(b) we fix the $\epsilon_c = 0.6$ and vary the value of τ from 2(hours) to 8(hours). We find that when τ increases the number of participants with more regular contacts increases since we require less strict movement from the participants in the regularity evaluation. When τ varies from 2(hours) to 8(hours), from 35 to 45 participants have at least 1 regular contact.

In conclusion, our evaluation shows that people make regular contacts for their daily activities.

7.5.3 Regularity of location visit

7.5.3.1 Definition of location

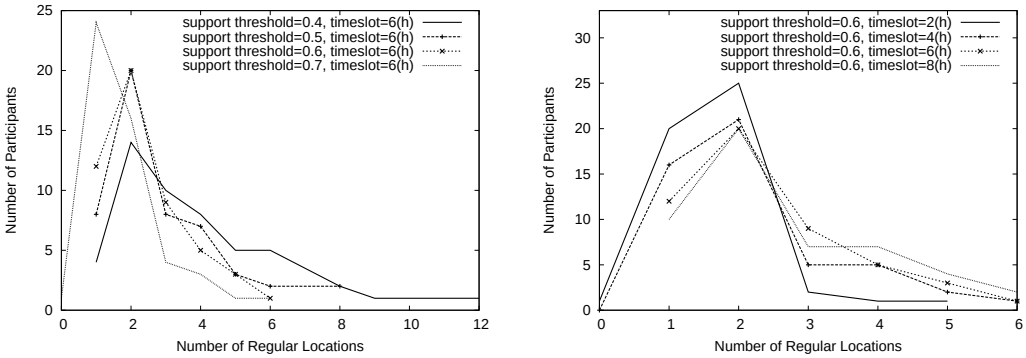
The collected set of Wifi access point MACs includes not only the Wifi MACs associated with buildings in the University of Illinois campus but also ones at the residential homes/apartments where the experiment participants stay. So, the overall scanned Wifi MACs give us the Wifi access point map of the university campus and the surrounding areas where the students and faculties live.

Our Wifi scanner obtains a list of Wifi MACs in the proximity of the experiment phone every 30 minutes. In reality, the Wifi access point usually stays inside its associated physical location and can be used as the "landmark" of the physical location. Since our phone can obtain the Wifi MACs, we basically obtain the landmarks of the locations, or the locations themselves [87]. So, in our context **a "location" is a list of Wifi MACs scanned by the Wifi scanner**.

7.5.3.2 Obtaining locations from Wifi MACs

There are several important aspects we need to consider to define the location from the scanned Wifi MACs. First, the Wifi transmission range of the phone is from 100 to 200 (meters). That means, anytime the phone scans, the list of surrounding Wifi MACs within the transmission range will be captured by our phone, not only the Wifi access point of the building where the phone stays inside. Second, an on-campus building/location usually has a set of Wifi access points. Thus, each scan of our phone will return a list of Wifi MACs. As a result, we define a location in our data set as a unique set of Wifi MACs. From all previous studies [112, 114, 115, 116, 118], only the Reality Mining data set offered some notion of location of the participants by providing the Cellular ID of the cellular base station which was associated with the phone [112]. However, since the cellular base station transmission range varied from several hundred meters (e.g., 500 m) to kilometers (e.g., 30 km), the Reality Mining trace did not provide a fine grain of physical locations as obtained by our Wifi scanner.

In order to obtain location, we apply the UIM Clustering algorithm presented in Section 8.3 for the Wifi trace of the set D of 50 selected phones.



(a) Threhold ϵ_l Sensitivity (b) Time Slot τ Sensitivity

Figure 7.9: Regularity of Location Visit of 50 participants

7.5.3.3 Regular location

In this section we seek the answer for the question: how regular is the location visit pattern in people daily movement? To this end, we divide a day time into time slots of size τ. For example, with $\tau = 6(hours)$, we have following time slots ([00:00:06:00), [06:00:12:00), [12:00:18:00), [18:00:24:00]).

For each phone p, let D_p be the number of days in which p collected Wifi trace, and L_p be set of locations the phone p visits during the experiment period. Notice that L_p is obtained by applying the UIM Clustering algorithm in Chapter 8.3.

For a location $l \in L_p$, let D_{lp} be the number of days p visits location l, in which these visits happen at the same time slot of these days. For example, p may visit location l at every weekday at 2PM. In our context, for a phone p, **a location $l \in L_p$ is the "regular location" of p if** $D_{lp} \geq D_p \cdot \epsilon_l$. This definition borrows the notion of "regular pattern" in Frequent Pattern Mining [122]. In other words, a location l is considered a regular location if the phone p visits l at the same time slot for at least $D_p \cdot \epsilon_l$ days during the experiment period of D_p days. In order to evaluate the regularity of social contact, we vary the size of time slot τ and the value of threshold (or support value) ϵ_l.

In Figure 7.9(a) we fix the time slot $\tau = 6(hours)$ and vary the value of ϵ_l from 0.4 to 0.7. We find that when ϵ_l increases the number of participants with smaller number of regular locations increases. Also, we find that most people have 2 regular locations, which may be their homes and work locations. Except $\epsilon_l = 0.7$, other values of ϵ_l show that all participants have at least 1 regular location.

In Figure 7.9(b) we fix the $\epsilon_l = 0.6$ and vary the value of τ from 2(hours) to 8(hours). We find

that when τ increases the number of participants with more regular locations increases since we require less strict movement from the participants in the regularity evaluation. When $\tau = 2(hours)$, most participants have from 1 to 2 regular locations.

In conclusion, our evaluation shows that people visit regular locations for their daily activities.

7.6 Related Work

There have been several efforts in collecting the human movement trace.

The first type of movement traces was collected by GPS-enabled devices carried by experiment participants [31, 123] in which the geographical coordinates of the experiment devices were obtained together with the timestamp. However, these devices could not collect the accurate geographical traces when they were indoor, which resulted in the wrong movement pattern. More importantly, the collected geographical locations can not be used to infer the connectivity between two geographically closed nodes since there might be obstacles between them. Meanwhile, connectivity is a crucial and fundamental characteristic used to evaluate the performance of protocols for wireless networks.

The second type of movement traces was collected from WLAN environments where the association between the laptop/PDA and the Wifi access points was captured with the corresponding time stamps [121, 124]. The information collected from these traces included the Wifi MAC addresses of the laptops and the MAC addresses of their associated Wifi access points. Since the laptop had a good battery capacity and laptop users usually charged their laptops when using the wireless network, the collected traces from WLAN provided a rich set of continuous data. Also, since the laptops were popular in corporate environments and university campuses, the trace collection experiment was easily scaled up to the entire corporate [125] or campus [124] environments. This offered a comprehensive set of wireless usage and detailed associations of the laptop devices and the Wifi access points [126]. Previous work used these WLAN traces to infer the location of the experiment devices, derived various mobility models [124, 127], and used these derived mobility models to validate performance of network protocols for MANETs and DTNs. However, there was a fundamental weakness of these trace collection methods. The reason was that the collected trace did not always represent the real movement of people and this might result in the wrongly derived mobility models. Obviously, the laptop user did not always turn on the laptop and did not carry it with her all the time. Moreover, a normal laptop user usually turned on her laptop and left

it on her office desk when she was doing other things (e.g., had lunch with friends, had meetings with colleagues, or went to exercise at the gym). Therefore, the location information inferred from the WLAN trace may not be the needed fine granularity. So, the collected associations of laptops and the Wifi access points could be used to understand the wireless usage rather than the real movement of people.

The third type of movement traces was collected by using portable (experiment) devices such as PDA, iMote, cell phone. These portable devices were assigned to participants so that they would carry the devices all the time when they were walking. The collected information included the Bluetooth ad hoc contacts between the experiment devices and external Bluetooth-enabled devices, or among experiment devices only. The collected data had the list of scanned Bluetooth MAC addresses with the corresponding time stamps [113, 112, 114, 115, 116, 117, 118, 128]. Due to the limitation of battery and the hardware capability of the experiment devices, only the Bluetooth ad hoc contacts were collected. Moreover, the scale of these experiments is much smaller in the number of participants and shorter in the experiment duration than those of WLAN experiments. It is clear that this method of trace collection captured more realistic movement trace since with high probability, the experiment devices were carried by the participants. However, this method of the movement trace collection did not collect the location information of the people movement, a critical factor to understand the movement behavior of people. Except [112], all previous works [113, 114, 115, 116, 117, 118] did not capture the location information of the movement. For [112], the location was inferred from the cellular ID associated with the experiment phone. However, since the transmission range of the cellular base station was ranging from several hundred meters (e.g., 500 m) to kilometers (e.g., 30 km), the location information inferred from the cellular ID did not provide the needed fine granularity. From our observation, the Wifi MAC address of the Wifi access point could be used to represent the location [87] since a Wifi access point usually is associated with a physical building or geographical location. Hence, this motivates us in designing a trace collection methods and implement the UIM scanning system to obtain both Bluetooth ad hoc contacts and Wifi MAC addresses of Wifi access points.

7.7 Conclusion

We present a new methodology of collecting people movement trace by using the cellular phones. We then apply the methodology to implement the UIM scanning system, which collects both social

contact and location information. To the best of our knowledge, the UIM system is the first system to collect both location and social contact information in one people movement trace.

By studying the regularity of people movement found in the trace collected by the UIM system, we find that although the contact patterns of people can be classified into 4 distinct classes, their daily movements exhibit regular patterns in which people: (1) visit regular locations and (2) have contacts with a regular set of people. This result motivates us to construct a predictive model of people movement in Chapter 8 and leverage this predictive model for a new content distribution protocol in Chapter 9.

CHAPTER 8

JYOTISH: A NOVEL METHOD FOR CONSTRUCTING PREDICTIVE MODEL OF PEOPLE MOVEMENT FROM JOINT WIFI/BLUETOOTH TRACE

8.1 Introduction

The ability to correctly predict the movement of people is crucial to the design of efficient data dissemination protocols and to the network resource planning for Infrastructure-based wireless networks, Mobile Ad hoc Networks (MANET), and Delay Tolerant Networks (DTN). While predicting the movement of a person, we are seeking answers to three fundamental questions: (1) where will the person stay at a future time? (location), (2) How long will she stay at the location? (stay duration), and (3) Who will she meet at a future time? (people/social contact). Providing answers to these questions altogether is challenging due to the dynamic nature of people movement and the lack of a large-scale people movement trace to construct a predictive model, which can predict people movement with a high precision.

In Chapter 7, we designed and implemented the UIM scanning system to collect both MAC addresses of Wifi access point and Bluetooth ad hoc contacts. The UIM system was deployed to Google phones carried by 123 experiment participants in University of Illinois campus from March to August 2010. Then, collected MAC addresses of Wifi access point were used to infer location and collected MAC addresses of Bluetooth devices were used to infer social contact. The characterization study in Section 7.5 shows that people movement in University of Illinois campus exhibits a relatively high regular pattern. In this chapter, we propose a novel method named Jyotish [1] that exploits the regularity of people movement found in the joint Wifi/Bluetooth data set to construct a predictive model to predict the location, stay duration, and contact. In summary, this Chapter has following contributions:

1. To the best of our knowledge, Jyotish is the first method, which constructs a predictive model of people movement from the joint Wifi/Bluetooth trace.

[1] In Sanskrit, Jyotish (Ji-o-tish) is the person who predicts future events.

2. Also, to the best of our knowledge the constructed predictive model is the first to predict future location, stay duration at the location, and contact altogether.

3. We present an efficient clustering algorithm to cluster Wifi access point information into locations by exploiting the regularity of people movement. Our algorithm overcomes the Wifi signal fluctuation in previous work [129, 130] and provides a finer grain of location than that derived from cellular base station [131].

4. We evaluate the constructed predictive model over the real Wifi/Bluetooth trace collected by 50 experiment participants in University of Illinois campus from March to August 2010.

This Chapter is organized as follows. We first present the joint Wifi/Bluetooth trace collected by UIM system and the overview of the predictive model in Section 8.2. Then, we present a clustering algorithm to cluster Wifi access points in the Wifi trace into locations in Section 8.3. These locations will be assigned to records in Bluetooth trace in Section 8.4. Then, the Bluetooth trace with assigned locations is used to construct location predictor, duration predictor, and contact predictor in Section 8.5. We evaluate our predictors in Section 8.6. Finally, we present related work in Section 8.7 and conclude the Chapter in Section 8.8.

8.2 Overview of UIM Trace and Jyotish Method

8.2.1 UIM collected trace

We deployed the UIM scanning system [44] on Google phones carried by 123 participants from March to August 2010 in University of Illinois campus, with three rounds: from beginning of March to end of March, from beginning of April to mid of May, and from end of May to mid of August. Many participants participated from one month to two months of experiment.

UIM system has a Wifi scanner and a Bluetooth scanner. The former periodically (i.e., every 30 minutes) captures MAC addresses of Wifi access points while the latter periodically (i.e., every 60 seconds) captures MAC addresses of Bluetooth-enabled devices in proximity of the experiment phones. The above scanning frequencies are set to conserve the phone battery since: (1) most participants use the experiment phones as their daily phones, and (2) the Wifi scanner consumes much more power than the Bluetooth scanner. These scanning frequencies conserve the phone

battery for 2 days (including other usages of participants) and make it acceptable for the participants to carry the phones for the prolonged experiment. The traces collected by scanners are called "Wifi trace" and "Bluetooth trace" respectively. Henceforth, we use the terms Bluetooth and BT interchangeably.

Scan Time	**Wifi MACs**
03/08/10 09:20	a_1, a_3
03/08/10 09:50	a_1, a_5
03/08/10 10:20	a_6
03/08/10 13:50	a_4, a_7, a_9
03/14/10 08:20	a_1, a_3

Table 8.1: Example of Wifi trace W

In order to clarify the presentation of this chapter, we use Relational Algebra [132] to represent and manipulate collected data set. So, we use the terms "set", "table" and "relation" interchangeably, "record" and "tuple" interchangeably. Also, we use "person" and "phone" interchangeably, "stay duration" and "duration" interchangeably. For an experiment phone p, let D be the entire collected data set, so $D = W \cup B$, in which W is the relation representing the collected Wifi trace and B is the relation representing the collected Bluetooth trace. Tables 8.1 and 8.2 show examples of W and B. W has multiple Wifi tuples: $W = \{w_1, w_2, w_3, ..., w_{|W|}\}$. Each tuple $w_i \in W$ is in the format of $w_i =< t_i, A_i >$, where A_i a set of Wifi MACs returned from one Wifi scan and t_i is the scan time of that Wifi scan. So, we have $A_i = \{a_1, a_2, ..., a_j, ...\}$, in which a_j is the j^{th} Wifi MAC scanned by the Wifi scanner of p during the entire experiment period. In Table 8.1, each row is one tuple w_i. Let W_A be the set of all Wifi MACs scanned by the Wifi scanner for the entire experiment period of one experiment phone. For the Table 8.1, $W_A = \{a_1, a_3, a_4, a_5, a_6, a_7, a_9\}$.

Scan Time	**BT MACs**
03/08/10 09:20	u_1, u_3
03/08/10 09:21	u_1, u_3
03/08/10 09:22	u_1
03/08/10 13:50	u_4, u_9
03/14/10 08:14	u_1, u_3, u_8

Table 8.2: Example of BT trace B

Similarly, the relation B has multiple BT tuples: $B = \{b_1, b_2, b_3, ..., b_{|B|}\}$. Each tuple $b_i \in B$ is in the format of $b_i =< t_i, U_i >$, where U_i a set of BT MACs returned from one BT scan and t_i is the scan time of that BT scan. So, we have $U_i = \{u_1, u_2, ..., u_j, ...\}$, in which u_j is the j^{th} BT MAC scanned by the BT scanner of p during the entire experiment period. Let B_A be the set of all BT MACs scanned by the BT scanner for the entire experiment period of one experiment phone. For the Table 8.2, $B_A = \{u_1, u_3, u_4, u_8, u_9\}$. Notice that since the Wifi scanner and BT scanner run concurrently, the scan times of tuples of W and B overlap.

8.2.1.1 Contact definition

We say *the experiment phone p has a contact with a device p_j whose BT MAC is u_j if u_j appears in one tuple of p's BT trace B*. This contact definition can also be found in previous papers [114, 115]. We assume that when p and p_j have a contact, the user of p and the user of p_j have a social contact. Henceforth, we use the term "social contact" and "contact" interchangeably.

8.2.2 Jyotish overview

Figure 8.1 shows steps of the Jyotish method to construct the predictive model from the joint Wifi/Bluetooth trace D. In the first and the second steps, we cluster Wifi records in W into locations (see Section 8.3). Then, in step 3 and 4, we construct a Naive Bayesian classifier to assign locations for records in BT trace B (see Section 8.4). In step 5 and 6, the BT trace with assigned location is used as the input to construct location predictor, stay duration predictor, and contact predictor (see Section 8.5). Henceforth, we use "stay duration" and "duration" interchangeably.

8.3 Clustering Wifi Records into Locations

This section presents an algorithm called "UIM Clustering" to cluster Wifi records into clusters. This section focuses on step 1 and 2 in Figure 8.1.

8.3.1 UIM clustering algorithm overview

There are several challenges in obtaining locations from Wifi records of W. First, since the Wifi signal fluctuates, although the phone stays in one fixed position, it may obtain different results for

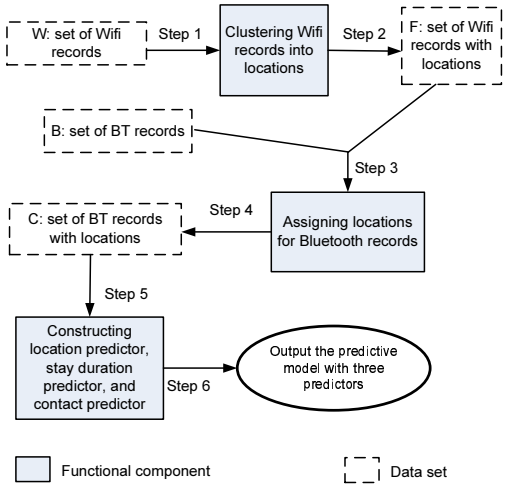


Figure 8.1: Overview of Jyotish

different Wifi scans. Previous works [129, 130] used the signal strength to cluster Wifi MACs into locations and suffered from the Wifi signal fluctuation. Second, if the phone is in the middle of two adjacent buildings, the Wifi scanned result might be partially overlapped with the scanned results obtained when the phone stays inside either of the buildings. Fortunately, the movement pattern of people is relatively regular since they tend to stay more frequently at their regular places. So, if two Wifi MACs a_1, a_3 appear together more frequently than two Wifi MACs a_1, a_5 in the Wifi trace W, then it is likely that a_1, a_3 stay close in a physical building. That means, it is better to group a_1 and a_3 into the same location than a_1 and a_5. So, we exploit the regularity of people movement to cluster Wifi MACs into locations. Moreover, our approach provides a finer grain of location than that derived from cellular base station [131] since the transmission range of Wifi access points is much shorter than that of the cellular base station.

Name	**Description**
Δ	Set of good records of W, $\Delta \subset W$
γ_{A_i}	Binary bit vector of A_i, $\lvert\gamma_{A_i}\rvert = \lvert W_A\rvert$
γ	Set of binary vectors: $\gamma_{A_i} \in \gamma$ where $A_i \in \Delta$
G_θ	Similarity graph: $G_\theta =< V_\theta, E_\theta >$
C_C	Candidate Cluster Set obtained from G_θ
$\gamma^S_{C_i}$	Signature vector of cluster $C_i \in C_C$
C_F	Final Cluster Set obtained from C_C
θ	Similarity threshold

Table 8.3: Major notations used by UIM Clustering Algorithm

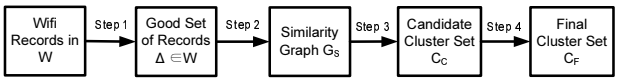


Figure 8.2: Execution of UIM Clustering algorithm

In our algorithm, for each record (or tuple) $w_i =< t_i, A_i >$, we do not use the scan time t_i and only use A_i. Thus, in this section, we use A_i to represent the record w_i. In other sections, we use w_i to represent the record i^{th} of W. We first define *location as a unique set of Wifi MACs, which appear* ***frequently*** *together in the records of* W. In Table 8.1, the pair a_1, a_3 appears twice together while a_1, a_5 appears once. So, we say a_1, a_3 appear together more frequently in W than a_1, a_5.

Figure 8.2 shows the execution block diagram of the UIM Clustering algorithm. In step 1, given the records in W, we obtain the sub set of *good* records $W_G \subset W$ (see Section 8.3.2). In step 2, we measure the similarity between all pairs of records of W_G and construct a similarity graph G_S, in which each vertex of G_S is a record of W_G. In step 3, we apply the Star Clustering algorithm [133] to cluster vertexes into a set C_C of candidate clusters. Finally, candidate clusters are merged based on their similarity measures to obtain the set C_F of final clusters. Each cluster in C_F can be used to represent one location. Table 8.3 represents major notations used by the UIM Clustering algorithm.

