CONTENT DISSEMINATION FOR STATIC AND DYNAMIC MOBILE P2P GROUPS

BY

ANJALI SRIDHAR

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Adviser:

Professor Klara Nahrstedt

ABSTRACT

In this thesis we present content dissemination protocols for static and dynamic peer-to-peer (P2P) groups. In the first part of the thesis we delve into the problem of inefficient content dissemination for static mobile device users due to cellular congestion. We propose the Sangam Framework which utilizes multiple radio interfaces and efficient scheduling policies based on device resources. In the second part of the thesis we explore solutions for efficient content dissemination for dynamic P2P groups. In order to obtain more information about mobile users, we analyze user mobility traces and attempt to construct a social P2P network. In the third part of the thesis we combine the Sangam framework and the mobility analysis in order to design the dynamic group based content distribution framework.

Solutions to the cellular network bandwidth problem have been presented by the community such as usage of alternative wireless (e.g., 802.11 WiFi) network and P2P file sharing over a group of wireless devices. However, often the theoretical and simulation approaches for file sharing within multi-radio P2P groups hide the complexity of systems and networks in real scenarios such as heterogeneity of phones in a P2P group, issues with scheduling policies within a group of devices, group formation, etc. In the first part of the thesis, we present Sangam ("confluence" in Sanskrit), an efficient cellular-WiFi group framework for file sharing where we address extensively system and network challenges in file sharing for real phone group scenarios. The Sangam framework accounts in its protocol, policy and algorithmic designs for (a) heterogeneity of phone group devices in terms of CPU and power levels, (b) different sizes and numbers of chunks in P2P part of the group-based content distribution, (c) hybrid scheduling policies for chunk dissemination within multi-radio wireless group environment, and (d) different group sizes. Sangam validation shows the impact and difference for simulations when considering real implementation of video file sharing within a cellular-WiFi

group of Android phones.

In the second part of the thesis, we attempt to explore user mobility and group formation. Smartphone users have created large virtual and physical mobile communities by virtue of their physical location and/or social relationship with other smartphone users. Social scientists observe, log via surveys, and classify these smartphone mobile communities according to their social interactions. Some social scientists also consider coarse physical features to develop socio-physical models. On the other hand, computer scientists observe, log low-level fine-grained physical features, and classify these communities according to their physical spatial and temporal measurements. However these detailed physical measurements are not well reflected in existing socio-physical models.

We present the physical refinement of the existing socio-physical model for Situated Wireless Communities (SWC) using physical phone traces of the Mobile Learning Community (MLC) at the University of Illinois at Urbana-Champaign. We take the bottom-up approach where through data mining and inference techniques, we analyze and infer from low-level physical phone features new higher-level social group behaviors such as *Physical Stable Temporal*, *Physical Stable Spatial*, *Physical Dynamic Temporal* and *Physical Dynamic Spatial* behavior. Our results show that the MLC groups exhibit new social behavior, in physical spaces across temporal and spatial dimensions which were not identified in the existing SWC socio-physical model. Our refinement approach also shows the methodology using which one can identify new socio-physical group behavior as more and more detailed physical phone features are being collected.

In the third and final part of the thesis, we will present the Dynamic Group Based Content Distribution Framework which combines the content distribution framework presented in part one and mobility analysis of users in part two. Initial P2P networks like Gnutella and Napster targeted scenarios where users were interested in popular media files. There was no social aspect of users that was considered. It was a need based network. Users willing to share the file were contacted, and based on the link quality the file was downloaded by the interested user. The overlay links did not consider social relationships, clustering of users with common preferences, etc. With the evolution of the internet and its applications, social networking has become popular. The social graph that can be constructed, using data from

social networking sites, provides a wealth of information. P2P networks like Socionet and Prometheus use this information to construct smarter overlays enabling users to query and fetch items more efficiently. With the large number of mobile phone users, P2P networks provide more advantages in terms of bandwidth, power and download time efficiency.

Using the mobility traces collected by the University of Illinois Movement (UIM) tool and the Sangam framework, the Group Based Content Distribution Framework groups different mobile users together at different locations. Using metrics such as average duration of time spent by a user at a given location, overhead of peer membership, distinct file segments offered by the peer, etc., for P2P links, a central server groups mobile users and schedules distinct file segments. The evaluation of the Dynamic Group Based Content Distribution Framework shows the bandwidth overhead and increased download time of peers in the absence of the Dynamic Group Based Content Distribution Framework.