8.3.2 Obtaining good set Δ of Wifi records

This section focuses on the Step 1 in Figure 8.2. First, we define *a good record as a record that consists of Wifi MACs appearing* ***frequently*** *together in the records of* W. We determine if a record $A_i \in W$ is a good record as follows: for each pair of Wifi MACs $(a_j, a_k) \in A_i$, we calculate the support value $s_{j,k}$, which represents how frequently the pair (a_j, a_k) appears together in the same records of W:

$$s_{j,k} = \frac{c(a_j, a_k)}{min\{c(a_j), c(a_k)\}} \tag{8.1}$$

In Equation 8.1, $c(a_j)$ is the number of records $A_i \in W$ in which $a_j \in A_i$. $c(a_j, a_k)$ is the number of records $A_i \in W$ in which $a_j \in A_i, a_k \in A_i$. Intuitively, $s_{j,k}$ is similar to the notion of support value of Frequent Item Set in Data Mining literature. For the denominator of Equation 8.1, we have min of $c(a_j)$ and $c(a_k)$ since we are interested in the Wifi MAC which appears in less number of records and the association of this Wifi MAC with the other one in the pair. This min value represents the coexistence of the two Wifi MACs in the records of W. We have $s_{j,k} \in [0, 1]$ and

the greater value of $s_{j,k}$ means the two Wifi MACs appear together in the same records of W more frequently.

Let $|A_i|$ be the number of Wifi MACs of the record A_i. For each $A_i \in W$, we have $\binom{|A_i|}{2}$ pairs of Wifi MACs and $\binom{|A_i|}{2}$ support values, which constitutes a distribution. Let λ_A and ξ_A be the mean and standard deviation of this distribution. If A_i has only one Wifi MAC, then $\lambda_A = 1, \xi_A = 0$. Intuitively, we prefer a greater value of λ_A since it means A_i contains Wifi access points that often appear together in the records of W. We prefer a smaller value of ξ_A since it means the support values stay in a small range. So, for A_i, we calculate the ratio $\frac{\xi_A}{\lambda_A}$ to: (1) select good record whose Wifi MACs appear together frequently in the same records of W, and (2) remove the bad records consisting of wifi MACs, which do not frequently appear together in records of W. Let F_W be the set of ratios $\frac{\xi_A}{\lambda_A}$ of all records of W. We then sort F_W increasingly and create the set Δ of good records from W using F_W as follows. Let Δ_A be the set of all Wifi MACs in the records of Δ. We scan F_W from the beginning and for a record $A_i \in F$, A_i is added to Δ if adding Wifi MACs of A_i to Δ_A increases the size of Δ_A. We stop adding records from F to Δ when $|\Delta_A| = |W_A|$. Since the added records into Δ has a small value of $\frac{\xi_A}{\lambda_A}$, we reduce the size of Δ and remove most of noise data in W.

8.3.3 Constructing similarity graph G_θ

This section focuses on the Step 2 in Figure 8.2. Given the good set Δ, we convert $A_i \in \Delta$ into a binary bit vector γ_{A_i} as follows. If the Wifi MAC $a_j \in A_i$, then the j^{th} bit of the vector γ_{A_i} is set to 1, $\gamma_{A_i}[j] = 1$; otherwise, $\gamma_{A_i}[j] = 0$. Figure 8.3 shows an example of the binary bit vector. Notice that $|\gamma_{A_i}| = |W_A|$.

Let γ be the set of binary vectors obtained from all records $A_i \in \Delta$. Then, we use the Tanimoto coefficient [134] (the cosine similarity for binary vectors) to calculate the similarity measure $T_{p,q}$ between a pair of vectors $\gamma_p \in \gamma, \gamma_q \in \gamma$:

$$T_{p,q} = \frac{\gamma_p \cdot \gamma_q}{||\gamma_p||^2 + ||\gamma_q||^2 - \gamma_p \cdot \gamma_q} \tag{8.2}$$

Next, we construct the similarity graph $G_\theta =< V_\theta, E_\theta >$, in which each vector $\gamma_p \in \gamma$ is considered a vertex $v_p \in V_\theta$. For a pair of vertexes $v_p, v_q \in V_\theta$, the edge (v_p, v_q) exists (i.e., $(v_p, v_q) \in E_\theta$) if $T_{p,q} \geq \theta$. θ is a threshold that determines the topology of G_θ and has important impacts on the clustering result (see Section 8.3.6).

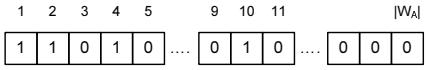


Figure 8.3: Bit vector γ_{A_i}, with $A_i = \{a_1, a_2, a_4, a_{10}\}$

8.3.4 Obtaining candidate cluster set C_C

This section focuses on Step 3 in Figure 8.2. Particularly, we apply the Star Clustering algorithm [133] to cluster vertexes of G_θ into clusters since Star Cluster does not require a pre-defined number of clusters like others such as k-means and hierarchical clustering. Star Clustering thus fits very well to our context since we do not know in advance the number of locations from the Wifi trace W. Applying Star Clustering algorithm, we first sort the vertexes decreasingly according to their node degrees. Then, we scan the sorted list of vertexes, for each vertex v_p if v_p is not in any clusters, v_p is considered as the center of a new cluster. For each neighbor v_q of v_p, if v_q does not belong to any clusters, v_q is included in the cluster centered at v_p. The process continues until all vertexes belong to clusters. We denote this set of clusters *the candidate cluster set* C_C.

8.3.5 Obtaining final cluster set C_F

This section focuses on the Step 4 in Figure 8.2. For a cluster $C_i \in C_C$, C_i consists of a set of vertexes, each vertex is a binary vector representing a record $w \in \Delta$. Let $\gamma^S_{C_i}$ be the signature vector of the cluster C_i. $\gamma^S_{C_i}$ is obtained by applying the OR bitwise operation over all the binary vectors of C_i. Intuitively, the signature vector $\gamma^S_{C_i}$ represents the set of Wifi MACs, which belong to the cluster C_i. Thus, the signature vector $\gamma^S_{C_i}$ can be used to uniquely distinguish clusters in C_C. Then, we use the signature vectors to merge cluster $C_1 \in C_C$ into cluster $C_2 \in C_C$ if C_1 is a sub cluster of C_2. Formally, C_1 is merged into C_2 if $\gamma^S_{C_2} = (\gamma^S_{C_1}\ OR\ \gamma^S_{C_2})$. So, we have the final set of clusters C_F, in which each cluster $C_j \in C_F$ can be used to represent one particular location.

Given the final cluster set C_F, we classify all Wifi records $A_i \in W$ into clusters in C_F as follows. Each record $A_i \in W$ is classified to the best matched cluster $C_i \in C_F$ based on the similarity measure between γ_{A_i} and $\gamma^S_{C_i}$ calculated by Equation 8.2. The output of this step is the relation F of all Wifi tuples in W with assigned locations as shown in Table 8.4. Formally, $F = \{w_i =< t_i, A_i, L_i >: w_i =< t_i, A_i >\in W\}$, where L_i is the location assigned by the UIM Clustering algorithm to w_i.

Scan Time	Wifi MACs	Loc
03/08/10 09:20	a_1, a_3	L_1
03/08/10 09:50	a_1, a_5	L_1
03/08/10 10:20	a_6	L_5
03/08/10 13:50	a_4, a_7, a_9	L_8
03/14/10 08:20	a_1, a_3	L_1

Table 8.4: Example of relation F.

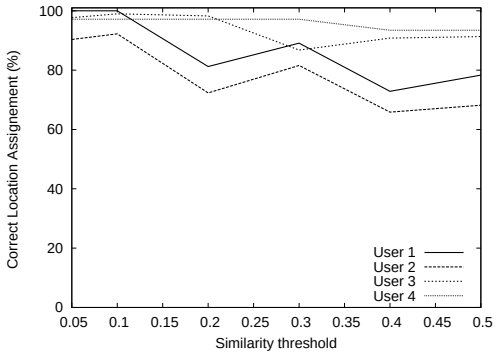
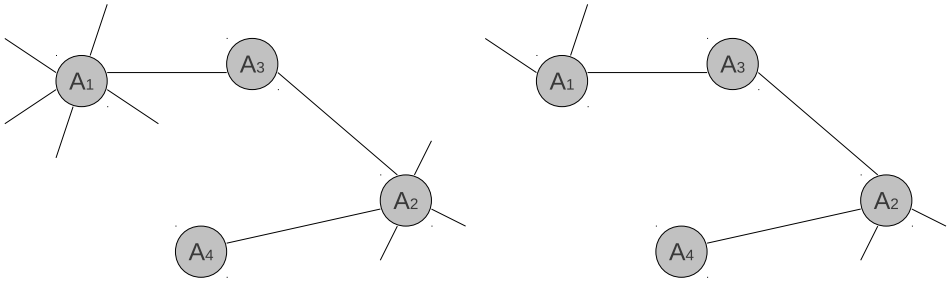


Figure 8.4: $\theta = 0.1$ gives most correct locations

8.3.6 Setting similarity threshold θ

In this section, we empirically set the value of θ as follows. We first select 4 different participants and create for each of them a development set W_D, which consists of 64 Wifi records scanned in two different days. Then, we ask the participants to manually label the location for their Wifi records (e.g., Long's home, Quang's home, Klara's office, etc.). For each value of $\theta \in [0.05, 0.9]$, we perform following steps. For each pair of records $(A_1, A_2) \in W_D$, we check cluster ids of A_1 and A_2 in F and compare these cluster ids with the labeled locations in W_D. A location assignment made by UIM Clustering algorithm is correct if: (1) A_1 and A_2 have the same labeled location in W_D and they are assigned into the same cluster in F, or (2) A_1 and A_2 have different labeled locations in W_D and they are assigned into different clusters in F. Figure 8.4 shows the percentage of correct classification the clustering algorithm makes when θ varies from 0.05 to 0.5. The best value of all people is 0.1, in which the correct prediction for all 4 people is greater than 96%. When $\theta = 0.05$, clusters are merged into big cluster; or nearby locations are merged into one location, it may incur "too big locations" and result in incorrect location assignment. In contrast, when θ increases (e.g., $0.1 < \theta \leq 0.9$), two Wifi records of the same physical location may be assigned into different clusters. In other words, when θ increases, the accuracy of location assignment decreases since with a higher value of θ the node degree of the graph is smaller and the graph is sparser,



(a) With θ_1, A_1 is processed before A_2 (b) With θ_2, A_2 is processed before A_1

Figure 8.5: Location assignment for $\theta_1 < \theta_2$. A_3, A_4 are manually marked the same location by user.

which results in more number of "smaller" clusters. For example, with $\theta = 0.1$, we have more number of accurate location assignments than that of $\theta = 0.5$.

From Figure 8.4, we also observe that for two values of $\theta_1 < \theta_2$ and the difference between θ_1 and θ_2 is small, the location assignment of θ_1 may not be always better than that of θ_2. This is true for User 3 and User 4 in Figure 8.4 with $\theta_1 = 0.2$ and $\theta_2 = 0.3$. This result is explained in Figure 8.5 as follows. In this figure, A_1 and A_2 are selected as cluster centers by the Star Clustering Algorithm in the clustering process since they have a higher node degree. Also, A_3 and A_4 are in the same physical location as manually marked by the experiment participants. For θ_1 in Figure 8.5(a), A_1 will be processed prior to A_2 by the Star Clustering Algorithm since A_2 has a higher node degree. In this case, A_3 belongs to the cluster centered at A_1 and A_4 belongs to cluster centered at A_2. This results in a wrong location assignment. In constrast, for $\theta_2 > \theta_1$ in Figure 8.5(b), the node degree of A_1 and A_2 changes and that makes A_2 be processed by the Star Clustering Algorithm prior to A_1. In this case, A_3 and A_4 belong to the same cluster and this is a true location assignment when compared to user manually marked data. So, if the difference between $\theta_2 > \theta_1$ is small, θ_2 may result in a better location assingment. However, if the difference between $\theta_2 > \theta_1$ is big, θ_2 results in a worse location assignment. This is confirmed in Figure 8.4 where $\theta = 0.2$ gets a better location assigment than that of $\theta = 0.5$.

Essentially, the value of θ for one person should be set by analyzing his own movement trace since people have different movement habits and they visit different locations, which have different densities of wireless access points. For our evaluation in Section 8.6, we use $\theta = 0.1$ to evaluate the predictors since: (1) we can not ask all 123 participants to mark their locations manually for each wifi scan in the data set, and (2) $\theta = 0.1$ gives very high location assigments for 4 people above.

Name	Description
μ	# of unique BT MACs collected by all phones
F	Relation of Wifi tuples with assigned locations
M	Relation of BT tuples created by F, B and α
C	BT trace with assigned locations
α	Time window (in second)
β_{min}	Threshold to assign "Unknown" location
ν	Type of day, $\nu \in \{weekend, weekday\}$
τ	Time slot

Table 8.5: Major notations used by Predictive Model

8.4 Assigning Locations for Bluetooth Records

Although tuples of F are assigned locations, they do not provide needed granularity since the Wifi scanner scans each every 30 minutes. During this period, the phone may move to different locations. Meanwhile, our BT scanner scans every minute. Our goal is to assign locations from tuples of F to tuples of B and thus obtain the finer granularity of people movement. The first step towards this goal is to map tuples of F and tuples of B using a time window α. This section focuses on step 3 and 4 in Figure 8.1. Table 8.5 presents the major notations used in following sections.

8.4.1 Mapping between Wifi records and BT records using time window α

For a tuple $w_k =< t_k, A_k, L_k > \in F$, we know that the person p stays at the location L_k at time t_k. We observe that during the time window $[t_k - \alpha, t_k + \alpha]$, if α is short enough, the person usually stays at the location L_k. Therefore, we can assign the location L_k to all BT records $b_i =< t_i, U_i > \in B$, in which $t_k - \alpha \leq t_i \leq t_k + \alpha$.

Let M be the relation of all BT tuples $b_i \in B$, which are assigned locations L_k by using the time window α and the tuple $w_k =< t_k, A_k, L_k > \in F$. Formally, $M = \{b_i' =< t_i, U_i, L_k >: b_i =< t_i, U_i > \in B, w_k =< t_k, A_k, L_k > \in F, t_k - \alpha \leq t_i \leq t_k + \alpha, 1 \leq i \leq |B|, 1 \leq k \leq |F|\}$. Table 8.6 shows an example of the relation M, which is created through the mapping between relation B and relation F using the time window α. We will present a separate section on how to set the value of α in Section 8.4.3.

Scan Time	BT MACs	Loc
03/08/10 09:20	u_1, u_3	L_1
03/08/10 09:21	u_1, u_3	L_1
03/08/10 13:50	u_4, u_9	L_8
03/14/10 08:14	u_1, u_3, u_8	L_1

Table 8.6: Example of relation M.

8.4.2 Assigning locations for Bluetooth records

We construct a Naive Bayesian classifier N_B to predict the locations of all BT records in B. Basically, we use the relation M to train the Naive Bayesian classifier N_B and then use N_B to assign locations to all records $b_i \in B$.

8.4.2.1 Training Naive Bayesian classifier N_B

For a BT record $b_i \in B$, the probability that b_i belongs to a location L_k is calculated by using the Bayesian Theorem as follows:

$$P(L_k|b_i) = \frac{P(b_i|L_k)P(L_k)}{P(b_i)} \tag{8.3}$$

Then, b_i belongs to the location L_{b_i} calculated as follows:

$$L_{b_i} = \arg\max_k \frac{P(b_i|L_k)P(L_k)}{P(b_i)} \tag{8.4}$$

Since $P(b_i)$ is the same for all locations L_k, we calculate $f(L_k) = P(b_i|L_k)P(L_k)$. To calculate $P(b_i|L_k)$, we assume that for $u_1 \in b_i$[2] and $u_2 \in b_i$, u_1 and u_2 are conditionally independent, or u_1 and u_2 appear conditionally independent in the proximity of the experiment phone when they are scanned (and b_i is created) by the BT scanner. This assumption usually holds in reality since people (with their Bluetooth-enable devices) appear at locations independently. Let $f(L_k) = \Pi_{u_j \in b_i} P(u_j|L_k)P(L_k)$, we have:

$$L_{b_i} = \arg\max_k \ f(L_k) \tag{8.5}$$

The relation M is used to calculate $f(L_k)$ in Equation 8.5 as follows. $P(L_k) = \frac{c(L_k)}{|M|}$, where $|M|$ is the size of M and $c(L_k)$ is the number of tuples $b_i' =< t_i, U_i, L_i > \in M$, in which $L_i = L_k$. For $P(u_j|L_k)$, we have:

[2]The correct notation should be $u_1 \in U_i$ and $b_i =< t_i, U_i >$. However, to shorten the notation, we use $u_1 \in b_i$ in this section.

$$P(u_j|L_k) = \frac{c(u_j)}{c(L_k)} \tag{8.6}$$

In Equation 8.6, $c(u_j)$ is the number of records $b_i{}' =< t_i, U_i, L_i >\in M$, in which $L_i = L_k$ and $u_j \in U_i$. Applying Equation 8.5 and Equation 8.6 for all records of M, we have the trained classifier N_B.

8.4.2.2 Applying additive smoothing technique

In section 8.4.1, since we only use a small time window α to create M, M does not cover all BT MACs in B. Thus, applying the trained classifier N_B for a record $b_i \in B$, the value $c(u_j)$ in Equation 8.6 might be 0 if u_j does not belong to any tuples of M. Thus, $c(u_j)$ cancels out the value of $P(u_l|L_k)$ of BT MACs $u_l \in b_i$ (i.e., $j \neq l$) in Equation 8.5. To avoid this, we apply the Additive Smoothing technique [135] for the Equation 8.6 as follows:

$$P(u_j|L_k) = \frac{c(u_j) + 1}{c(L_k) + \mu} \tag{8.7}$$

In Equation 8.7, μ is the number of unique BT MACs collected by all participants for the entire experiment period. Adding μ to the denominator of Equation 8.7 means we take into account all possible BT MACs in calculating the probability of the BT MAC u_j. With Equation 8.7, $P(u_j|L_k) \neq 0$ for all u_j and we have:

$$f(L_k) = \Pi_{u_j \in b_i} \frac{c(u_j) + 1}{c(L_k) + \mu} P(L_k) \tag{8.8}$$

So, we have a new trained classifier N'_B by applying Equation 8.5 and Equation 8.8 for all tuples of M.

8.4.2.3 "Unknown" location

Applying N'_B to assign locations to BT records $b_i \in B$, we encounter records b_i whose value of $f(L_{b_i})$ calculated by Equation 8.4 is extremely small. These records b_i are scanned in the middle of two consecutive Wifi scans (the period between two Wifi scans is 30 minutes) when the phone carrier moves to another location, which is not captured by the Wifi scanner. Therefore, assigning any known location from the Wifi trace to b_i results in a wrong assignment. To avoid this, we

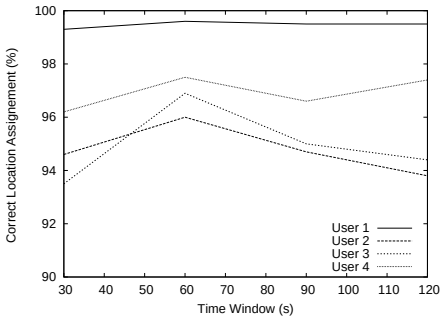


Figure 8.6: Time window $\alpha = 60(s)$ gives most correct locations

define a new location named "Unknown" location and assign the "Unknown" location to b_i.

The next question is "How small the value of $f(L_{b_i})$ is" so that the record b_i is assigned to "Unknown" location. To answer this question, we use Equation 8.8 to calculate $f(L_{b_i'})$ for all records $b_i' \in M$. Let $\beta_{min} = min_{b_i' \in M} f(L_{b_i})$. We then use β_{min} as the threshold value to assign "Unknown" location to a BT record $b_i \in B$. The intuition is as follows. We assume that records in M are "good records" whose locations are assigned correctly by the time window α. So, the minimum value of $f(L_{b_i'})$ of all records $b_i' \in M$ represents the cutoff value for all records whose locations are assigned correctly. For a record $b_i \in B$, we have:

$$L_{b_i} = \begin{cases} \arg\max_i \quad f(L_k) & \text{if} \quad f(L_{b_i}) \geq \beta_{min} \\ \text{"}Unknown\text{"} & \text{otherwise} \end{cases} \tag{8.9}$$

Equation 8.9 means b_i will be assigned L_{b_i} location only if $f(L_{b_i}) \geq \beta_{min}$. Otherwise, b_i will be assigned the "Unknown" location. Although this approach seems to be conservative in assigning correct locations to BT records, it does provide good result in our evaluation of the predictive model in Section 8.6. Let C be the relation consisting of all records in B, which are assigned locations. Then, we sort C increasingly according to the scan times of its tuples and use C as the input to construct our predictors in Section 8.5.

8.4.3 Setting time window α

As we presented in Section 8.4.1, the value of α decides the mapping between Wifi records and BT records and the size of relation M, which is used to train the Naive Bayesian classifier N_B. In this section, we use the same technique in Section 8.3.6 to empirically set value for α.

Particularly, we select 4 participants and for each of them, we create a set B_D of BT records of two days and ask the participants to manually label locations for records in his B_D. For two days, each B_D has 960 records. For each pair of records $(b_1, b_2) \in B_D$, we check locations of b_1 and b_2 in C assigned by our Naive Bayesian classifier and compare these locations with the labeled locations in B_D. Figure 8.6 shows that when $\alpha = 60(s)$, the relation C outputted by the Naive Bayesian classifier obtains the best location assignment, in which the correct prediction for all 4 people is greater than 95%. With $\alpha = 30(s)$, the relation M consists of too few records to train a good Naive Bayesian classifier. Meanwhile, $\alpha > 60(s)$ is too large a time window, which incurs noisy data in the relation M since BT records may be assigned wrong locations if they fall into this big time window. The trained classifier N_B then performs worse than that with $\alpha = 60(s)$. So, we use $\alpha = 60(s)$ to evaluate the performance of our predictive model in Section 8.6.

8.5 Constructing Location Predictor, Duration Predictor, and Contact Predictor

Given the relation C, we construct the location predictor, duration predictor, and contact predictor. This section focuses on step 5 and step 6 in Figure 8.1. To construct our predictors, we use two parameters: type of day and time slot. Let ν be the "type of day" and τ be the "time slot". Particularly, we classify days into two types: weekend and weekday, so $\nu \in \{weekday, weekend\}$, and divide time of a day into time slot of size 1, 2, 4, etc. hours. The motivation for the use of these two parameters is that people may visit different places and contact different people for the weekday and weekend. For each record $r \in C$, we map r's scan time into type of day ν and time slot τ. Table 8.7 shows an example of the relation C in which its tuples are mapped into type of day and time slot of size 2 hours. The relation C in this new format is used to construct our predictors.

For a person p, the input query for p's movement prediction is a record X in the format of $X = \{\nu_1, \tau_1\}$, in which ν_1 represents the type of day and τ_1 represents the time slot. The output will be location the p stays at, the duration p stays at the location, and contacts p has for the type of day ν_1 and during time slot τ_1.

ν	τ	**Scan Time**	**Loc**	**BT MACs**
weekday	08-10	03/08/10 09:20	L_1	u_1, u_3
weekday	08-10	03/08/10 09:21	L_1	u_1, u_3
weekday	08-10	03/08/10 09:22	L_1	u_1, u_8
weekday	12-14	03/08/10 13:50	L_8	u_4, u_9
weekend	08-10	03/14/10 08:12	L_8	u_4, u_{12}
weekday	08-10	03/15/10 09:47	L_1	u_1
weekend	14-16	03/20/10 15:23	L_3	u_{15}

Table 8.7: Example of Bluetooth trace with assigned location C

8.5.1 Location predictor

We use the Naive Bayesian classifier to predict the location of the person as follows.

$$L_X = \arg\max_i \{P(\nu = \nu_1, \tau = \tau_1 | L_i) P(L_i)\} \tag{8.10}$$

The Equation 8.10 outputs the most likely location L_X for the input query X. Moreover, the Equation 8.10 can be easily customized to return the top-k of the most likely locations for the input query X. In this case, L_X is the set of top-k most likely locations and we have a *top-k location predictor*.

In Equation 8.10, we do not assume the conditional independence between $\nu = \nu_1$ and $\tau = \tau_1$ since in reality a person may visit different locations in the weekdays and weekend for the same time slot. For example, in the time slot from 9AM to 11AM, the person may stay in the office in her workplace for the weekdays but she may be at home at the weekend.

8.5.2 Duration predictor

The duration predictor is constructed based on the location predictor. If the location predictor returns the top-k locations, the duration predictor will return the predicted stay duration for each of k locations.

We first define the *"stay session at the location L_k" is the continuous time period that the person stays at L_k*. In our context, since the BT scanner obtains BT records every minute, the *"stay session at the location L_k in minute" is the size of the relation Φ of consecutive tuples in relation C such that for two consecutive tuples $r_1, r_2 \in \Phi$, the difference of scan times between r_1 and r_2 is exactly 1 minute.* Let $|\Phi|$ denote the session length of one stay session of L_k.

We first use location predictor to obtain the location L_k for the input query $X = (\nu_1, \tau_1)$. Then,

we create a sub relation $C' = \sigma_{\nu=\nu_1,\tau=\tau_1,Loc=L_k}(C)$ where σ is the selection operator over the relation C [132]. Then, we calculate the session lengths for L_k from the relation C' using the above session definition. Let Γ_k be the set of all stay session lengths for the location L_k obtained from set C', $\Gamma_k = \{\Phi_1, \Phi_2, \Phi_3, ..., \Phi_{|\Gamma_k|}\}$, Γ_k forms a distribution of session lengths. Let λ_k and ξ_k denote the mean and standard deviation of this distribution. For example, the location L_1 in Table 8.7 has $\Gamma_1 = \{3, 1\}$, here $|\Phi_1| = 3$ and $|\Phi_2| = 1$ (Φ_1 consists of the first three records). The output of the duration predictor includes λ_k, and ξ_k for each location L_k.

On one hand, the duration predictor can predict how long the person stays at one location. On the other hand, since our location is inferred from the collected Wifi access point trace, the duration predictor essentially predicts the potential wireless connection opportunity. Thus, the duration predictor can be used as the fundamental building block for the design of network protocols in mobile wireless networks.

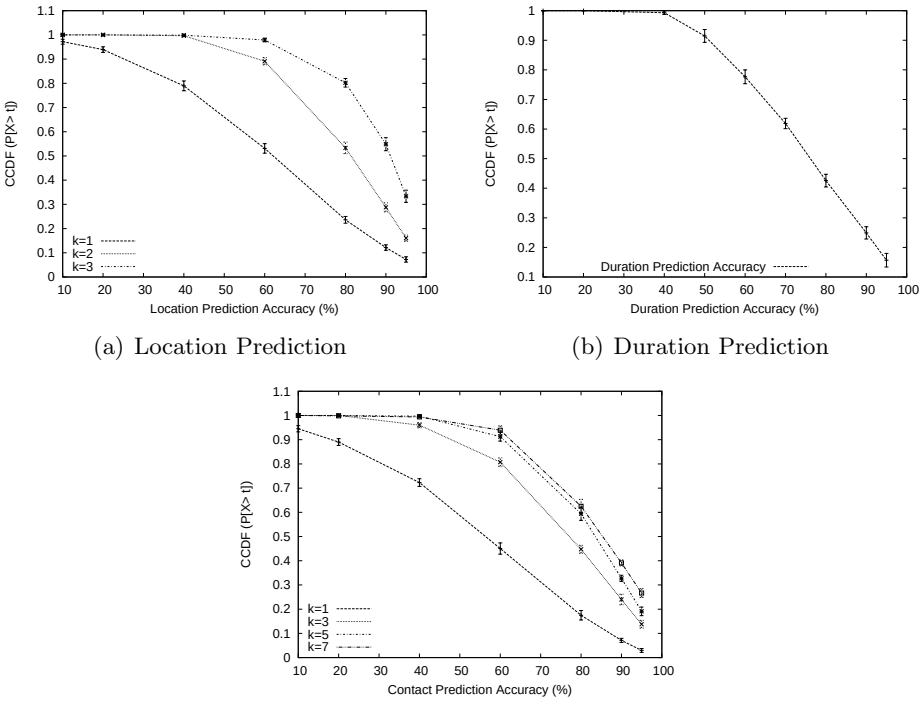
8.5.3 Contact predictor

In order to construct the contact predictor, we assume that each BT MAC scanned by the BT scanner is associated with a distinct person. As a result, each scanned BT MAC in a record of the BT trace represents a contact. We apply the Bayesian classifier to find the most likely contact the person p will have for the input $X = \{\nu_1, \tau_1\}$ as follows:

$$U_X = \arg\max_j \{P(\nu = \nu_1, \tau = \tau_1 | u_j) P(u_j)\} \tag{8.11}$$

The Equation 8.11 outputs the most likely contact U_X for the input query X. The Equation 8.11 can be easily customized to return the top-k of the most likely contacts for the input query X. In this case, U_X is the set of top-k most likely contacts and we have a *top-k contact predictor*. The contact predictor predicts the future contacts, which is crucial for the design of routing protocols and content distribution protocols in MANET and DTN.

In Equation 8.11, we do not assume the conditional independence between $\nu = \nu_1$ and $\tau = \tau_1$ since in reality a person may meet different set of people in the weekdays and weekend for the same time slot. For example, in the time slot from 2PM to 4PM, the person may meet friends in class for the weekdays but she may meet his family members at the weekend.



(a) Location Prediction

(b) Duration Prediction

(c) Contact Prediction

Figure 8.7: Correctness of Location predictor, Stay duration predictor, and Contact predictor

8.6 Evaluation

8.6.1 Evaluation settings

From March to August 2010, we had 50 joint Wifi/Bluetooth trace collected by 50 experiment participants in University of Illinois campus. Each trace is from 20 to 50 days. Let D_i be the Wifi/Bluetooth trace of the i^{th} participant in 50 participants: $D_i = W_i \cup B_i$, where W_i is the Wifi trace and B_i is the BT trace. For i^{th} participant, we first apply the UIM Clustering Algorithm over W_i to obtain locations. Then, we apply steps in Section 8.4 to assign locations to records in B_i. For the user i^{th}, let C_i be the Bluetooth trace with assigned location. We divide the relation C_i into two distinct sub set called training set Ψ_i and testing set Ω_i, in which $\Psi_i \cap \Omega_i = \emptyset$. The training set Ψ_i has 80% of records in C_i and Ω_i has 200 records randomly picked from the set $C_i \setminus \Psi_i$. We use Ψ_i to train three predictors (i.e., location, stay duration, and contact) and use Ω_i to evaluate these predictors. Each record $r \in \Omega_i$ is converted into the format of $X = \{\nu_1, \tau_1\}$ and used as the input for our predictors. We set $\theta = 0.1$, $\alpha = 60(s)$, and time slot to 2 hours in the following plots. We run the experiment 20 times (i.e., each time a new set Ω_i is created at random)

and plot the average with the 95% confident interval.

8.6.2 Correctness of predictors

8.6.2.1 Location predictor

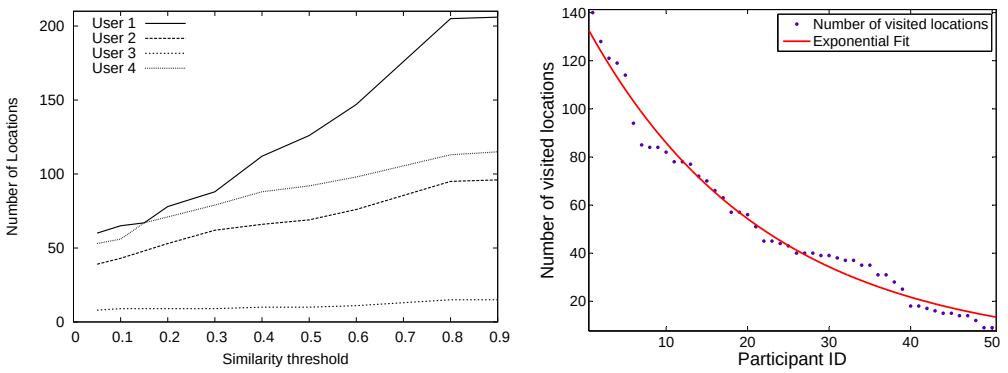
Let ${L^i}_p$ be the location predictor of the i^{th} experiment participant. For each record $r \in \Omega_i$, we use ${L^i}_p$ to predict the location of r using technique in Section 8.5.1. Let L_r be the location of $r \in \Omega_i$. Notice that we only evaluate the correctness of ${L^i}_p$ for record r whose L_r is not "Unknown". Since the predictor ${L^i}_p$ can output the top-k most likely locations, let L_{pred} be the set of predicted locations outputted by ${L^i}_p$ so $|L_{pred}| = k$. ${L^i}_p$ makes a correct prediction if $L_r \subseteq L_{pred}$.

Figure 8.7(a) shows the correctness of ${L^i}_p$ for 50 users with k from 1 to 3. When k increases, the set L_{pred} has more elements, thus the prediction is more likely to be correct, which is confirmed in this figure. Particularly, when $k = 2$, about 80% of nodes have more than 70% correct predictions. When $k = 3$, about 85% of nodes have more than 80% correct predictions. This shows that the location predictor provides an accurate location prediction.

8.6.2.2 Duration predictor

Let ${\Lambda^i}_p$ be the duration predictor of the i^{th} experiment participant. Let λ_{pred} and ξ_{pred} be the mean and standard deviation values return by ${\Lambda^i}_p$ for the input query $X = \{\nu_1, \tau_1\}$. Then, we use the definition in Section 8.5.2 to find the stay session that contains r in C_i. Notice that r should belong to an unique session since r has its own scan time and location. Let Λ_r be the length of the stay duration session that contains r in C_i.

Since stay duration may vary significantly, predicting stay duration becomes challenging. Thus, we evaluate the correctness of ${\Lambda^i}_p$ as follows: if $\lambda_{pred} - \xi_{pred} \leq \Lambda_r \leq \lambda_{pred} + \xi_{pred}$, then ${\Lambda^i}_p$ makes a correct prediction. Here, we use the top-1 location predictor whose returned location is not "Unknown". Figure 8.7(b) shows that the duration predictor performs considerably well. Particularly, 80% of nodes obtain about 60% correct prediction and 40% of nodes have about 80% correct prediction. Since the stay duration of people at one location is difficult to predict, we believe this result confirms that the duration predictor can provide an relatively accurate duration prediction.



(a) # of locations increases when θ increases (b) # of locations visited by all participants

Figure 8.8: Sensitivity of θ on location

8.6.2.3 Contact predictor

Let ${P^i}_p$ be the contact predictor of the i^{th} experiment participant. For each record $r \in \Omega_i$, let P_{pred} be the set of top-k contact returned by ${P^i}_p$, so $|P_{pred}| = k$. We evaluate ${P^i}_p$ as follows. First, let P_r be the set of contacts appearing in C_i in the same day of type ν_1 and during the same time slot τ_1 of r. Second, the predictor ${P^i}_p$ makes a correct prediction if $P_{pred} \cap P_r \neq \emptyset$. The intuition is that ${P^i}_p$ predicts that in the day of type ν_1 and during the time slot of τ_1, P_{pred} is the set of contacts, in which the person p will have at least one.

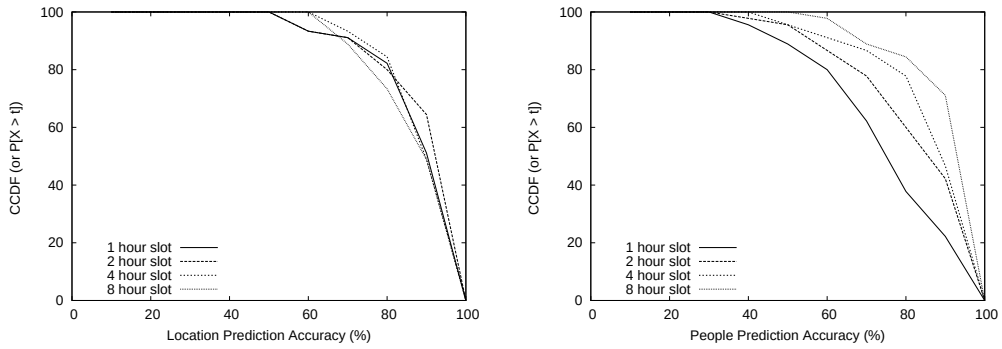
Figure 8.7(c) shows that ${P^i}_p$ performs better when k increases from 1 to 7. With k=7, about 80% of participants can obtain more than 70% correct prediction and about 60% of participants obtains more than 80% correct prediction.

In conclusion, the evaluation shows that location predictor, stay duration predictor, and contact predictor provide a considerably high prediction accuracy.

8.6.3 Sensitivity of similarity threshold θ on location

To understand the sensitivity of θ, we use the same set of 4 people in Section 8.3.6. Then, we vary θ in the range of [0.05, 0.9] and count the number of unique locations each of the four above people visited during their entire experiment periods. Figure 8.8(a) shows that the number of clusters increases nearly linearly when θ increases from 0.05 to 0.9. This result is expected since for greater value of θ, the graph G_θ is sparser, so the cluster size is smaller and the number of cluster increases.

In order to known how many locations a participant may visit during the experiment period, for each participant i^{th}, we count the number of unique locations the i^{th} visits for her entire



(a) Impact of time slot on location predictor (b) Impact of time slot on contact predictor

Figure 8.9: Impact of time slot size on performance of Prective Model

experiment period. Then, we sort the list of participants decreasingly according to their number of visited locations. Notice that for this plot, we have $\theta = 0.1$ for all participants. Figure 8.8(b) shows that the number of unique locations the participants visited during the experiment period can be fitted by an exponential function $y = ae^{bx}$ in Matlab, where $a = 135.2, b = -0.05023$. This result is important since it gives a concrete model for the number of locations for mobile nodes, which can be used for simulation purpose in mobile networking research. Particularly, instead of taking a random number as the number of locations for a mobile node in simulations, researchers might take a number following an exponential function as shown in Figure 8.8(b).

8.6.4 Impact of time slot size on predictors

We vary the size of time slot τ and evaluate the performance of our predictors.

8.6.4.1 Location predictor

Figure 8.9(a) shows that when the time slot τ varies the performance of location predictor changes slightly. This confirms that the location predictor is robust to the variation of time slot τ. The reason the location predictor stays robust to the variation of τ is that people in university campus do not move very frequently and people usually stay at one location for a long period. Thus, the impact of τ becomes less significant. Notice that for this figure, we use the top-3 location predictor.

8.6.4.2 Contact predictor

Figure 8.9(b) shows that when the time slot τ increases in size, the contact predictor performs better. For this figure, we use top-3 contact predictor. Particularly, when $\tau = 8$ hours, about 80% of nodes have 80% correct contact prediction. Meanwhile, with $\tau = 1$ hour, only 40% of nodes have 80% correct contact prediction. This is expected since for a bigger time slot τ, we have a bigger subset P_{pred} and a bigger set P_r (as discussed in Section 8.6.2.3), then the prediction is more likely to be correct.

8.7 Related Work

It is believed that people movement exhibits a high degree of repetition, in which people usually visit regular places for their daily activities [49]. As a result, there have been several efforts using past data to construct models to predict people movement.

The first class of prediction methods focused on predicting location of people movement [136, 137, 138, 139, 140, 127, 141], which essentially only answered the first question above. In particular, a large number of previous papers used the association trace between the laptop/PDA and the Wifi access points (i.e., WLAN trace) to derive and evaluate their location predictors [140, 127, 142, 143]. However, there was a fundamental weakness of using WLAN trace in constructing location predictor. The reason was that the laptop user did not always turn on the laptop and did not carry it with her all the time. Moreover, a normal laptop user usually turned on her laptop and left it on her office desk when she was doing other things (e.g., had lunch with friends, had meetings with colleagues, or went to exercise at the gym). So, the collected associations of laptops and the Wifi access points could be used to understand the wireless usage rather than to predict the location of people. Other previous projects used cellular data trace to construct the location predictor [138, 137, 139, 131], in which the location was inferred from the cellular ID of the cellular base station. However, since the transmission range of the cellular base station was ranging from several hundred meters (e.g., 500 m) to kilometers (e.g., 30 km), the location predictor derived by this inferred location might not provide needed fine granularity and accuracy.

The second class of prediction methods answer the first two questions above by providing predictions for only the stay duration [144] or both location and stay duration at the location [126]. The stay duration of commuters is predicted so that the source of media content will be chosen from all

candidate commuters traveling the same train of the subway line [144]. However, the stay duration was calculated based on the assumption that commuters check-in the subway line at the same station and at the same time will stay in the same train. This assumption might not be true in reality [144]. In their paper, Lee and Hou modeled user mobility by a semi-Markov process and devised a timed location prediction algorithm that predicted the future access point (i.e., location in the paper's context) of the user and the association duration [126]. Since the model was constructed and evaluated by WLAN trace, it suffered from the same fundamental weaknesses as discussed in the previous paragraph. However, this paper has been the only method so far which could provide predictions for both location and stay duration.

The aforementioned works only answered the first two questions about location and stay duration of people movement since they lacked of ad hoc contact traces, which could be used to infer social contacts and answer the third question. Recently, there have been several projects collecting ad hoc contact traces using portable experiment devices such as iMote, cellphone, PDA [113, 114, 115, 116, 117, 118]. Due to the limitation of battery and the hardware capability of the experiment devices, only the Bluetooth ad hoc contacts were collected by these projects. Moreover, the scale of these experiments is much smaller in the number of participants and shorter in the experiment duration than those of WLAN experiments in [140, 127]. More importantly, these collected traces did not have the location information and thus could not be used to answer the first two questions about location and stay duration.