To my parents, B.L. Sridhar and A.Gayathri Devi,
I am nothing without your unconditional love and support.

ACKNOWLEDGMENTS

I would like to thank my adviser, Professor Klara Nahrstedt, for her immense support and encouragement. Her guidance has been pivotal throughout the course of my graduate study. Working with Professor Nahrstedt on various research problems under her tutelage has been one of the most cherished periods of my life.

I would also like to thank Long Vu, friend and fellow MONET researcher for collaborating with me. Long has been a wonderful mentor and it has been a pleasure working with him. His words of support and encouragement were most helpful during the course of my graduate study.

A special thanks to Lynette Lubben for helping me arrange various academic meetings. Her cheerful demeanor and quick responses were extremely valuable. I want to thank all the past and present MONET members for the wonderful academic and non-academic interactions. A special mention goes to my friend Sukhada Palkar who has been my staunch supporter and objective critique throughout our friendship. I want to thank her for talking to me during the highs and lows of graduate school.

Words fail to express how thankful I am to my parents, B.L. Sridhar and A. Gayathri Devi. Their continuous belief in me has been tremendous. All my achievements so far including this thesis are dedicated to them. I want to thank my brother, Abhijit Bellur, the only person in the world who knows me completely and can cheer me up under the worst circumstances. Finally I want to thank my husband, Sunil Srinivasa; his kind words of support and absolute faith in my abilities have become one of the most important elements in my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

The surge of mobile devices in the electronic market today has led to its becoming a part of everyday life of people in an expedited fashion. Mobile devices have improved their processing power and networking capacity tremendously over the last decade. The rise of the smartphone industry has only led to the explosive demand and surge of mobile applications that serve different needs of users such as organization, travel, communication, entertainment etc. Applications related to communication and entertainment almost always require a constant internet connection. Studies show that there is an increasing trend in the number of streamed/downloaded videos by mobile users [3]. The cellular bandwidth is limited and as the number of mobile users increases, problems relating to bandwidth congestion causing high download times and slow internet connectivity drastically reduce the quality of service (QoS) provided to mobile users. Solutions to the cellular network bandwidth problem have been presented by the community such as usage of alternative wireless (e.g., 802.11 WiFi) networks, peer-to-peer (P2P) file sharing over a group of wireless devices etc.

P2P networks have gained immense popularity since their inception [6]. Two important contributions of P2P networks when compared to a traditional client-server architecture are 1) ability of users to share content in a distributed fashion without a central server and 2) ability of users to download files from collocated peers and achieve bandwidth savings. All peers in a P2P network form an overlay network over the underlying network infrastructure. Each peer may have a single peer link or multiple peer links depending on the type of P2P network constructed. There are two types of P2P networks - Structured and Unstructured. Structured P2P networks involve peers connected to one another over a structured topology. For example, Chord P2P network [43] is based on a distributed hash table (DHT) overlay topology. The advantage of structured P2P is efficient large scale implementation and bounded routing time for a query. However, the disadvantage is the maintenance of the structure during churn. Unstructured P2P networks do not use any algorithm or topological rules for constructing a P2P network. The links between peers are formed in an adhoc fashion. There are three types of unstructured P2P networks 1) Pure P2P - no centralized control and all peers connect to one another in a distributed fashion, e.g., Gnutella, Freenet, 2) Centralized P2P - a central node is used for bootstrapping and indexing files, e.g., Napster 3) Hybrid P2P - bootstrapping nodes are deployed in the P2P network, e.g., Kazaa. Advantage of unstructured P2P networks is the ability to construct P2P networks in an adhoc fashion and the disadvantage is the large number of flooding messages that may result due to it.

P2P network applications were initially designed for wired networks, e.g., Gnutella, BitTorrent, etc. However the internet model has been evolving at a fast pace. The sudden growth in the mobile device market has infused new interest and ideas into the P2P networking field. The faster, more powerful mobile devices introduced only reinforced the idea of smarter mobile P2P applications. As the number of mobile device users has increased, the demand for applications that can enable content distribution in a local and global fashion has also increased. Mobile devices have challenges with respect to bandwidth, connectivity, memory etc. A group of local mobile device users prefer to exchange content using the least amount of cellular bandwidth, power and download time. A P2P mobile network is able to accomplish this given the mobile device constraints and demands of users. To the list of traditional challenges of P2P networks such as session interval length, number of peers, query failure and success, time to live, lookup latency, lookup accuracy rate, bandwidth etc., we add the added mobile P2P challenges of wireless connectivity, memory, device heterogeneity, power consumption etc.