In this chapter, we present the Jyotish method to construct a predictive model of people movement from the joint Wifi/Bluetooth trace to predict the future of location, stay duration, and contact of people movement.

8.8 Conclusion

The Jyotish method provides an efficient solution to construct a predictive model of people movement from the joint Wifi/Bluetooth trace. The evaluation over the real Wifi/Bluetooth trace collected by 50 participants shows that the constructed predictive model predicts location, stay duration, and contact with a considerably high accuracy.

To the best of our knowledge, the Jyotish method is the first method to derive a predictive model from a real joint Wifi/Bluetooth trace. The constructed model is also the first to provide prediction of location, stay duration, and contact altogether. Since future knowledge of people movement is

fundamental for numerous domains such as computer network, HCI, social science, urban planing, environmental science, etc., we thus believe Jyotish method and the constructed predictive model are widely applicable.

CHAPTER 9

COMFA: EXPLOITING REGULARITY OF PEOPLE MOVEMENT FOR DATA FORWARDING IN COMMUNITY-BASED DELAY TOLERANT NETWORKS

9.1 Introduction

It is well known that the daily movement of people exhibits a high degree of repetition in which people usually stay at regular places for their daily activities [145]. It is also believed that people usually belong to certain social communities [146], where they have social relationships and social contact with other community members. Therefore, exploiting the regular movement patterns of people within the same community to improve content distribution has drawn significantly attention from research community [147, 148, 149].

In Chapter 7, we presented the UIM scanning system to collect the joint Wifi/Bluetooth trace. In Chapter 8, we exploit the regular movement patterns found in the collected Wifi/Bluetooth trace to construct a predictive model of people movement, including location predictor, stay duration predictor, and contact predictor. In this chapter, we leverage the contact predictor in the design of a new content distribution named COMFA [1]. COMFA exploits the regularity of encounter pattern (or contact pattern) of people within the same community to maximize message delivery probability while preserving message delivery deadline. Particularly, COMFA divides the time dimension of social contacts into a more detailed level of type of day and time slot to capture the regularity of encounter pattern. Then, COMFA uses the contact predictor in Chapter 8 to calculate the contact probability between a pair of community members to construct the routing table. The routing table is then used to estimate the message delivery probability and select the message forwarder in the routing process. We evaluate and compare COMFA with Prophet routing [47] and Epidemic routing [17] protocols over the real movement trace of 100 mobile nodes collected by 9 participants in the same research group in University of Illinois from March 01 to March 20, 2010. The evaluation results show that COMFA considerably outperforms other alternatives by

[1] COMFA stands for COMmunity-based data ForwArding protocol.

improving the delivery time and reducing message overhead.

This Chapter is organized as follows. We first present the Community-based Delay Tolerant Network in Section 9.2. Then, we present the design of COMFA in Section 9.3. In Section 9.4, we compare the performance of COMFA with Epidemic routing and Prophet routing protocols. Finally, we present the related work in Section 9.5 and conclude the Chapter in Section 9.6.

9.2 Community-based Delay Tolerant Network

According to Palla et al., most real networks contain parts in which the nodes are more highly connected to each other than to the rest of the network, and these sets of nodes are usually called communities [150]. In this chapter, we focus on the community formed by nodes that have the social relationship and stay in the same geographical area since the co-location of mobile nodes is crucial for peer-to-peer data forwarding. This type of community exists virtually everywhere in the real world. For example, students in the department of Computer Science can form a community called "CS Student Community", professors in the department of Computer Science can form a community called "CS Professor Community", and researchers in the same research group can form a small community as well.

Since the members of the community stay in the same geographical area, they have more frequent physical contact/interaction with each other. As presented in Section 7.5, people exhibit a high degree of regularity in their daily movements. Thus, these contacts among community members are also regular. Since the topology of the network formed by mobile nodes carried by the community members depends fully on the regular movement pattern of community members, the wireless connectivity between a pair of mobile nodes is intermittenly connected and also regular. In our context, the network formed by these mobile nodes is called Community-based Delay Tolerant Networks.

In this chapter, we consider present a routing solution of the message m between two community members s and r, in which s is the sender and r is the receiver of m. We exploit the regularity of contacts among community members to expedite message forwarding in the network. Henceforth, we use the terms "community member", "mobile node", "node", "person", "Bluetooth MAC", "BT MAC" interchangeably.

9.3 COMFA: Community-based Data Forwarding Protocol

This section presents the design of COMFA protocol. First, we present the system model and design objective of the protocol. Second, we present how to bootstrap COMFA and update contact counter C_n. Third, we present how to use the contact predictor to construct the routing table. Finally, we present how the message is forwarded among mobile nodes in the network.

Name	**Description**
m	Message transmitted in the network
s	Mobile node, the sender of m
r	Mobile node, the receiver of m
n	Mobile node in the network
D_m	Delivery deadline of the message m
T_m	Time at which n delivers m to r
${P^n}_t$	Probability that n delivers m to r during $[t, D_m]$
R_n	Routing table of n
ρ	Time slot size (i.e., $\rho \in [1, 2, 3, 4, ..., 24]$)
ν	Type of day
m_ν	Type of day when m is routed from s to r
τ	Time duration (i.e., of a time slot)
${p^c}_{u_i}$	Contact probability between n and node u_i
U_n	Set of mobile nodes n has met so far
C_n	Contact counter, $C_n = \|U_n\|$

Table 9.1: Notations used for design of COMFA

9.3.1 System model and design objective

9.3.1.1 System model

We consider a network formed by the Peer-to-Peer connectivity among mobile devices carried by the members of the same community. Each node in the network has a mandatory P2P communication interface such as Wifi, Bluetooth, etc. Node mobility is regular and nodes have regular contacts. The data message forwarded among mobile nodes in this network in the format of text message or short video clips.

9.3.1.2 Design objective and protocol overview

Table 9.1 shows the notations used by the COMFA protocol. For a message m, when m is sent out by the the sender s, s sets the delivery deadline D_m. Our objective is to route the message m from s to r by the delivery deadline D_m. We formulate the COMFA protocol as the optimization problem, which maximizes the delivery probability of the message m from s to r and meets the delivery deadline D_m:

$$\max \quad {P^n}_t$$

$$s.t.\ T_m \leq D_m \tag{9.1}$$

COMFA obtains the objective function for the constraint of Equation 9.1 as follows. Let C_n be the number of unique mobile nodes the node n has met so far. At the time t in the routing process, assuming that m is carried by a mobile n_1 and n_1 meets n_2, if ${P^{n_2}}_t > {P^{n_1}}_t$, then n_1 transmits m to n_2, which will carry and forward m towards r. n_2 will then find a node with better delivery probability to transmit m towards r. If ${P^{n_2}}_t = {P^{n_1}}_t$, the value of C_n is used to break tie and select the better forwarder. That means, if the delivery probabilities are equal, the node with a greater number of unique contacts will be chosen as the next message forwarder since it has a higher probability to reach the receiver r. Our COMFA protocol keeps only a single copy of the message m among all the nodes in the network during the routing process to reduce message overhead. Since all the nodes in the network perform the same way, we are maximizing the delivery probability of m by the delivery deadline D_m. Constraint of Equation 9.1 is satisfied because T_m is taken into account in the calculation of ${P^n}_t$. For a mobile node n, we have $0 \leq {P^n}_t \leq 1$, which is derived from the routing table R_n of n. The routing table R_n is constructed by using the contact predictor [46]. Since the mobile nodes are within a community, it is likely that the receiver r has regular contacts with members in the same community. These regular contacts will help in forwarding m to r. In the following sections, we present the detail of COMFA protocol.

9.3.2 Bootstrapping COMFA

This section presents how the protocol COMFA bootstraps. We focus on the scenario where the mobile user carries his mobile phone during his daily activities. The phone runs the Bluetooth scanner that captures the Bluetooth MAC addresses of Bluetooth-enabled devices in the proximity of the phone. In other words, the phone captures the social contacts between this mobile user

and his friends. The Bluetooth scanner stores the collected MAC addresses in the memory of the phone. Since the mobile user has regular movement pattern and meets regular set of people in his community, the phone captures the regular social contacts between this mobile user and his friends. After several weeks, the phone captures enough the regular social contact patterns of this mobile user and then it starts using contact predictor to construct the first routing table for message forwarding. Notice that after the first routing table is constructed, the Bluetooth scanner continues to run and collects more social contacts. The routing table will be updated periodically later by using the newly collected social contacts since the mobile user continues his regular movement behavior. For example, the routing table can be updated every week. As a result, the more amount of collected social contact the Bluetooth scanner has, the more accurate the routing table is.

9.3.3 Updating number of unique contacts C_n

Besides the routing table, mobile node n needs to keep the number of unique contacts C_n that n has so far. In order to update C_n, n keeps a set U_n of known contacts, anytime n encounters a new node n_1, n adds to U_n. We have $C_n = |U_n|$. Basically, C_n represents the total number of unique contacts n has had so far. C_n can be used as an indicator of future contacts since in reality a person who has made many contacts in the past would likely make many contacts in the future. C_n is similar to the node degree of the social graph used by the previous works to select the best message forwarder [147, 148, 149].

9.3.4 Constructing routing table

With COMFA protocol, each node n has a routing table R_n, which is used to forward the message m during the routing process. R_n is basically a table with multiple columns and rows, which is similar to a relation in the relational algebra. To clarify the presentation in the following sections, we use Relational Algebra [132] to represent and manipulate the routing table R_n. Henceforth, we use the terms routing table and relation interchangeably.

We exploit the regularity of people movement and the contacts among community members to construct the routing table R_n by classifying time into type of day and time slot. Particularly, we use the contact predictor to calculate the probability that n has contact with one particular person for a type of day and during a time slot. The probability will be used to select the best

message forwarder during the routing process. The intuition of classifying time into type of day and time slot is that people usually follow their scheduled routines for the daily activities and thus a person may meet the regular set of people for a specific time period in a particular day. Also, the movement behavior of people may be different for the weekday and weekend.

ν	τ	u_1	u_2	u_3	u_4
weekday	(08:00,09:00]	0.4	0.2	0.1	0
weekday	(09:00,10:00]	0.1	0	0.6	0
weekday	(10:00,11:00]	0	0	0.5	0.6
weekday	(12:00,13:00]	0	0	0.8	0
...	...	...	...	...	...
weekend	(08:00,09:00]	0	0.8	0	0
weekend	(09:00,10:00]	0	0	0.2	0.7
...	...	...	...	...	...

Table 9.2: Example of routing table (or relation) R_n for time slot size $\rho = 1$ hour.

Let $U_n = \{u_i : 1 \leq i \leq |U_n|\}$ be the set of all people the node n has met so far, R_n is a relation of $|U_n| + 2$ attributes (or the routing table R_n has $|U_n| + 2$ columns). The relation R_n has multiple tuples (or the routing table R_n has multiple routing entries). Table 9.2 shows an example of a relation (or routing table) R_n. Each tuple represents an entry in the routing table, in which the first two values of the tuple are the type of day ν and time slot τ. The last $|U_n|$ values of a tuple are the contact probabilities between n and the corresponding mobile node for that type of day and during that time slot. In this table, for the type of day *weekday* and during the time slot $(8, 9]$ the node n meets the person u_1 with the contact probability of 0.4. One tuple in this relation (or one row in the table) is $< weekday, (08:00, 09:00], 0.4, 0.2, 0, 0 >$. Let ${P^n}_p$ be the contact predictor of the mobile node n, which is constructed in Chapter 8. In this section, we leverage the contact predictor to construct the routing table R_n.

First, we create the set of queries $\chi = \{X_1, X_2, ..., X_k, ..., X_{|\chi|}\}$ in which $X_k = \{\nu_k, \tau_k\}$, $\nu_k \in \Upsilon = \{weekday, weekend\}$, and $\tau_k \in \Sigma$. The set Σ is constructed based on value of the time slot size ρ. For example, if $\rho = 1$, then a day has 24 slots and $\Sigma = \{(00:00, 01:00], (01:00, 02:00], ..., (23:00, 24:00]\}$; if $\rho = 2$, then a day has 12 slots and $\Sigma = \{(00:00, 02:00], (02:00, 04:00], ..., (22:00, 24:00]\}$, and so on.

Then, the set of queries χ is used as input for the contact predictor ${P^n}_p$. For each query $X_k \in \chi$, ${P^n}_p$ returns a tuple $e_k =< \nu_k, \tau_k, {p^c}_{u_1}, {p^c}_{u_2}, ..., {p^c}_{u_j}, ..., {p^c}_{u_{|U_n|}} >$, in which ${p^c}_{u_j}$ is the

contact probability between n and u_j for the type of day ν_k and during the time slot τ_k. Here, $1 \leq j \leq |U_n|$ and $0 \leq {p^c}_{u_j} \leq 1$. Notice that, we use the contact predictor to *proactively* calculate the contact probability between n and other nodes in the community (that n has met so far). This approach will reduce the forwarding delay of the message since the probability is available in the routing table for selecting the best message forwarder.

The set of all returned $|\chi|$ tuples from ${P^n}_p$ forms the relation $R_n = \{e_1, e_2, ..., e_{|R_n|}\}$. The relation R_n has $|\chi| = |\Upsilon| \cdot |\Sigma|$ tuples, in which $|\Upsilon| = 2$ and $|\Sigma|$ depends on ρ. For example, with $\rho = 1$, R_n has 48 tuples.

9.3.4.1 Extensible routing table

When time goes on, n may encounter new mobile nodes in n's community. Therefore, R_n should be an extensible routing table that can grow when more mobile nodes are added into the set U_n. In other words, the table R_n should have more columns when n meets new community members.

9.3.4.2 Finer grain routing table

Classifying time into type of day and time slot does capture the regularity of people movement. However, there are cases where people move in a more strictly repeated schedule. For example, a student may always attend a class from 10AM to 11AM every Tuesday, a professor may always give a lecture from 3PM to 4PM every Friday. For these cases, we can classify time into a finer granularity, for example time can be classified into day of week such as Monday, Tuesday, etc. (rather than type of day $\{weekday, weekend\}$), and time slot of size ρ. With this finer grain time classification, we have a finer grain routing table with more routing entries. For example, for $\rho = 1$, we have $7 \cdot 24 = 168$ entries in the routing table R_n. However, there are two important tradeoffs. First, this finer classification requires more collected data (or a longer training time) to calculate a more accurate contact probability to construct the routing table. Second, this finer classification works better only for people whose movements are strictly repeated; as a result, for people with a more relaxed movement patterns, a finer classification may result in a worse routing decision.

9.3.5 Message forwarding decision

Given the routing table (or the relation) R_n of the mobile node n, R_n is used to route the message m from the sender s to the receiver r. This section presents how the node n_1 decides whether it should transmit m to another node n_2 when n_1 and n_2 encounter. Notice that since n_1, n_2, r are in the same community, they encounter each other with high probability. However, as we present later in this section, with the use of C_n in the forwarding process, COMFA works even if n_1 and n_2 have not met the receiver r before.

At time t, assuming that m is carried by a node n_1. When node n_1 meets another node n_2, n_1 needs to decide whether n_1 will transmit the message m to n_2 so that m is forwarded to r in an efficient fashion. Notice that only one copy of the message m is kept among mobile nodes in the network during the routing process. So, if n_2 is a better forwarder for m, n_1 will transmit m to n_2 and then n_1 does not carry m anymore.

As presented in Equation 9.1, during the routing process we prefer the node, which provides a higher delivery probability of the message m to the receiver r. Therefore, at time t when n_1 meets n_2, n_1 calculates its delivery probability ${P^{n_1}}_t$, which is the probability n_1 delivers m to r during the time period $[t, D_m]$. Similarly, n_2 calculates its delivery probability ${P^{n_2}}_t$. The two nodes n_1 and n_2 then compare their delivery probabilities. If ${P^{n_1}}_t < {P^{n_2}}_t$, m is transmitted from n_1 to n_2 and then n_2 becomes the only carrier of m in the network from that time. If ${P^{n_1}}_t = {P^{n_2}}_t$, C_{n_1} and C_{n_2} are compared and the node with a greater size of C_n will be the next forwarder. The intuition is as follows. When the two nodes n_1 and n_2 have equal delivery probabilities of m, the next forwarder is the one that has met more nodes in the past since that node will likely meet more nodes in the future (including the receiver), and thus deliver m with a higher probability. The next question is how to calculate ${P^n}_t$ for a mobile node n.

${P^n}_t$ is calculated using the relation R_n in a totally decentralized fashion as follows. When m is sent at the sender s, s obtains the type of day $m_\nu \in \{weekday, weekend\}$ and attaches m_ν with m. At time t, m_ν is retrieved by n and n creates a relation E by performing a selection operation over R_n as follows:

$$E = \sigma_{\varphi=\{\nu=m_\nu, \tau \in \Sigma'\}} R_n \tag{9.2}$$

In Equation 9.2, φ is the condition of the selection operation over R_n, which basically filters out irrelevant tuples. In particular, relation E consists of only the tuples of R_n that have the type of day m_ν and the time slot in the set Σ', which is created as follows. For a time slot τ_i, let ${\tau^s}_i$ be the

starting time of τ_i and ${\tau^e}_i$ be the ending time of τ. For example, if $\tau_i = (08:00, 10:00]$, then we have ${\tau^s}_i = 08:01$ and ${\tau^e}_i = 10:00$. For the duration $[t, D_m]$, we have $\Sigma' = \{\tau_i : {\tau^s}_i \geq t, {\tau^e}_i \leq D_m\}$. For example, in Table 9.2, if $m_\nu = weekday$, $t = 08:00$, and $D_m = 11:00$, then the relation E consists of the first three tuples as shown in Table 9.3.

ν	τ	u_1	u_2	u_3	u_4
weekday	(08:00-09:00]	0.4	0.2	0.1	0
weekday	(09:00-10:00]	0.1	0	0.6	0
weekday	(10:00-11:00]	0	0	0.5	0.6

Table 9.3: Relation E with $m_\nu = weekday$, $t = 08:00$, and $D_m = 11:00$.

Given the relation E with all tuples for the type of day m_ν and for all time slots during the period $[t, D_m]$, we then create a relation S by performing a projection over E as follows:

$$S = \pi_{u_i = r}(E) \tag{9.3}$$

In Equation 9.3, we obtain the relation S by extracting the attribute $u_i = r$ from the relation E. In other words, the table S has only one column, which consists of the contact probabilities between n and receiver r obtained by projecting the column of receiver r in relation E. For example, in Table 9.2, if $m_\nu = weekday$, $t = 08:00$, $D_m = 11:00$, and $r = u_3$, then $S = \{0.1, 0.6, 0.5\}$ as shown in Table 9.4. Since S has only one column, we use the term "set S", "relation S", and "table S" interchangeably in this section. Formally, we have $S = \{p_1, p_2, p_3, ..., p_{|S|}\}$, where $0 \leq p_j \leq 1$, $1 \leq j \leq |S|$. p_j represents the contact probability between n and r during the time slot j^{th} and $p_j = 0$ means n and r have no contact during the j^{th} time slot.

$u_3 = r$
0.1
0.6
0.5

Table 9.4: Relation S with $m_\nu = weekday$, $t = 08:00$, $D_m = 11:00$, and $r = u_3$.

Notice that the order of elements in S corresponds to the order of the time slots in the relation E. That is, p_1 is the contact probability between n and r during the first time slot after time t,

and $p_{|S|}$ is the contact probability between n and r during the last time slot before D_m. The set S is then used to calculate ${P^n}_t$, the probability that n delivers m at r during the period $[t, D_m]$ as follows:

$$ {P^n}_t = \sum_{i=1}^{|S|} {p^d}_i \tag{9.4} $$

In Equation 9.4, ${p^d}_i$ is the delivery probability that n successfully delivers m to r at the time slot i^{th}. ${p^d}_i$ is calculated based on contact probability in the set S as follows. We observe that node n only delivers m to r at the time slot i^{th} if n fails to deliver m to r in the first $(i-1)$ time slots and n successfully delivers m to r at time slot i^{th}. This happens only if n and r have no contacts during the first $(i-1)$ time slots and n and r have contact in the i^{th} time slot. So, we have:

$$ {p^d}_i = \{\prod_{j=1}^{i-1}(1-p_j)\} \cdot p_i \tag{9.5} $$

The Equation 9.5 is used to calculate the delivery probability for each time slot i, and then ${P^n}_t$ is calculated accordingly in Equation 9.4.

9.3.5.1 Update overnight message

During the routing process, if n finds that the value of m_ν extracted from the message m is different from the type of day of the current time, then node n replaces the value of m_ν with the current type of day and attaches the new value of m_ν with m before forwarding m to another node. This step is needed if the message m is routed overnight from weekday to weekend and via versa.