However, often the theoretical and simulation approaches for file sharing within multi-radio P2P groups hide the complexity of systems and networks in real scenarios such as heterogeneity of phones in a P2P group, issues with scheduling policies within a group of devices, group formation, etc. In the first part of this thesis, we present Sangam, an efficient cellular-WiFi group framework for file sharing where we address extensively system and network challenges in file sharing for real phone group scenarios. The Sangam framework accounts in its protocol, policy and algorithmic designs for (a) heterogeneity of phone group devices in terms of CPU and power levels, (b) different sizes and numbers of chunks in P2P part of the group-based content distribution, (c) hybrid scheduling policies for chunk dissemination within multi-radio wireless group environment, and (d) different group sizes. Sangam validation shows the impact and difference for simulations when considering real implementation of video file sharing within a cellular-WiFi group of Android phones.

Mobile P2P groups are dynamic groups that are formed or disintegrate as peers move from one location to another. The Sangam framework addresses issues relating to content dissemination in static mobile P2P groups present at a given location. The use of Bluetooth, Wi-Fi and GPS location traces have formed the basis for a number of mobility models [28], [51], [35]. These mobility models give us information about the pattern of movement followed by people in different settings - university campuses, conferences, etc. In these environments, the traces can be used to predict the future location and interaction of different users. These traces only give information about individual movements, possible future locations and interaction times.

Smartphone users have created large virtual and physical mobile communities by virtue of their physical location and/or social relationship with other smartphone users. Social scientists observe, log via surveys, and classify these smartphone mobile communities according to their social interactions. Some social scientists also consider coarse physical features to develop socio-physical models. On the other hand, computer scientists observe, log low-level fine-grained physical features, and classify these communities according to their physical spatial and temporal measurements. However, these detailed physical measurements are not well reflected in existing socio-physical models.

In the second part of this thesis, we present the physical refinement of the existing socio-physical model for Situated Wireless Communities (SWC) [44] using physical phone traces [48] of the Mobile Learning Community (MLC) at the University of Illinois at Urbana-Champaign. We take the bottom-up approach where through data mining and inference techniques, we analyze and infer from low-level physical phone features new higher-level social group behaviors such as *Physical Stable Temporal*, *Physical Stable Spatial*, *Physical Dynamic Temporal* and *Physical Dynamic Spatial* behavior. Our results show that the MLC groups exhibit new social behavior, in physical spaces across temporal and spatial dimensions, which were not identified in the existing SWC socio-physical model. Our approach also shows the methodology using which one can identify new socio-physical group behavior as more and more detailed physical phone features are being collected. The motivation for refining the socio-physical model is for the future efficient management of group mobility and development of mobile content dissemination protocols.

Initial P2P networks like Gnutella and Napster targeted scenarios where users were interested in popular media files. There was no social aspect of users that was considered. It was a need-based network. Users willing to share the file were contacted, and based on the link quality the file was downloaded by the interested user. The overlay links did not consider social relationships, clustering of users with common preferences, etc. With the evolution of the internet and its applications, social networking has become popular. The social graph that can be constructed, using data from social networking sites, provides a wealth of information. P2P networks like Socionet [32] and Prometheus [29] use this information to construct smarter overlays enabling users to query and fetch items more efficiently. With the increased number of mobile phone users, P2P networks provide more advantages in terms of bandwidth, power and download time efficiency.

In the last part of the thesis we present the design and initial implementation of the Dynamic Group Based Content Distribution Framework. The design of the Dynamic Group Based Content Distribution Framework is motivated by the presence of data that provides social context to the P2P network. Using the mobility traces collected by the UIM tool [48], the Dynamic Group Based Content Distribution Framework groups different mobile users together at different locations based on the average duration of time spent by a user at a given location, overhead of peer membership, non-overlapping file segments offered by the peer, etc. Using these metrics for P2P links, a central server groups mobile users and schedules non - overlapping file segments using the Sangam framework for content dissemination. The evaluation of the Dynamic Group Based Content Distribution Framework shows the bandwidth overhead and increased download time of peers in the absence of the Dynamic Group Based Content Distribution Framework.