9.3.5.2 Discussion

When two nodes n_1 and n_2 have contact at time t, ${P^{n_1}}_t$ and ${P^{n_2}}_t$ are calculated by Equation 9.4 and compared, the node with the greater value of delivery probability is chosen as the next forwarder of m. If the two delivery probabilities are equal, C_n will be used to select the better message forwarder. Doing this, we obtain the objective function of the optimization problem in Equation 9.1 by maximizing the delivery probability of m in the routing process. Notice that the routing decision is made in a totally distributed manner since ${P^n}_t$ is calculated using the local routing table R_n of the node n and does not rely on information from other nodes. Also, since the

set E is created based on the message delivery deadline D_m, calculation of ${P^n}_t$ takes into account the message delivery deadline D_m to satisfy the constraint of Equation 9.1. Therefore, we believe COMFA provides a robust and efficient routing solution for community-based DTNs.

9.4 Evaluation

9.4.1 Settings

We select a set Δ of 9 Bluetooth traces collected by 9 participants in the same research group in the department of Computer Science, University of Illinois from March 1, 2010 to March 19, 2010. These 9 traces are extracted from the data set D_1 in Table 7.1. The set Δ gives us 500 unique BT MACs. From this set of 500 Bluetooth MACs, we select a subset $D \subset \Delta$ of 100 Bluetooth MACs, where each BT MAC $b \in D$ encounters at least 3 out of 9 participants in the set Δ during the experiment period from March 01, 2010 to Mach 19, 2010. Since the participants in Δ come from the same research group, D represents contacts of a realistic community.

Scan Time	**Set of Scan Bluetooth MACs**
03/09/10 09:15	u_1, u_3
03/09/10 09:16	u_1, u_3
03/09/10 09:17	u_1
...	...
03/09/10 13:50	u_4, u_9
...	...
03/14/10 08:14	u_1, u_3, u_8

Table 9.5: Example of Bluetooth trace B collected by one experiment phone

Table 9.5 shows an example of a Bluetooth trace B collected by our UIM system. Here, each row in the table is the result of one Bluetooth scan. The set D is created by projecting the column "Set of Scan Bluetooth MACs" from B. B is also used to infer contacts among mobile nodes in D to construct the routing table as discussed in Section 9.3. The scan time is converted to the corresponding type of day and time slot.

We then use D to evaluate the performance of COMFA forwarding and compare its performance with two other protocols: Epidemic routing [17] and Prophet routing [47]. Epidemic routing is basically a flooding-based scheme which floods the message from the current forwarder to any new

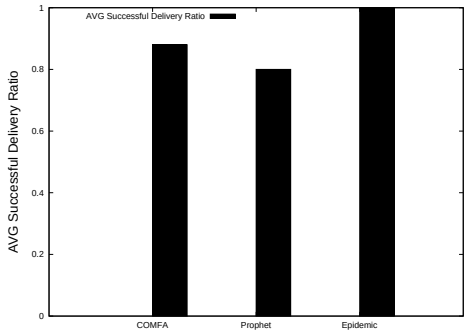


Figure 9.1: Comparison of Average Successful Delivery Ratio

encountered nodes, which have not carried the message. This protocol provides a high delivery probability; however, it incurs a high message overhead. Meanwhile, Prophet uses the summarized contact probability between pairs of nodes for the entire time period rather than classifying time into type of day and time slot like what COMFA does. Then, the forwarding decision of Prophet is made based on this long-term summarized contact probability. Since the calculated probability is not for type of day and time slot, it does not capture the detailed contact information and it may not exploit fully the regularity of people movement in forwarding message.

To compare the performance of COMFA with Epidemic and Prophet protocols, we create a set $\Gamma = \{(s_i, r_i) : 1 \leq i \leq 100\}$, in which s_i is the sender, r_i is the receiver, and $|\Gamma| = 100$. For a pair of (s_i, r_i), we first select $s_i \in \Delta$ at random (notice that we have $|\Delta| = 9$). Then, we select a random day $d \in [03/01, 03/19]$ during the experiment period, let ${D^d}_{s_i}$ be set of BT records collected by s_i during the day d, so s_i exists in all records of ${D^d}_{s_i}$. Notice that records of ${D^d}_{s_i}$ are sorted increasingly according to the scan time (similar to the Table 9.5). Let R be the set BT MACs extracted from the last 30 records in ${D^d}_{s_i}$, formally $R = \{u_i : u_i \in b_j, b_j \in {D^d}_{s_i}, |{D^d}_{s_i}| - 30 \leq j \leq |{D^d}_{s_i}|\}$, in which b_j is a record in the Bluetooth trace as shown in Table 9.5. Finally, the receiver r_i is selected at random from the set R.

There are two motivations to select the receiver r_i as above. First, as presented in a survey of 300 computer science faculty members and students [118], the forwarding delay that people can tolerate is from one to several hours, depending on the delay-tolerant networking applications and services [118]. We also observe that since the set D consists of BT MACs collected by participants in the same research group, nodes in D usually have contacts during the office hour period from 8AM to 6PM. So, by selecting r_i from the last 30 records of ${D^d}_{s_i}$, we basically set the deadline for the message transmission at the end of the office hour (i.e., around 6PM) and thus the delivery

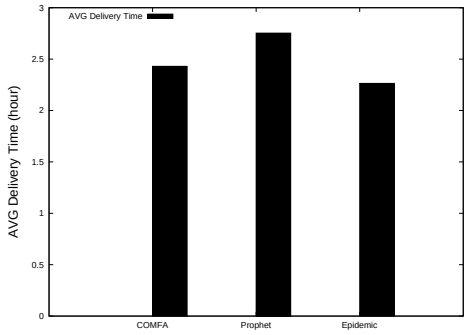


Figure 9.2: Comparison of Average Delivery Time for Delivered Messages

deadline is in the range of 8 to 10 hours. We believe this delivery deadline is reasonable in reality. Second, r_i is selected based on the set of ${D^d}_{s_i}$ means that applying Epidemic routing, s_i always can deliver the message m to r_i. Due to the construction of our testing set Γ, we expect that Epidemic routing will have 100% delivery ratio and a shortest delivery time since it floods the network with the message. We then use Epidemic routing as the base line in our performance comparison. In order to compare the performance of COMFA with Epidemic and Prophet routing protocols, we use three metrics Average Successful Delivery Ratio, Average Delivery Time, and Average Message Overhead.

9.4.2 Results

Figure 9.1 shows that Epidemic routing outperforms UIM routing and Prophet routing in terms of Average Message Delivery Ratio since Epidemic routing is essentially a flooding-based routing protocol. In Epidemic routing, a node n_1 which did not receive the message m will take a copy of the message m once n_1 has a contact with n which is carrying m. Since the receiver is chosen in the same day of the sender, epidemic routing will always can deliver the message to the receiver. Meanwhile, UIM obtains 89% of successful delivery since it exploits the regularity of people movement forward the message, which is not exploited by Prophet. This figure also shows that Prophet only obtains 78% successful delivery.

Figure 9.2 shows that Epidemic routing obtains the shortest delivery time of 2.26 hours. Correspondingly, COMFA obtains 2.42 hours and Prophet obtains 2.75 hours. Notice that this figure is only for delivered messages. In other words, 89% of messages delivered by COMFA and 78% of messages delivered Prophet are taken into calculation for this plot.

Figure 9.3 shows that while COMFA needs 2.2 messages, Prophet needs 2.65 messages to send

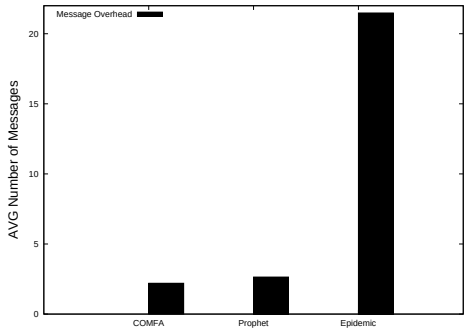


Figure 9.3: Comparison of Average Message Overhead

one message m from the sender to the receiver (or to confirm that m misses the deadline), Epidemic needs 21.5 messages to send one message m successfully. So, Epidemic incurs 10 times of message overhead. For this plot, all messages are taken into consideration even if the transmission misses the delivery deadline.

In conclusion, the comparison shows that COMFA outperforms Prophet in all metrics: Average Delivery Ratio, Average Delivery Time, and Average Message Overhead. Moreover, COMFA outperforms Epidemic in Message Overhead metric.

9.5 Related Work

There have been two major approaches [2] to improve message forwarding of routing protocols in DTN research, including: (1) learning the past contact patterns resulted from regular patterns of people movement to predict future contacts, and (2) leveraging the social contacts among members within the same community.

For the first approach, the regular patterns from the previous contact history of mobile nodes was extracted and learned to predict the future contacts for data forwarding [47, 151, 152, 153, 154, 155, 156, 157, 158]. The selected forwarder was the one who was likely to deliver the message or bring the message closer to the receiver. The best forwarder of the message is selected based on the contact probability between node pairs. However, the contact probability calculated by previous works [47, 151, 152, 153, 154] essentially represented the summarized contact between a node pair, which was similar to the contact summary oracle in the seminal DTN routing paper [159]. By using the summarized contact information, previous works did not fully leverage the regularity of

[2] Besides other major approaches

people movement for message forwarding [160]. This is because people usually follow their daily schedules and thus they might repeatedly meet each other during a certain time period of a day, rather than everyday. For example, if the summarized contact probability between a node n_1 and a node n_2 is very low, n_1 is not selected as the message forwarder for n_2. However, in reality n_1 might have contacts with n_2 only during a certain time period τ of a day. As a result, although n_2's summarized contact probability with n_1 is low, the contact probability of n_1 and n_2 during the period τ might be still higher than those of other nodes during this period τ. In this case, n_2 should be considered as the potential message forwarder of n_1 for the period τ.

The second approach in DTN forwarding was to leverage the social relationships, social contacts, and centrality of people within the same community to expedite data forwarding among community members [147, 148, 149, 161, 162, 163]. Since in reality the members of the same community have a rich set of contacts during their daily activities, the forwarding performed by community members can be expedited by this set of contacts. In these previous works, the community was viewed as a graph of vertexes, which were community members. The edge between a vertex pair exists if the two corresponding community members have social relationship. Each vertex has a centrality (i.e., node degree), which measures the popularity of a member in his community. The node with the highest centrality is selected as the best message forwarder. Similar to the summarized contact probability in the first approach, centrality essentially is a compression of the time-related contact information. In other words, regardless how often a pair of community members were in contact, there exists only one edge between them. As a result, like the first approach, the second approach did not fully exploit the regular movement patterns of community members for data forwarding.

It is clear that the summarized contact probability and centrality of mobile nodes did not exploit fully contact information in data forwarding. In this chapter, we calculate the detailed contact probability and explore the regularity of contact pattern among community members to expedite data forwarding in DTN.

9.6 Conclusion

COMFA exploits the regular movement patterns of people within the same community to forward data messages. Moreover, COMFA provides a distributed solution to calculate and utilize detailed contact probability for different types of day and time slots. We evaluate and compare COMFA with Epidemic and Prophet routing protocols over the real Bluetooth contact trace and results

show that COMFA outperforms other alternatives and efficiently deliver the data messages by the delivery deadline. People make regular social contacts in their daily activities. By exploiting regular movement patterns in data forwarding, COMFA is widely applicable.

CHAPTER 10

CONCLUDING REMARKS

10.1 Conclusion

In this thesis, we present a framework to characterize and leverage people movement for content distribution in mobile P2P networks. Particularly, we study the Schelling behavior and repetitive behavior of people movement.

We first observe that the grouping behavior of people, who share the mutual content interest and move towards the same Point of Interest, exists naturally in numerous real-world scenarios. We also observe that the grouping behavior of people movement is similar to the segregation behavior (or Schelling behavior) presented by Thomas Schelling, a Nobel prize winner in economics. We then characterize and validate the Schelling behavior by simulating the people movement on three real Google maps and by modeling people movement using the Mobius modeling tool. The validation study shows that by following the Schelling behavior, mobile nodes create a dynamic coalition P2P network, in which the size of coalitions formed by co-located mobile nodes increases at the closer distance from the Point of Interest. More importantly, the study shows that the coalition pattern follows an exponential function with respect to the distance from the Point of Interest. We then exploit the characteristics of Schelling behavior in the design of three content distribution protocols: COADA, iShare, and DENTA. We evaluate and compare our protocols with other state-of-the-art content distribution solutions. The evaluation results show that our protocols outperform other alternatives and provide improvements of content distribution in mobile P2P networks.

In the second part of thesis, we study the repetitive movement behavior of people and leverage the regularity of people movement for improvements of content distribution. We present a new method to collect people movement trace on the cellular phones. Applying this method, we implement the UIM scanning system to collect Wifi trace and Bluetooth trace. The UIM system then is deployed on 123 Google Android phones carried by faculties and students at the University of

Illinois campus from March 2010 to August 2010. The collected joint Wifi/Bluetooth trace is then used to characterize people movement and the characterization study confirms that people exhibit a high degree of regularity in their daily movements. We then propose the Jyotish method to construct a predictive model of people movement from the joint Wifi/Bluetooth trace. Using the Jyotish method over our collected joint Wifi/Bluetooth trace, we construct a predictive model to predict location, stay duration at the location, and social contact of people movement. Finally, we leverage the constructed predictive model to design the COMFA content distribution protocol for the Community-based Delay Tolerant Networks. We evaluate and compare the performance of COMFA with Epidemic routing and Prophet routing over the real movement trace collected by the UIM system. Evaluation resutls show that COMFA outperforms the other alternatives considerably.

10.2 Future Work

In this thesis, we has explored the improvements of content distribution for the dynamic coalition P2P networks of a single Point of Interest. Meanwhile, in its general form, the dynamic coalition P2P network includes multiple Points of Interest as shown in Figure 3.2. In the future, new content distribution protocols for the general dynamic coalition P2P networks should be considered. Also, current protocols (i.e., COADA, iShare, DENTA) are only designed for mobile nodes , which have a single content interest at one time. New content distribution protocols are needed for mobile nodes with multiple content interests. The combination of multiple content interests and multiple Points of Interest are also potential to be explored.

Currently, the UIM system consists of two scanners capturing Bluetooth and Wifi MACs of devices in proximity of the experiment phones. The scanning frequency is fixed to 60(s) for Bluetooth scanner and 30(minute) for Wifi scanner. In the future an adaptive scheme might be deployed to optimize scanning frequency and conserve phone battery. One possibility is to use motion sensor to detect the movement and activity of mobile users and then decide the scanning action accordingly.

Our current design of COMFA protocol takes into account only the social contact in P2P data forwarding. Since UIM data set provides a richer set of context information including location, stay duration, and social contact, a new content distribution protocol that leverage all these information will be useful.

The data set collected by our UIM system provides a rich set of Wifi and Bluetooth traces.

We believe the collected data set can be extensitvely used for future research. For example, Wifi trace can be used for localization using Wifi access points and Bluetooth MACs. A hybrid content distribution protocol using cellular, Wifi access point, and P2P (Bluetooth) connectivity can be designed and evaluated using our data set. Furthermore, the collected Bluetooth trace offers a novel opportunity to study a social network created by the ad hoc contacts found in the collected Bluetooth trace. Finally, a comparative study to highlight the similarities and differences between collected WLAN traces [121, 124] and our joint Wifi/Bluetooth trace will be useful for future research in mobile wireless networking.

APPENDIX A

RELATIONAL ALGEBRA

In this chapter, we present the main operations of Relational Algebra, which are used in the thesis. Other operations and notations that are not used in this thesis, will not be presented.

A.1 Terminology

1. Set - a mathematical definition for a collection of objects which contains no duplicates. For example, $S = \{Monday, Tuesday, Wednesday, Thursday, Friday\}$ is a set of weekdays. Here, elements of S do not duplicate.

2. Domain - a set of atomic values. For example, a domain can be "Weekday" whose values are elements of the set S in the above example.

3. Attribute - a real world role played by a named domain. For example, we can give the name "weekday" to the domain "Weekday".

4. Tuple - a collection of attributes which describe some real world entity. For example, $< Monday, 10AM >$ is a tuple, which consists of two attributes $Monday$ and $10AM$. The first is the day and the second is time.

5. Relation - a set of tuples. Table A.1 shows an example of the relation R, in which attributes are "Name" and "Location". Each row in this table is a tuple of the relation R.

Name	**Location**
Student 1	Urbana
Student 2	Urbana
Student 3	Champaign
Student 4	Urbana
Student 5	Champaign

Table A.1: Example of relation R

A.2 Retrieval Operators

We only present two main operators used in the thesis: selection and projection. The input of these operators is a relation R_{in} and the output is a relation R_{out}.

1. Selection: $R_{out} = \sigma_{\varphi} R_{in}$, where φ is a propositional formula that consists of atoms as allowed in the normal selection and the logical operators $\bigwedge$ (and), $\bigvee$ (or) and $\neg$ (negation). This selection operator selects all those tuples in R_{in} for which φ holds. Table A.2 shows an example of the selection operator over the relation R in Table A.1.

Name	**Location**
Student 1	Urbana
Student 2	Urbana
Student 4	Urbana

Table A.2: Output of selection operator $R_{out} = \sigma_{\varphi}(R)$ with $\varphi = (\text{“}Location'' = \text{“}Urbana'')$

2. Projection: $R_{out} = \pi_{a_1,a_2,..,a_n}(R_{in})$, where $a_1, a_2, .., a_n$ are attribute names. The output relation R_{out} consists of projected attributes $a_1, a_2, .., a_n$ and R_{out} has the same number of tuples as R_{in}. Table A.3 shows the example of the output of the projection operator, which takes the attribute “Name” from the relation R in Table A.1.

Name
Student 1
Student 2
Student 3
Student 4
Student 5

Table A.3: Output of projection operator $R_{out} = \pi_{\text{“}Name''}(R)$

APPENDIX B

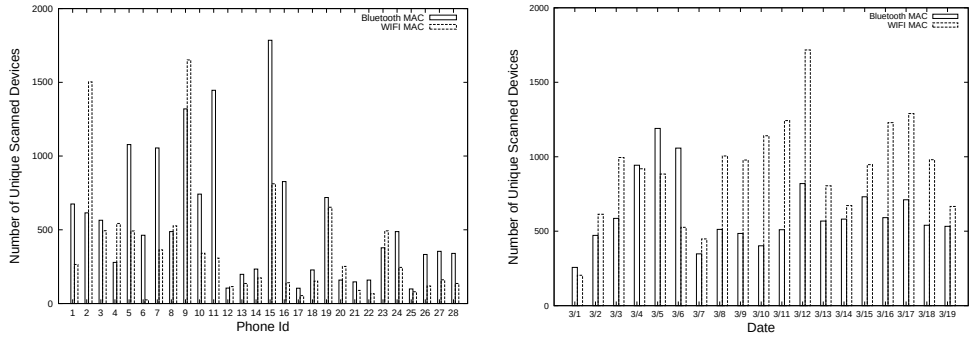
MEASUREMENT RESULTS FROM JOINT WIFI/BLUETOOTH TRACE COLLECTED BY UIM SYSTEM

This chapter is organized as follows. We first present the overall characteristics of the joint Wifi/Bluetooth trace collected by the UIM scanning system in Section B.1. Then, we present the analysis of ad hoc contact distribution in Section B.2. In Section B.3 we discuss the ad hoc graphs formed by collected Bluetooth contacts in the collected trace. Finally, we present the analysis on location visit in Section B.4.

B.1 Overall Characteristics of Collected Joint Wifi/Bluetooth Traces

B.1.1 Number of scanned devices

Result in this section is obtained from the data set D_1 in Table 7.1. Figure B.1(a) shows the total number of unique scanned devices each phone obtains for the entire experiment period. We see that the numbers are considerably different among phones. This is because some participants move much more than others. Also, some locations may have more Bluetooth-enabled devices than other locations. Figure B.1(b) shows the total number of unique scanned devices (both ad hoc MAC and Wifi MAC) over 19 days of experiment. During our experiment, there are two weekends (03/07 and 03/14). These two days have slightly less number of scanned devices than other days. The first day of experiment (03/01) has the least number of scanned devices since we assigned the phones to participants in the late afternoon. Interestingly, in many days during the experiment period, the number of scanned Wifi MACs is more than that of ad hoc MACs. This is because our experiment participants are moving within the University campus and residential areas, which have buildings equipped with Wifi Access Points.



(a) Scanned Device Distribution Over Devices (b) Scanned Device Distribution Over Time

Figure B.1: Number of unique scanned devices in UIM collected trace

B.1.2 Device characteristics

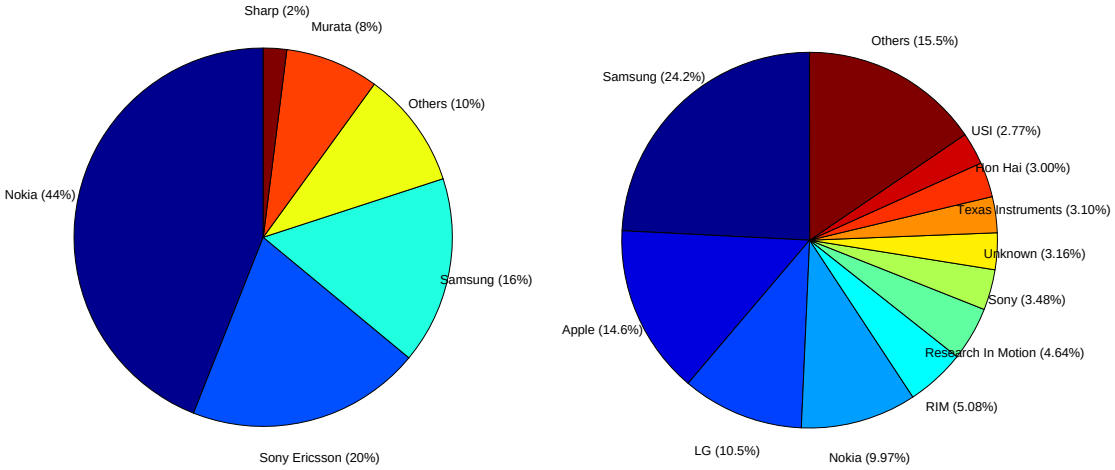
B.1.2.1 Device manufacturer

Result in this section is obtained from the data set D_1 in Table 7.1. Figure B.2 shows the characteristics of the Bluetooth-enabled devices collected in UIM trace, in which Figure B.2(a) show the manufacturers of devices collected in Cambridge City trace [116] and Figure B.2(b) shows the manufacturers of devices collected by our UIM system. Similar to technique used in [116], we use the database of Organizationally Unique Identifiers (OUI) maintained by the IEEE to map MAC address prefixes in the data set to their manufacturers[1]. Figure B.2(a) shows that almost all devices belong to three manufacturers with Nokia (44%), Samsung (16%), Sony (20%). However, Figure B.2(b) shows that Samsung becomes the most popular manufacturer with 24.2% while Nokia and Sony decrease significantly to 9.97% and 3.48%. Meanwhile, Apple is the second most popular with 14.6%, it might because iPhone was not released in 2006 when Cambridge City trace was collected. Figure B.2(b) also shows that 15.5% of devices belong to Other category. We have checked devices and many of them are Android OS devices.

B.1.2.2 Device type

Figure B.3(a) shows types of Bluetooth-enabled devices collected by the UIM system. We see that only 63% of devices are phones, the rest includes computer, headset, GPS, etc. Previous traces [113, 114, 115, 116, 117, 118] did not capture the type of the scanned devices and thus all scanned devices were considered the mobile devices. As a result, previous content distribution protocols

[1] http://standards.ieee.org/regauth/oui/



(a) Manufacturers Collected in [114] (b) Manufacturers Collected by UIM

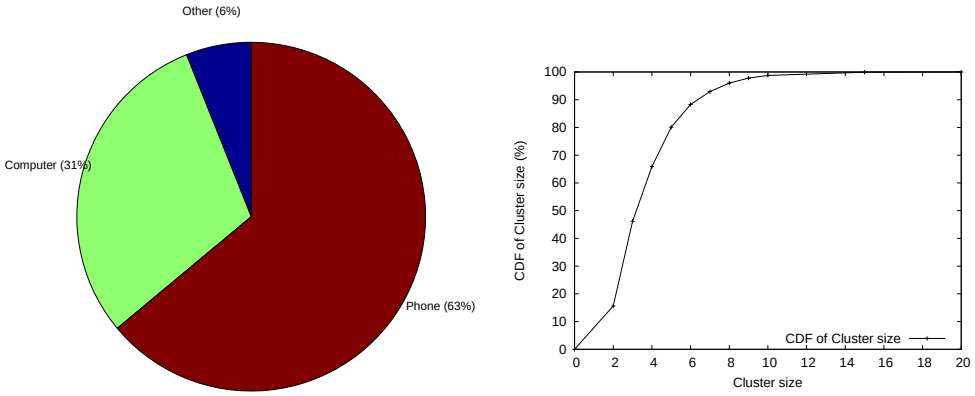
Figure B.2: Device Characteristics

relying on these collected traces did not distinguish between the phones and the other Bluetooth-enabled devices in data dissemination process [114, 115, 116, 118]. Obviously, for the forwarding of the data message, if the selected forwarder is a mobile node, the forwarder has a higher probability to meet other nodes and thus the message has a higher probability to reach the destination. In contrast, if the forwarder is a computer, the forwarding process might be less efficient since the computer (even the laptop computer) may not be carried with the user as much as the phone. To the best of our knowledge, we are the first to collect the type of the devices. This information is useful for the design of a data dissemination protocol, which takes into account the device type in selecting the next forwarder of the data message.

B.1.3 Instant cluster size

Result in this section is obtained from the data set D_1 in Table 7.1. We define the instant cluster C as follows. If two ad hoc MAC u_1 and u_2 appear in one ad hoc scan of the phone p, then (p, M_1, M_2) are in the same instant cluster C. As a result, all ad hoc MACs in one ad hoc scan of the phone p are in the same instant cluster C. For two instant clusters C_1 and C_2, if $C_1 \cap C_2 \neq \emptyset$, then $C_3 = C_1 \cup C_2$ is an instant cluster. So, the definition of instant cluster is transitive.

In order to obtain the instant cluster, we use the ad hoc trace of the data set D_1. For a recorded time stamp t in the ad hoc trace, we consider a time window $[t - 45(s), t + 45(s)]$ and aggregate all ad hoc scanned results of all experiment phones within this time window into an aggregated



(a) Device Type (b) Instant Cluster Size

Figure B.3: Device type and Instant Cluster Size Distrubution

record r. Then, we find the instant clusters within each aggregated record r, and for all records r. The 90-second time window is reasonable since we assume that the cluster size remains unchanged during this time window. Figure B.3(b) shows that about 90% of clusters in our data set have sizes less than 6, which is a small cluster size. Since we already aggregate scanned data for 90 (s) when calculating the cluster size, the cluster size of 6 implies that there are not many big clusters in the network. Therefore, protocols in multicasting, content distribution, and congestion control in DTNs context [164] may need to take this cluster size distribution into consideration.

This result has several insights. The cluster size depends on scanning frequency of the Bluetooth scanner, number of experiment phones, and the phone's Bluetooth hardware capability. More importantly, in university campus, grads and faculties usually stay in their research offices. Meanwhile, the Bluetooth scanner can only scan Bluetooth-enabled devices in the range of 10 meters, hence there might not be big clusters. For undergrads, their class sessions can give big clusters of nodes with high probability. However, the scanning range of an experiment phone is about 10 meters while we have a limited number of undergrad participants in the same class session, we may not be able to capture these clusters. Another possibility is there might not be many Bluetooth-enabled devices in the class sessions. However, other trace collection methods face the same challenges.

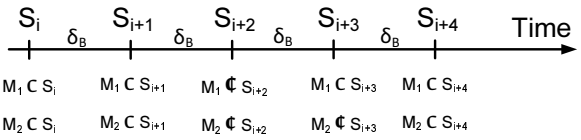


Figure B.4: Contact Definition

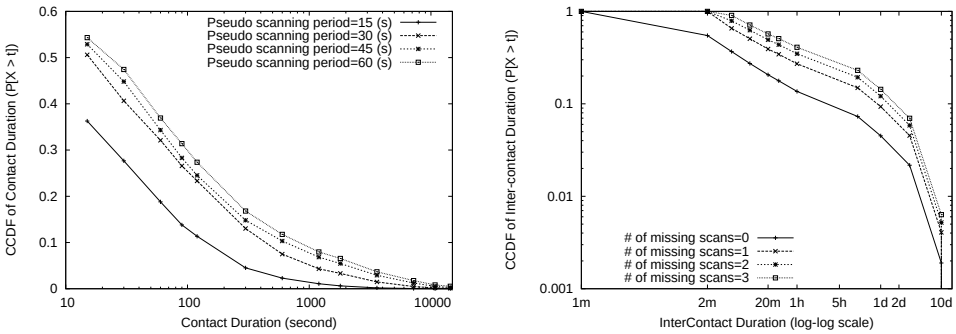
B.2 Contact Analysis

Contact duration and inter-contact duration are two important metrics used to design data forwarding protocols for DTNs. In this section, we analyze the contact duration and inter-contact duration to provide more insights about these two metrics, which have not been provided in the previous studies [112, 114, 115, 116, 118]. Notice that the terms "contact duration" and "inter-contact duration" are the same as the terms "contact time" and "inter-contact time" used in previous studies. We use these two new terms in this chapter since we believe the word "duration" represents properly the meaning of time period while the word "time" does not. Notice that the result in this section is obtained from the data set D_1 in Table 7.1.

B.2.1 Contact definition

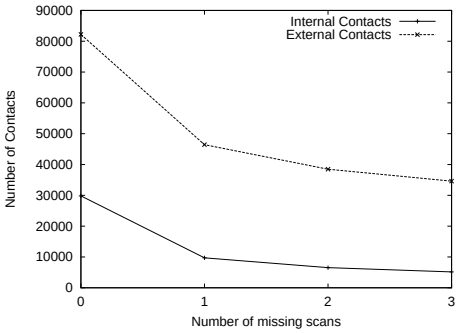
In our context, a phone p and an ad hoc MAC u are said to have a contact if u exists in the Bluetooth scanned result of p. Let T_C denote the contact duration between a phone p and an ad hoc MAC u. T_C could be calculated by using the scanning period δ_B. For example, let N be the number of p's consecutive scans where u appears in the scanned results, $T_C = N \times \delta_B$. However, due to the hardware limitation of the Bluetooth driver at the phone and the unreliable wireless communication channel, it is possible that p does not receive u in its scanned result even when u is inside the Bluetooth sensing range of p. Therefore, in previous works [112, 114, 115, 116, 118], people accepted the missing scans in contact definition as follows: for p and u, although p does not see u in its scanned result for a certain number of scans, p and u are still considered in contact if the number of missing scans is acceptable. Figure B.4 shows an example of contact definition. Let S_i denote the scanned result of p at time t_i. u_1 and u_2 are two ad hoc MACs scanned by the phone p. If the accepted number of missing scan for this figure is 1, from t_i to t_{i+4}, p and u_1 have one contact with the duration of $4\delta_B$ while p and u_2 have two contacts with the durations of $2\delta_B$ and δ_B respectively.

The accepted number of missing scans depends on the trace collection procedures. For example,



(a) Contact Duration Distribution

(b) Inter-contact Distribution



(c) Contact Number Distribution

Figure B.5: Contact Sensitivity

in [114, 115, 116], the number of missing scans is one (with $\delta_B = 120(s)$); however, in [118], this number is 60 (with $\delta_B = 1(s)$). To generalize this, we define Δ_B as the *accepted number of missing scans* in the definition of contact. The contact duration T_C then depends on δ_B and Δ_B. Notice that Δ_B defines the boundary between the two consecutive contacts of a node pair.

B.2.2 Impact of δ_B on contact duration

Figure B.5(a) shows the sensitivity of contact duration when we vary value of δ_B. To obtain this plot, we have 6 students carry phones for 1 week, we set $\delta_B = 15(s)$ for the Bluetooth scanner. Notice that the lower bound (hardware limitation) of Bluetooth scan frequency for Google phone is 12 (s) [165]. We have tried the Bluetooth scan every 10 (s) and most of the time the scanned results are empty. Thus, we set $\delta_B = 15(s)$. Let ${D_1}'$ denote the data set obtained from these 6 phones with $\delta_B = 15(s)$. Each element in ${D_1}'$ is the results obtained by one scan of any phone in the set of 6 phones. Since $\delta_B = 15(s)$, we can derive ${D_2}'$ data set from ${D_1}'$ using pseudo $\delta'_B \in [15, 30, 45, 60]$ as follows: if $\delta'_B = 30(s)$, ${D_2}'$ is the set of odd or even scans in ${D_1}'$. With $\delta'_B = 45(s)$, we take i^{th} scan

from $D_1{}'$ and put into $D_2{}'$, and skip the $(i+1)^{th}$ and $(i+2)^{th}$ scans. Figure B.5(a) is obtained from these $D_2{}'$ sets of corresponding pseudo δ'_B and $\Delta_B = 0$. Notice that this figure is in log-log scale. The figure shows that with different values of δ'_B, we obtain different curves. More importantly, although the curves look similar in shape, the difference between them is significant, ranging from 15% to 20%. That means, the Bluetooth scanning period δ_B has important impacts on calculating contact duration. This has not been investigated in previous studies [112, 114, 115, 116].

Figure B.5(a) also shows that a large amount of contacts (from 35% to 55%) are short contacts (less than 15(s)). Previous studies [112, 114, 115, 116] have not studied the short contact distribution due to their low scanning frequency (see Table 7.2). Except one study in a workplace environment [118], we are the first to study the distribution of short contact in university campus.

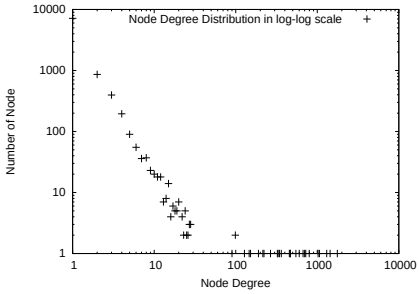
B.2.3 Impact of Δ_B on contact duration

Besides δ_B, Δ_B has an important role in defining the contact duration T_C. This section studies the impacts of Δ_B on inter-contact duration and total number of contacts. Notice that the plots in this section are obtained from entire data set D with $\delta_B = 60(s)$.

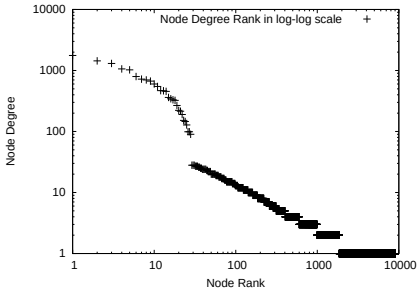
As defined in previous studies [114, 115, 116], inter-contact duration is the time duration between the two consecutive contacts of a given node pair. It is well-known from the previous studies that the inter-contact duration follows the power law [112, 114, 115, 116, 118, 166]. Figure B.5(b) shows that overall the inter-contact duration follows the power law and about 60%-80% of inter-contact duration is less than 1 hour. That means, if a pair of nodes meets at time t, this pair will meet again within one hour after time t with high probability. This figure also shows that when Δ_B varies from 0 to 3, the inter-contact duration varies up to 15%, although the shapes of the curves are similar. So, the value of Δ_B has a clear impact on the inter-contact duration distribution.

Figure B.5(c) shows that when Δ_B varies, the number of external contacts and internal contacts in the data set D_1 changes significantly [2]. For the greater value of Δ_B, the definition of contact is more "robust" to missing scans and thus the contact lasts longer; thus, we have less number of contacts. For example, when Δ_B increases from 0 to 1, the number of external contacts decreases more than 30% from about 80000 to about 50000, while that of internal contacts decreases more than 60% from about 30000 to 10000. Similarly, when Δ_B increases from 1 to 2, the number of external contacts decreases 20% and that of the internal contacts decreases 30%.

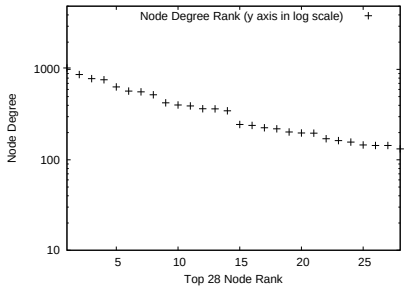
[2] Definitions of internal and external contact can be found in [44].



(a) Node Degree Distribution



(b) Node Degree Rank



(c) Top 28 of Node Rank

Figure B.6: Node Degree of Connectivity Graph

In conclusion, definition of contact depends on δ_B and Δ_B. So, when using (inter-) contact duration distribution reported in previous studies [114, 115, 116], the readers should carefully consider corresponding values of δ_B and Δ_B since they have significant impacts on (inter-) contact duration distribution.

B.3 Mining Ad hoc Graph

We investigate the connectivity graph and contact graph formed by the ad hoc trace of the data set D_1 in Table 7.1.

B.3.1 Connectivity graph

In our context, the connectivity graph $G =< V, E >$ is an undirected graph, which is defined as follows. V is the set of nodes, including experiment phones and external ad hoc MACs. For a pair of nodes $v_1, v_2 \in V$, if v_1 is a phone and v_2 appears in one scanned result of v_1, then the edge $(v_1, v_2) \in E$. Notice that, if v_1 is a phone and v_2, v_3 exist in one scanned result of v_1, then in our context, $(v_1, v_2) \in E$, $(v_1, v_3) \in E$, but $(v_2, v_3) \notin E$. This definition determines the node degree

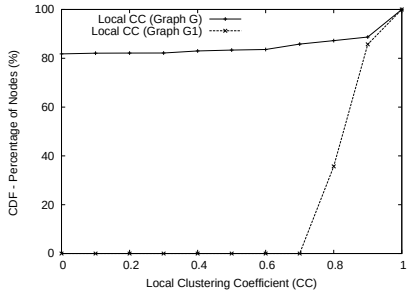


Figure B.7: Local Clustering Coefficient Distribution

distribution of G in the following discussion.

Figure B.6 shows that the node degree[3] distribution of the connectivity graph G follows the Zipf distribution with a heavy-tailed cut-off at the node degree greater than 28. We also find that node degree mean of G is 3.26. Next, we plot the node rank in terms of node degree in Figure B.6(b). This figure shows that for the node rank greater than 28, the node degree follows the Zipf distribution. We then focus on the first 28 nodes in Figure B.6(c), which shows that the node degree linearly decreases with respect to the node rank. These 28 nodes are experiment phones, which have a much higher node degree according to the definition of our connectivity graph. We conclude that the node degree distribution of G exhibits an Exponential-Zipf distribution.

To further examine the structure of the graph G, we calculate the local clustering coefficient (CC) [167] for all nodes in V. As shown in Figure B.7, more than 80% of nodes has $CC = 0$, these nodes are all leaf nodes which have only one neighbor (i.e., the experiment phone). Since 80% of nodes have $CC = 0$, the global CC of the graph is 0.157.

Moreover, we create a connectivity graph $G_1 =< V_1, E_1 >$ of experiment phones and calculate the CC for G_1. Here, V_1 is the set of experiment phones and $|V_1| = 28$. Also, for a pair of nodes $v_1, v_2 \in V_1$, if v_2 appears in one scanned result of v_1, then the edge $(v_1, v_2) \in E_1$. Figure B.7 shows that 60% of experiment phones have the local CC greater than 0.8, thus the global CC of G_1 is 0.814, which indicates that the graph formed by phones is highly clustered.

From our analysis, G is a connected graph with 9015 nodes and graph diameter is 4. The low mean of node degree (e.g., 3.26) results from the ad hoc MACs, which are the leaf nodes in the graph with only edges to the experiment phones. From Figure B.6(b) we see that the first 50 nodes have degree greater than 25, these nodes form the hubs of G and reduce the graph diameter. Meanwhile, we have only 28 phones, that means the external ad hoc MACs also are hubs in G.

[3]Number of direct neighbors of a node

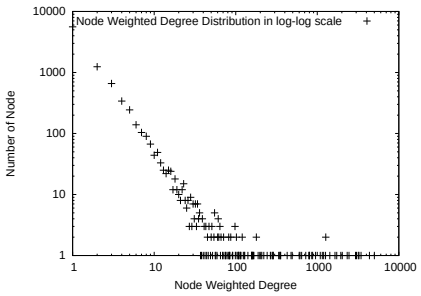
This is further confirmed in Figure B.7 where the local CC of many phones in G_1 is less than 1, which means there exist cases where the two phones are not connected by a direct edge in G_1. For these cases, the external ad hoc MACs, which are hubs, connect these phones to make G connected and reduce the graph diameter. Besides, although the global CC of the graph G is 0.157, it is considerably greater than the global CC of a random graph with $|V| = 9015$ and mean node degree 3.26 (which is $3.26/9015 = 0.00036$). So, we conclude that G exhibits a small-world network in structure.

This finding has several implications. First, it is well-known that the social network is a small-world network in structure, since the connectivity graph represents the social relationship among nodes in the graph, the finding is an expected result. Second, since the connectivity graph is a small-world network, there are hubs in the network, which can be exploited to expedite message forwarding in the network. This further confirms the recent social-based forwarding protocols in DTN research [147, 161, 149].

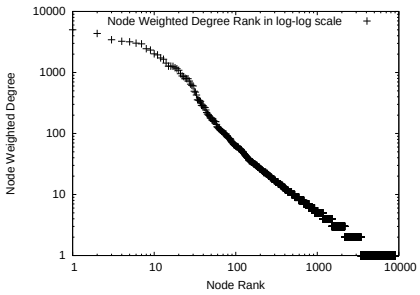
B.3.2 Contact graph

In our context, the contact graph $G_C =< V_C, E_C >$ is a "weighted" version of the connectivity graph G. G_C can be obtained from G as follows. For an edge $(v_1, v_2) \in E$, we have a weighted edge $(v_1, v_2)_w \in E_C$, where the weight is the number of contacts between v_1 and v_2 in the ad hoc trace. Notice that the definition of contact can be found in Section B.2.1 or from previous studies [44, 114, 115, 116, 117].

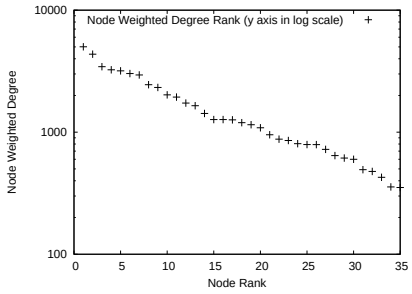
For a vertex $v_1 \in V_C$, the weighted degree of v_1 is the sum of weights of edges, which are adjacent to v_1 in G_C. The graph G_C is a connected graph with 9015 nodes. The graph diameter is 4 and the mean of node weighted degree is 9.86. Figure B.8(a) shows that the node weighted degree follows a Zipf distribution with a heavy-tailed cut-off. This is confirmed in Figure B.8(b) where we plot the rank of nodes in terms of node weighted degree. In this figure, starting from rank 35^{th}, node weighted degree follows very well the Zipf distribution. We then focus on the top 35 node rank in Figure B.8(c), which shows that the node weighted degree linearly decreases with respect to node rank. Therefore, we conclude that the weighted node degree distribution exhibits an Exponential-Zipf distribution.



(a) Node Weighted Degree Distribution



(b) Node Weighted Degree Rank



(c) Top 35 of Node Rank

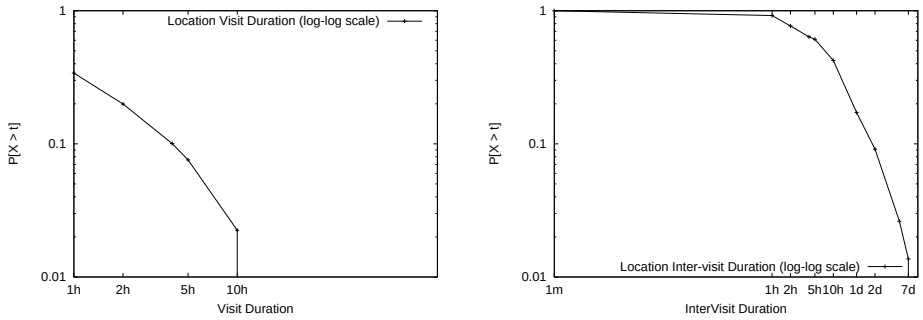
Figure B.8: Node Weighted Degree of Contact Graph

B.4 Characterizing Location Visit

The result in this section is taken from the data set D_1 in Table 7.1. We apply the UIM Clustering algorithm from Section 8.3 to obtain locations for the below analysis.

B.4.1 Location visit duration

In our context, we consider a phone p has a "location visit" with a location L if L appears once in the location trace of p. Notice that the definition of "visit" between a phone and a location is similar to the definition of contact between a phone and an ad hoc MAC. We first calculate the "location visit duration", which is the duration the phone stays at a particular location. Similar to the contact definition, the definition of location visit depends on Wifi scanning frequency δ_W and the accepted number of missing scans Δ_W. In Figure B.9, we have $\delta_W = 30(min)$, $\Delta_W = 0$, and we use the entire data set D. Figure B.9(a) shows that about 60% of location visits is less than 1 hour and the longest location visit is 10 hours. Since we have $\delta_W = 30(min)$, the result from this figure relies on the following assumption: for two consecutive Wifi scans, if p scans the same set of Wifi MACs, that means p stays at the same location during the last 30 (min). This might not



(a) Location Visit Duration (b) Location Inter-Visit Duration

Figure B.9: Location Visit Analysis

be true if the phone carriers move to another location and then come back to L within the last 30 (min). However, this is the common limitation of existing trace collection methods since we only can obtain the "discrete" rather than "continuous" scanned result. The result in Figure B.9 also results from our scan configuration since UIM scanners stop scanning at night to conserve phone battery.

B.4.2 Location inter-visit duration

For a pair of phone p and location L, the inter-visit duration is the time duration between the two consecutive visits of p at L. Figure B.9(b) shows the inter-visit duration distribution of our data set. This result differs from the figure obtained from the association time between a laptop and its Wifi access point [4], which was a heavy-tailed distribution as presented in previous study [127]. From this figure we see that the longest inter-visit duration is one week and for a pair of phone p and location L, if p visit L at time t, it is unlikely that p will return to L after one hour.

B.4.3 Location popularity

In our context, **the location L_1 is more popular than the location L_2 if there are more Bluetooth MACs scanned by phones at L_1 than at L_2.**

Notice that our Wifi scanner scans the Wifi MACs (e.g., the location) every 30 minutes while the Bluetooth scanner scans every 60 (s). To obtain the popularity of locations in data set D_1, we *combine the Bluetooth trace and Wifi trace* using the scan time as follows. If at time t the phone p appears at the location L in the data set D_1 (e.g., p scans the Wifi APs of L), we look into

[4] A Wifi access point was a physical location in previous study.

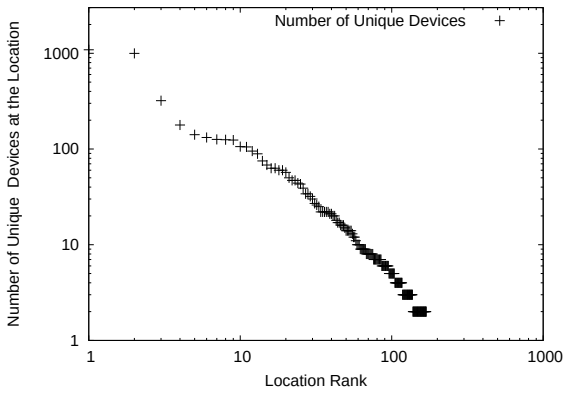


Figure B.10: Location Popularity

all ad hoc MACs scanned by p in the Bluetooth trace during the period $[t - 5(min), t + 5(min)]$, aggregate these ad hoc MACs, and assign them as scanned ad hoc MACs at the location L. After repeating this for the entire data set D_1 and for all locations from 170 locations, we then aggregate all ad hoc MACs of the same location L into a unique set, which represents the set of ad hoc MACs scanned at L by all experiment phones for the entire experiment period. Figure B.10 shows the location popularity in terms of number of scanned ad hoc MACs at the locations (Notice that the figure is in log-log scale). This figure shows that location popularity exhibits a heavy-tailed distribution. Particularly, when location rank is greater than 10, the location popularity follows a Zipf distribution.

APPENDIX C

HYBRID EPIDEMIC DATA DISSEMINATION

C.1 Design of Hybrid Epidemic Protocol

As shown in Figure B.1(b), in many days during our experiment period, the number of scanned Wifi MACs is more than that of ad hoc MACs. Also, in the data set D_1 in Table 7.1, our 28 phones collected 7004 Wifi access points and 8508 Bluetooth-enabled devices during the experiment period. So, we believe a data dissemination protocol, which uses both Wifi access points and ad hoc contacts to forward data messages, becomes applicable in reality. This motivates us to design a new data dissemination protocol named Hybrid Epidemic Data Dissemination protocol (or Hybrid Epidemic protocol for short), which combines both Wifi access points and ad hoc contacts in data dissemination.

Figure C.1 shows the network model of the Hybrid Epidemic protocol with two main components: Wifi access points and mobile nodes. We assume that the Wifi access points are connected via the WLAN or Internet connection (e.g., AP_1 and AP_2 are connected and can exchange data via the WLAN backbone or Internet connection). Besides, the mobile nodes can communicate in infrastructured-based and ad hoc modes. The ad hoc connectivity can be either Bluetooth or Wifi.

Figure C.1 also shows how the Hybrid Epidemic protocol works. Particularly, when the sender P_1 sends a message m to the receiver P_5, P_1 uploads m to the Wifi access point AP_1[1] whenever P_1 is within the transmission range of AP_1. At the same time P_1 and other nodes in the network perform the epidemic procedure to forward m toward the destination. For example, P_1 sends the message m to P_2 when two mobile nodes are in contact. Again, P_2 forwards m to P_3 when P_2 and P_3 are in contact, and so forth. After P_1 uploads m to the Wifi access point AP_1, AP_1 broadcasts m to other Wifi access points in the network. Upon receiving m from AP_1, AP_2 advertises m to its surrounding area and thus if P_4 is in AP_2's range, P_4 can download m from AP_2 via the

[1]We assume that the Wifi access point has the storage to cache data message.

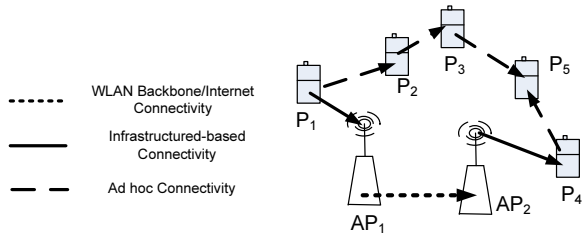


Figure C.1: Network Model of Hybrid Epidemic protocol

infrastructured connectivity. After that, P_4 can use ad hoc contacts to expedite the forwarding of m toward the receiver P_5 when P_4 and P_5 are in contact. The receiver P_5 can receive m from Wifi access points or ad hoc contact.

We use the "forwarding delay" metric to evaluate Hybrid Epidemic protocol. For a pair of sender/receiver (s/r), the forwarding delay is the time period since s starts sending m toward r, until r receives m. We are seeking the answer for the question: how much forwarding delay improvement the Hybrid Epidemic protocol can achieve. To obtain the answer, we compare Hybrid Epidemic protocol to the Epidemic protocol [17], which only uses ad hoc contacts to forward data messages (no Wifi access points are used in Epidemic protocol). The Epidemic protocol has been the fundamental data dissemination protocol in DTN research [144]. In Figure C.1, if the Epidemic protocol is used to forward data message m from P_1 to P_5, the forwarding path is P_1, P_2, P_3, P_5. Notice that although Epidemic protocol incurs a high network overhead, it does achieve a high delivery ratio and a nearly optimal forwarding delay [168] since Epidemic protocol exploits all possible ad hoc paths from the sender to the receiver.

Since we are only interested in the forwarding delay rather than other metrics (e.g., network overhead), we have left following design issues for our future work. First, we do not limit the number of copies of m in the network (e.g., mobile node P_1 makes a copy of m and forwards to P_2). That means, the message m is "flooded" to the entire network by the ad hoc contact and Wifi access points[2]. Second, how long the Wifi access points cache the data messages is not the focus of this paper. Figure B.10 shows that the location popularity follows a heavy-tailed distribution. That means, it is possible that the Wifi access points at the more popular locations can cache data message for a longer period so that mobile nodes have a higher probability to obtain the message. Third, figure 7.9 shows that people movement is regular and they visit same locations for their daily activities. Our protocol does not exploit this movement regularity in data dissemination yet. The

[2]This is the nature of epidemic data dissemination.

Hybrid Epidemic protocol follows the trend of combining infostations and ad hoc connectivity to improve data delivery in previous works [169][2], which were evaluated by mean of simulation. Our main contribution in this paper is to *evaluate and compare the performance of Hybrid Epidemic and Epidemic protocols on the real movement trace obtained by UIM.*

C.2 Evaluation Setting

As presented in a survey of 300 computer science faculty members and students [118], the forwarding delay that people can tolerate is from one to several hours, depending on the delay-tolerant networking applications and services [118]. Therefore, we evaluate the performance of Hybrid Epidemic protocol for a day-long period. Particularly, we compare the performance of Epidemic protocol and Hybrid Epidemic protocol using the ad hoc trace and Wifi trace collected by 8 phones carried by grad students in the same research group for two different days extracted from the data set D_1 in Table 7.1. Let D_4 and D_5 denote the collected traces for these two days.

Since the participants come from the same research group, D_4, D_5 provide richer sets of contacts among 8 phones as well as a richer overlapping set of external ad hoc MACs. This is important for the Epidemic protocol in order to improve the forwarding delay of the message since the performance of this protocol depends fully on ad hoc contact. If D_4 and D_5 are traces collected by a random set of participants, the data set may not have a good overlapping set of ad hoc MACs, which makes the data forwarding unreachable or incurs an unacceptable long forwarding delay. Our formation of D_4 and D_5 also represents the realistic scenario since 8 participants from the same research group are from the same "community", thus they may share mutual content interest and share mutual contacts. Notice that D_4 has 186 unique Bluetooth MACs while D_5 has 209 unique Bluetooth MACs.

Using the data sets D_4 and D_5, the Hybrid Epidemic protocol works as follows: for each data set, at time t_1 a phone P_1 starts sending message m to a receiver P_2. After time t_1, P_1 can forward m to P_4 if P_1 sees P_4 in its Bluetooth trace. At time $t_2 \geq t_1$, P_1 uploads m to the Wifi access points AP_1 if P_1 sees AP_1 in its Wifi trace at that time. After the time t_2, another phone P_3 can download m from the Wifi access point AP_3 if P_3 sees AP_3 in its Wifi trace. Notice that P_3 also uses the ad hoc contacts in its Bluetooth trace to forward m. In many cases P_3 may receive m from its ad hoc contacts before P_3 encounters a Wifi access point. The the limitation of our Wifi scanner is that UIM scan Wifi access points every 30 minutes and can not capture the "continuous" Wifi trace.

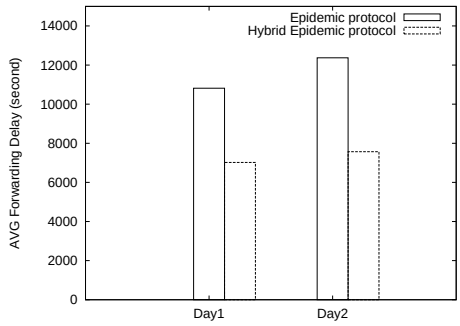


Figure C.2: Performance of Hybrid Epidemic protocol

However, it is also the limitation of any Wifi scanners due to the battery consumption constraint. More importantly, if we have a more frequently scanned Wifi trace (e.g., $\delta_W < 30(minutes)$), the performance of Hybrid Epidemic protocol may only improve since P_3 can download m from its Wifi access point at a earlier time after P_1 uploads m to the Wifi access points. Here, P_2 can receive m from Wifi access point or ad hoc contact.

C.3 Performance Evaluation

We select 50 random pairs of (sender, receiver) from D_4, and 50 random pairs of (sender, receiver) from D_5 and apply Epidemic protocol and Hybrid Epidemic protocol on the two data sets to forward the message from the sender to the receiver. Figure C.2 shows that Hybrid Epidemic protocol achieves much shorter average forwarding delay than Epidemic protocol (e.g., 3500 (s) and 5500 (s) for day 1 and day 2, respectively). This is because Hybrid Epidemic protocol combines both ad hoc contact and Wifi access points to improve the forwarding of the data messages. In our experiment, when Hybrid Epidemic protocol scheme is used to forward messages, the messages always reach the destination. However, for Epidemic protocol, 10% and 17% of messages can not reach the destination, for day 1 and day 2 respectively. Notice that in Figure C.2, the forwarding delay for a pair (sender,receiver) is only taken for the average delay calculation if the message is received at the receiver.

REFERENCES

[1] A. Heinemann, J. Kangasharju, F. Lyardet, and M. Mühlhäuser, "iClouds - peer-to-peer information sharing in mobile environments," 2003.

[2] M. Motani, V. Srinivasan, and P. S. Nuggehalli, "Peoplenet: Engineering a wireless virtual social network," in *Proceedings of Mobicom*, 2005.

[3] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of Mobicom*, 1999.

[4] M. Papadopouli and H. Schulzrinne, "Effects of power conservation, wireless coverage anc cooperation on data dissemination among mobile devices," in *Proceedings of Mobicom*, 2001.

[5] O. Wolfson, B. Xu, and R. M. Tanner, "Mobile peer-to-peer data dissemination with resource constraints," in *Proceedings of MDM*, 2007.

[6] O. Wolfson, B. Xu, H. Yin, and H. Cao, "Search-and-discover in mobile p2p network databases," in *Proceedings of ICDCS*, 2006.

[7] X. Yang and A. Bouguettaya, "Using a hybrid method for accessing broadcast data," in *Proceedings of MDM*, 2005.

[8] H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu, "The design and evaluation of unified cellular and ad hoc networks," *IEEE Transactions on Mobile Computing*, 2007.

[9] L. Vu, R. Malik, Q. Wang, and K. Nahrstedt, "An adaptive content distribution protocol for dynamic peer-to-peer coalition networks," in *Proceedings of 4th International workshop on Adaptive and DependAble Mobile Ubiquitous Systems (ADAMUS 2010)*, 2010.

[10] L. Vu, I. Rimac, V. Hilt, M. Hofmann, and K. Nahrstedt, "Exploiting opportunistic ad hoc connections for improving data download of cellular users," in *Proceedings of 1st IEEE Workshop on Pervasive Group Communication (Globecom 2010)*, 2010.

[11] L. Vu, K. Nahrstedt, and M. Hollick, "Exploiting schelling behavior for improving data accessibility in mobile peer-to-peer networks," in *Proceedings of Mobiquitous*, 2008.

[12] T. Repantis and V. Kalogeraki, "Data dissemination in mobile peer-to-peer networks," in *Proceedings of MDM*, 2005.

[13] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1997, pp. 90–100.

[14] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Kluwer Academic Publishers, 1996, pp. 153–181.

[15] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad-hoc wireless networks," in *Proceedings of Infocom*, April 2001, pp. 1360–1369.

[16] S. Jung, U. Lee, A. Chang, D.-K. Cho, and M. Gerla, "Bluetorrent: Cooperative content sharing for bluetooth users," *Pervasive and Mobile Computing*, vol. 3, pp. 609–634, 2007.

[17] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," in *Technical Report CS-200006, Duke University*, 2000.

[18] W. H. Yuen, R. D. Yates, and S.-C. Mau, "Exploiting data diversity and multiuser diversity in noncooperative mobile infostation networks," in *Proceedings of Infocom*, 2003.

[19] X. Hong, M. Gerla, G. Pei, and C. chuan Chiang, "A group mobility model for ad hoc wireless networks," in *Workshop on Modeling, Analysis and Simulation of wireless and mobile systems*, 1999.

[20] J. Tian, J. Haehner, C. Becker, I. Stepanov, and K. Rothermel, "Graph-based mobility model for mobile ad hoc network simulation," in *Proceedings of the 35th Annual Simulation Symposium*, 2002.

[21] A. P. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri, "Real-world environment models for mobile network evaluation," *Journal on Selected Areas in Communications*, vol. 23, pp. 622–632, 2005.

[22] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "On clustering phenomenon in mobile partitioned networks," in *Proceedings of MobilityModels*, 2008.

[23] P. Nain, D. Towsley, B. Liu, and Z. Liu, "Properties of random direction models," in *Proceedings of Infocom*, 2005.

[24] J. Kim and S. Bohacek, "A survey-based mobility model of people for simulation of urban mesh networks," in *Proceedings of MeshNets*, 2005.

[25] D. Lam, D. Cox, and J. Widom, "Teletraffic modeling for personal communications services," *Communications Magazine*, vol. 35, pp. 79–87, Feb 1997.

[26] M. Musolesi and C. Mascolo, "Designing mobility models based on social network theory," *IACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, pp. 59–70, July 2007.

[27] D. Tang and M. Baker, "Analysis of a metropolitan-area wireless network," in *Proceedings of Mobicom*, 1999, pp. 13–23.

[28] M. Hollick, T. Krop, H.-P. Huth, and R. Steinmetz, "Modeling mobility and workload for wireless metropolitan area networks," *Computer Communications*, vol. 27, pp. 751–761, May 2004.

[29] C. A. H. M. C. Gonzalez and A. L. Barabasi, "Undestanding individual human mobility patterns," *Nature*, vol. 7196, pp. 779–782, 2008.

[30] A. Millonig and G. Gartner, "Monitoring pedestrian spatio-temporal behaviour," in *Proceedings of BMI*, 2007.

[31] *Location Based Services and TeleCartography II (Mapping Pedestrian Movement: Using Tracking Technologies in Koblenz).* Springer Berlin Heidelberg, 2009, pp. 95–118.

[32] S. Nanda, "Teletraffic models for urban and suburban microcells: cell sizes and handoff rates," *IEEE Transaction on Vehicular Technology*, vol. 42, pp. 673–682, 1993.

[33] J. Kim, V. Sridhara, and S. Bohacek, "Realistic mobility simulation of urban mesh networks," *Elsevier Science*, vol. 7, pp. 411–430, June 2009.

[34] W. jen Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling spatial and temporal dependencies of user mobility in wireless mobile networks," *CoRR*, vol. abs/0810.3935, 2008.

[35] S. C. Nelson, A. F. H. III, and R. Kravets, "Eventdriven, rolebased mobility in disaster recovery networks," in *Proceedings of CHANTS*, 2007.

[36] Y. Huang, W. He, K. Nahrstedt, and W. C. Lee, "Corps: Event-driven mobility model for first responders in incident scene," in *Proceedings of Milcom*, 2008.

[37] F. Bai, N. Sadagopan, and A. Helmy, "Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks," in *Proceedings of Infocom*, 2003.

[38] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Proceedings of scenario-based performance analysis of routing protocols for mobile ad-hoc networks," in *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking.* ACM Press, 1999, pp. 195–206.

[39] M. Musolesi, S. Hailes, and C. Mascolo, "An ad hoc mobility model founded on social network theory," in *Proceedings of ACM International symposium on Modeling, Analysis and simulation of wireless and mobile systems*, 2004.

[40] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Proceedings of International Symposium on Mobile Ad Hoc Networking and Computing*, 2006.

[41] "Dartmouth College (Mobility Trace): http://crawdad.cs.dartmouth.edu/meta.php?name=dartmouth/campus," 2009.

[42] "University of Florida (MobiLib): http://nile.cise.ufl.edu/mobilib/," 2009.

[43] L. Vu, K. Nahrstedt, R. Malik, and Q. Wang, "COADA: Leveraging dynamic coalition peer-to-peer network for adaptive content download of cellular users," *International Journal of Adaptive, Resilient and Autonomic Systems (IJARAS)*, 2011. (Accepted, to appear).

[44] L. Vu, K. Nahrstedt, S. Retika, and I. Gupta, "Joint bluetooth/wifi scanning framework for characterizing and leveraging people movement in university campus," in *Proceedings of MSWiM*, 2010.

[45] "Nus data set. http://crawdad.cs.dartmouth.edu/nus/contact."

[46] L. Vu, Q. Do, and K. Nahrstedt, "Exploiting joint wifi/bluetooth trace to predict people movement," Tech. Rep., 2010. [Online]. Available: http://www.ideals.illinois.edu/handle/2142/16944

[47] A. Lindgren, A. Doria, and O. Scheln, "Probabilistic routing in intermittently connected networks," in *Proceedings of Mobihoc*, 2003.

[48] D. D. Deavours, G. Clark, T. Courtney, D. Daly, S. Derisavi, J. M. Doyle, W. H. Sanders, and P. G. Webster, "The mobius framework and its implementation," *IEEE Transactions on Software Engineering*, vol. 28, no. 10, pp. 956–969, October 2007.

[49] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility pattern," *Nature*, vol. 453, pp. 779–782, June 2008.

[50] H. Yoon, J. Kim, F. Tany, and R. Hsieh, "On-demand video streaming in mobile opportunistic networks," in *Percom*, 2008.

[51] M. Stiemerling and S. Kiesel, "A system for peer-to-peer video streaming in resource constrained mobile environments," in *CoNext U-Net Workshop*, 2009.

[52] T. C. Schelling, "Models of segregation," *American Economic Review*, vol. 59, 1969.

[53] "Google geolocation API, http://code.google.com/apis/gears/api_geolocation.html."

[54] *Location Based Services and TeleCartography II (Ways of Walking - Developing a Pedestrian Typology for Personalised Mobile Information).* Springer Berlin Heidelberg, 2009, pp. 79–94.

[55] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based performance analysis of routing protocols for mobile ad-hoc networks," in *Proceedings of MobiCom.* ACM Press, 1999, pp. 195–206.

[56] G. Sharma, R. Mazumdar, and N. Shroff, "Delay and capacity tradeoffs in mobile ad hoc networks: a global perspective," in *Proceedings of IEEE Infocom*, 2006.

[57] J. Broch, D. A. Maltz, D. B. Johnson, Y. chun Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of ACM/IEEE MobiCom*, 1998, pp. 85–97.

[58] J. Ng and Y. Zhang, "Reference region group mobility model for ad hoc networks," in *Proceedings of Wireless and Optical Communications Networks*, 2005.

[59] K. H. Wang and B. Li, "Group mobility and partition prediction in ad hoc wireless networks," in *Proceedings of IEEE International Conference on Communications*, 2002.

[60] M. Sanchez and P. Manzoni, "A java-based ad hoc networks simulator," in *Proceedings of the SCS Western Multiconference Web-based Simulation Track*, 1999.

[61] B. Zhou, K. Xu, and M. Gerla, "Group and swarm mobility models for ad hoc network scenarios using virtual tracks," in *Proceedings of Milcom*, 2004.

[62] N. Aschenbruck, M. Frank, P. Martini, and J. Tolle, "Human mobility in manet disaster area simulation - a realistic approach," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*, 2004.

[63] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communication and Mobile Computing Special Issue on Mobile Ad Hoc Networking*, vol. 2, no. 5, pp. 483–502, 2002.

[64] S. Ray, "Realistic mobility for manet simulation," in *Master Thesis, The University of British Columbia*, 2003.

[65] C. Tuduce and T. Gross, "Mobility model based on wlan traces and its validation," in *Proceedings of Infocom*, 2005.

[66] M. Mcnett and G. M. Voelker, "Access and mobility of wireless PDA users," *SIGMOBILE Mobile Computing Communication Review*, vol. 9, no. 2, pp. 40–55, April 2005.

[67] W.-J. Hsu, K. Merchant, H.-W. Shu, C.-H. Hsu, and A. Helmy, "Weighted waypoint mobility model and its impact on ad hoc networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, no. 1, pp. 59–63, January 2005.

[68] F. Ekman, A. Kernen, J. Karvo, and J. Ott, "Working day movement model," in *Proceedings of MobilityModels*, 2008.

[69] K. Maeda, A. Uchiyamaa, T. Umedua, H. Yamaguchia, K. Yasumotob, and T. Higashinoa, "Urban pedestrian mobility for mobile wireless network simulation," *Ad Hoc Networks*, vol. 7, pp. 153–170, January 2009.

[70] J. Fruin, "Metropolitan association of urban designer and environmental planners," in *Proceedings of Pedestrian Planning Design*, 1971.

[71] J. Dijkstra, A. Jessurun, and H. Timmermans, "A multi-agent cellular automata model of pedestrian movement," in *Proceedings of Pedestrian and Evacuation Dynamics*, 2001.

[72] *Location Based Services and TeleCartography II.* Springer Berlin Heidelberg, 2009, pp. 301–315.

[73] P. Maymounkov and D. Mazieres, "Rateless codes and big downloads," in *Proceedings of IPTPS03*, 2003.

[74] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi, "CDMA/HDR: A bandwidth-efficient high-speed wireless data service for normadic users," *IEEE Communications Magazine*, 2000.

[75] H. Hanano, Y. Murata, N. Shibata, K. Yasumoto, and M. Ito, "Video ads dissemination through wifi-cellular hybrid networks," in *Proceedings of Percom*, 2009.

[76] S. G. Aly and A. A. Elnahas, "Sustained service lookup in areas of sudden dense population," *Wireless Communications and Mobile Computing*, vol. 8, pp. 61–74, 2008.

[77] H.-Y. Hsieh and R. Sivakumar, "On using peer-to-peer communication in cellular wireless data networks," *IEEE Transactions on Mobile Computing*, vol. 3, 2004.

[78] K. S. and M. M., "Efficient mobile access to internet data via a wireless peer-to-peer network," in *Pervasive Computing*, 2004.

[79] I. Ioannidis, B. Carbunar, and C. Nita-Rotaru, "High throughput routing in hybrid cellular and ad-hoc networks," in *World of Wireless Mobile and Multimedia Networks WoWMoM*, 2005.

[80] J. C. Park and S. Kasera, "Enhancing cellular multicast performance using ad hoc networks," in *Wireless Communications and Networking Conference WCNC*, 2005.

[81] C. Gkantsidis and P. Rodriguez, "Network coding for large scale content distribution," in *Proceedings of Infocom*, 2005.

[82] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE Journal on Selected Areas in Communications, Special Issue on Advances in Peer-to-Peer Streaming Systems*, vol. 25, December 2007.

[83] U. Lee, J.-S. Park, S.-H. Lee, W. W. Ro, G. Pau, and M. Gerla, "Efficient peer-to-peer file sharing using network coding in manet," *Journal of Communications and Networks (JCN), Special Issue on Network Coding*, vol. 10, pp. 422–429, December 2008.

[84] O. Jrv, A. Aasa, R. Ahas, and E. Saluveer, *Weather dependence of tourist's spatial behavior and destination choices: case study with passive mobile positioning data in Estonia*, 2007.

[85] Y. Mansfeld and K. Ya"acoub, "Patterns of tourist destination-choice and travel behaviour among members of the urban and rural arab community of israel: A comparative study of haifa and ibilin," *GeoJournal*, vol. 35, no. 4, pp. 459–470, 2004.

[86] A. Papatheodorou, "Why people travel to different places," *Annals of Tourism Research*, vol. 28, no. 1, pp. 164–179, 2001.

[87] "Skyhook. http://www.skyhookwireless.com."

[88] "GPS IIF.
http://www.boeing.com/defense-space/space/gps/index.html."

[89] L. Vu, I. Rimac, V. Hilt, M. Hofmann, and K. Nahrstedt, "iShare: exploiting opportunistic ad hoc connections for improving data download of cellular users," in *UIUC Technical Report*, 2009.

[90] M. Stiemerling and S. Kiesel, "Cooperative p2p video streaming for mobile peers," in *Proceedings of IEEE ICCCN*, 2010.

[91] T. K. Madsen, Q. Zhang, F. Fitzek, and M. Katz, "Exploiting cooperation for performance enhancement and high data rates," *Journal of Communications*, 2009.

[92] A. Correia, J. Silva, N. Souto, L. Silva, A. Boal, and A. Soares, "Multi-resolution broadcast/multicast systems for MBMS," *IEEE Transactions on Broadcasting*, vol. 53, pp. 224–234, 2007.

[93] D. Gomez-Barquero, A. Aguilella, and N. Cardona, "Multicast delivery of file download services in 3g mobile networks with MBMS," in *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*, 2008.

[94] F. Hartung, U. Horn, J. Huscheke, M. Kampmann, T. Lohmar, and M. Lundevall, "Delivery of broadcast services in 3G networks," *IEEE Transactions on Broadcasting*, vol. 53, 2007.

[95] M. Luby, T. Gasiba, T. Stockhammer, and M. Watson, "Reliable multimedia download delivery in cellular broadcast networks," *IEEE Transactions on Broadcasting*, vol. 53, pp. 235–246, 2007.

[96] S. Parkvall, E. Englund, M. Lundevall, and J. Torsner, "Evolving 3G mobile systems: Broadband and broadcast services in WCDMA," *IEEE Communications Maganize*, 2006.

[97] R. Bhatia, L. E. Li, H. Luo, and R. Ramjee, "ICAM: Integrated cellular and ad-hoc multicast," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 1004–1015, August 2006.

[98] K. Chen, Z. Yang, C. Wagener, and K. Nahrstedt, "Market models and pricing mechanisms in a multihop wireless hotspot network," in *Proceedings of Mobiquitous*, 2005.

[99] M. Goemans, L. Li, V. Mirrokni, and M. Thottan, "Market sharing games applied to content distribution in ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 24, pp. 1020–1033, May 2006.

[100] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Sustaining cooperation in multi-hop wireless networks," in *NSDI*, 2005.

[101] N. Salem, L. Buttyan, J. Hubaux, and M. Jakobsson, "Charging and rewarding scheme for packet forwarding in multihop cellular networks," in *Mobihoc*, 2003.

[102] W. Wang, X. Y. Li, and Y. Wang, "Truthful multicast in selfish wireless networks," in *Mobicom*, 2004.

[103] H. Lim, L.-C. Kung, J. C. Hou, and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *Proceedings of Infocom*, 2006.

[104] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proceedings of Infocom*, 1999.

[105] L. McNamara, C. Mascolo, and L. Capra, "Content source selection in bluetooth networks," in *Proceedings of Mobiquitous*, 2007.

[106] A. Khelil, C. Becker, J. Tian, and K. Rothermel, "An epidemic model for information diffusion in manets," in *Proceedings of MSWiM*, 2002.

[107] W. Lou and J. Wu, "Double-covered broadcast (dcb): a simple reliable broadcast algorithm in manets," in *Proceedings of Infocom*, 2004.

[108] L. Vu and K. Nahrstedt, "Adaptive mobility-assisted data dissemination in mobile disaster/recovery environments," in *Proceedings of MILCOM*, 2007.

[109] G. Kortuem, J. Schneider, and D. Preuitt, "When peer-to-peer comes faces-to-faces: Collaborative peer-to-peer computing in mobile ad hoc networks," in *Proceedings of International Conference on P2P Computing*, 2001.

[110] Y.-B. Ko and N. Vaidya, "Geotora: a protocol for geocasting in mobile ad hoc networks," in *Proceedings of ICNP*, 2000.

[111] Y.-B. Ko and N. Vaidya, "Geocast enhancements of aodv for vehicular networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, 2002.

[112] N. Eagle and A. (Sandy), "Reality mining: sensing complex social systems," *Personal and Ubiquitous Computing*, vol. 10, pp. 255–268, 2006.

[113] P.-U. Tournoux, J. Leguay, F. Benbadis, V. Conan, M. D. de Amorim, and J. Whitbeck, "The accordion phenomenon: Analysis, characterization, and impact on dtn routing," in *Proceedings of IEEE Infocom*, 2010.

[114] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on the design of opportunistic forwarding algorithms," in *Proceedings of Infocom*, 2006.

[115] P. Hui, A. Chaintreaum, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking*, 2005.

[116] J. Leguay, A. Lindgren, J. Scott, T. Friedman, and J. Crowcroft, "Opportunistic content distribution in an urban setting," in *Proceedings of CHANTS*, 2006.

[117] J. Su, A. Chin, A. Popivanova, A. Goel, and E. de Lara, "User mobility for opportunistic ad-hoc networking," in *Proceedings of the Sixth IEEE Workshop on Mobile Computing Systems and Applications*, 2004.

[118] S. Gaito, E. Pagani, and G. P. Rossi, "Opportunistic forwarding in workplaces," in *Proceedings of ACM WOSN*, 2009.

[119] E. Miluzzo, N. D. Lane, S. B. Eisenman, and A. T. Campbell, "Cenceme - injecting sensing presence into social networking applications," in *Proceedings of Second European Conference on Smart Sensing and Context (EuroSSC 2007)*, 2007.

[120] Y.-C. Cheng, "CRAWDAD trace ucsd/cse/jigsaw/wireless (v. 2008-08-25)," Downloaded from http://crawdad.cs.dartmouth.edu/ucsd/cse/jigsaw/wireless, Aug. 2008.

[121] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *Proceedings of Mobicom*, 2004.

[122] *Data Mining: Concepts and Techniques*, 2nd ed. Morgan Kaufmann Publishers, 2001, pp. 225–276.

[123] I. Rhee, M. Shin, S. Hong, K. Lee, S. Kim, and S. Chong, "Trace ncsu gps," 2009.

[124] W. Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in *Proceedings of Infocom*, 2007.

[125] M. Balazinska and P. Castro, "Characterizing mobility and network usage in a corporate wireless local-area network," in *Proceedings of Mobisys*, 2003.

[126] J.-K. Lee and J. C. Hou, "Modeling steady-state and transient behaviors of user mobility: formulation, analysis, and application," in *Proceedings of Mobihoc*, 2006.

[127] R. Jain, A. Shivaprasad, D. Lelescu, and X. He, "Towards a model of user mobility and registration patterns," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, pp. 59–62, 2004.

[128] C. Shepard, A. Rahmati, C. Tossel, L. Zhong, and P. Kortum, "Livelab: measuring wireless networks and smartphone users in the field," in *Proceedings of HotMetrics*, 2010.

[129] J. Krumm and K. Hinckley, "The nearme wireless proximity server," in *In Proceedings of Ubicomp*, 2004.

[130] A. LaMarca, J. Hightower, I. Smith, and S. Consolvo, "Self-mapping in 802.11 location systems," in *In Proceedings of Ubicomp*, 2005.

[131] M. A. Bayira and M. Demirbasa, "Mobility profiler: A framework for discovering mobility profiles of cell phone users," *Pervasive and Mobile Computing*, 2010.

[132] "Relational algebra.
http://en.wikipedia.org/wiki/relational_algebra."

[133] J. Aslam, K. Pelekhov, and D. Rus, "Static and dynamic information organization with star clusters," in *Proceedings of the Conference on Information Knowledge Management*, 1998, pp. 208–217.

[134] "Tanimoto coefficient,
http://en.wikipedia.org/wiki/cosine_similarity."

[135] "Additive smoothing,
http://en.wikipedia.org/wiki/additive_smoothing."

[136] G. Liu and G. Maguire, "A class of mobile motion prediction algorithms for wireless mobile computing and communications," *Mobile Networks and Applications*, vol. 1, pp. 113–121, June 1996.

[137] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, M. Kyriakakos, and A. Kalousis, "Predicting the location of mobile users: a machine learning approach," in *Proceedings of the 2009 international conference on Pervasive services*, 2009.

[138] K. Laasonen, "Clustering and prediction of mobile user routes from cellular data," in *Proceedings of PKDD*, 2005, pp. 569–576.

[139] P. N. Pathirana, A. V. Savkin, and S. Jha, "Mobility modelling and trajectory prediction for cellular networks with mobile base stations," in *Proceedings of MobiHoc*, 2003.

[140] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating location predictors with extensive wi-fi mobility data," in *Proceedings of Infocom*, 2004.

[141] M. Sun and D. Blough, "Mobility prediction using future knowledge," in *Proceedings of MSWiM*, 2007.

[142] W. Gao and G. Gao, "Fine-grained mobility characterization: Steady and transient state behaviors," in *Proceedings of Mobihoc*, 2010.

[143] I. Constandache, X. Bao, M. Azizyan, and R. R. Choudhury, "Did you see bob? using mobile phones to locate people," in *Proceedings of ACM MobiCom*, 2010.

[144] L. McNamara, C. Mascolo, and L. Capra, "Media sharing based on colocation prediction in urban transport," in *Proceedings of Mobicom*, 2008.

[145] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility pattern," *Nature*, vol. 453, pp. 779–782, June 2008.

[146] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Proceedings of ACM/SIGMOBILE Workshop on Multi-hop Ad Hoc Networks: from theory to reality (REALMAN 2006)*, 2006.

[147] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *Proceedings of MobiHoc*, 2008.

[148] T. N. Dinh, Y. Xuan, and M. T. Thai, "Towards social-aware routing in dynamic communication networks," in *Proceedings of the 28th IEEE International Performance Computing and Communications Conference (IPCCC)*, 2009.

[149] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proceedings of Mobihoc*, 2007.

[150] G. Palla, I. Derenyi, I. Farkas1, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, pp. 814–818, 2005.

[151] S. C. Nelson, M. Bakht, and R. Kravets, "Encounterbased routing in dtns," in *Proceedings of Infocom*, 2009.

[152] H. Dubois-ferriere, M. Grossglauser, and M. Vetterli, "Age matters: Efficient route discovery in mobile ad hoc networking using encouter ages," in *Proceedings of Mobihoc*, 2003.

[153] E. P. C. Jones, L. Li, J. K. Schmidtke, and P. A. S. Ward, "Practical routing in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 6, 2007.

[154] Q. Yuan, I. Cardei, and J. Wu, "Predict and relay: An efficient routing in disruption-tolerant networks," in *Proceedings of Mobihoc*, 2009.

[155] C. M. L. McNamara and L. Capra, "Media sharing based on colocation prediction in urban transport," in *Proceedings of Mobicom*, 2008.

[156] Z. Guo, B. Wang, and J.-H. Cui, "Prediction assisted single-copy routing in underwater delay tolerant networks," in *Proceedings of IEEE Globecom*, 2010.

[157] S. Merugu, M. Ammar, and E. Zegura, "Routing in space and time in networks with predictable mobility," Tech. Rep., 2004.

[158] H. Cai and D. Y. Eun, "Aging rules: What does the past tell about the future in mobile ad-hoc networks?" in *Proceedings of MobiHoc*, 2009.

[159] S. Jain, K. Fall, and R. Patra, "Routing in delay tolerant network," in *Proceedings of Sigcomm*, 2004.

[160] T. Hossmann, F. Legendre, and T. Spyropoulos, "From contacts to graphs: Pitfalls in using complex network analysis for dtn routing," in *Proceedings of First IEEE International Workshop on Network Science For Communication Networks (NetSciCom 09)*, 2009.

[161] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: A social network perspective," in *Proceedings of Mobihoc*, 2009.

[162] F. Li and J. Wu, "Localcom: A community-based epidemic forwarding scheme in disruption-tolerant networks," in *Proceedings of SECON*, 2009.

[163] J. Niu, X. Zhou, K. Wang, and J. Ma, "A data transmission scheme for community-based opportunistic networks," in *Proceedings of WiCOM*, 2009.

[164] N. Thompson, S. Nelson, M. Bakht, T. Abdelzaher, and R. Kravets, "Retiring replicants: Congestion control for intermittently-connected networks," in *Proceedings of Infocom*, 2010.

[165] "Android development. http://developer.android.com/."

[166] T. Karagiannis, J.-Y. L. Boudec, and M. Vojnovic, "Power law and exponential decay of inter contact times between mobile devices," in *Proceedings of Mobicom*, 2007.

[167] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, 1998.

[168] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the SIGCOMM workshop on Delay-tolerant networking*, 2005.

[169] T. Small and Z. J. Haas, "The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)," in *Proceedings of MOBIHOC*, 2003.