1.2 Contribution

This thesis presents a content dissemination framework for static and dynamic mobile P2P groups which has been designed using the fine grained mobility traces collected from the UIM tool. The contribution of the thesis is as follows:

1. We present an efficient file sharing framework for cellular-WiFi P2P group of mobile devices.
2. The Sangam framework's design and implementation accounts for system and networking issues encountered in implementing the file sharing protocol on a group of Android devices.
3. The Sangam framework presented accounts for the heterogeneity of mobile devices in the P2P group.
4. We present a refinement of the existing socio-physical Situated Wireless Communities model using mobility traces collected using the UIM tool.
5. By applying data mining techniques to the low-level physical trace data, we infer higher level social group behavior.
6. The Dynamic Group Based Content Distribution Framework implements the formation of social P2P groups using the refined socio-physical SWC model.
7. Based on the Sangam framework, the Dynamic Group Based Content Distribution Framework implements scheduling and content distribution for dynamic social P2P groups.

1.3 Outline

We first present the Sangam framework for content dissemination in static P2P groups in Chapter 2. Chapter 3 describes the data mining and inference techniques employed to refine the socio-physical SWC model. In Chapter 4, we present the Dynamic Group Based Content Distribution Framework which uses the refined SWC model and the Sangam framework to develop content dissemination protocols for mobile P2P groups. The concluding remarks along with the future work section is presented in Chapter 5.

CHAPTER 2

SANGAM - EFFICIENT CELLULAR-WIFI GROUP FRAMEWORK FOR FILE SHARING SERVICE

In this chapter we present a framework for file sharing among a group of cellular - Wi-Fi enabled mobile devices. The framework model and assumptions are presented in Section 2.4 and the problem description is stated in Section 2.5. This is followed by a description of the Sangam architecture in Section 2.6 and download protocol in Section 2.7. The implementation details are presented in Section 2.8 followed by the evaluation in Section 2.9.

2.1 Introduction

The mobile smartphone industry has revolutionized the way that users use the internet and connect with other users. One of the most popular services provided by smartphones is downloading and playing videos. Cisco reports that it expects video traffic to account for 60% of the global IP traffic by 2014 [3]. The network speed offered by current cellular interface is only a few megabytes per second [5]. The cellular network bandwidth is becoming saturated, hence file download services are suffering from long delays, losses and even drop in connectivity in co-located group scenarios. Comparison studies in [20] and [11] show efficiency of the Wi-Fi interface in comparison to the cellular interface. The Wi-Fi interface not only provides a higher throughput but also consumes much less power than the cellular interface while downloading large files. In [41] we see that the transmissions over cellular networks and Wi-Fi access points require the largest percentage of power consumption in a device.

Furthermore we see increased growth of phones with access to multi-radio wireless infrastructure such as cellular and Wi-Fi. In this chapter, we will show that this type of multi-radio infrastructure can be very useful in military scenarios or disaster management. For example, if no wired infrastructure exists in hostile or disaster locations, a group of nodes (phones) downloads the encrypted file of interest over cellular in collaborative fashion by each phone downloading a chunk of stored file and then exchanging chunks among peer phones over Wi-Fi and/or Bluetooth as shown in Figure 2.1. It is also the case that in a military scenario, a group of peers are likely to be interested in the same file since co-located peers are deployed for a common mission. Security concerns can be overcome by adopting network coding techniques to encrypt file chunks.

Our solution, the Sangam framework leverages the proximity of the co-located group of peers and their interest in a common file to minimize the file download time, cellular bandwidth use and energy consumption. Sangam combines the cellular and Wi-Fi wireless interfaces to create a hybrid overlay network for group-based file sharing. The target scenario where Sangam protocol can be applied consists of two network technologies of different speeds and power efficiencies.

2.2 Background

The Ishare architecture and its simulation in [50] are the starting point of our exploration into the real world mobile system design and validation of the hybrid network. Ishare envisions a scenario where there are co-located mobile devices requesting the same file from a server. The file of interest is divided into segments and is downloaded by each of the peers over the cellular network. The mobile device also requests file segments from other mobile devices in the peer-to-peer network. In Ishare design, the download from the server over the cellular interface and the download from the peers over the Wi-Fi interface takes place simultaneously. Each file segment is requested at random and each peer tries to download the maximum number of chunks possible in the least amount of time.

Our original aim was to build the system with simultaneous download over both cellular and Wi-Fi interface of the mobile device. However in the real system framework we ran into many problems. For example, using the requestRouteToHost function in the Connectivity Manager class of the Android SDK [4] we were able to gain marginal success by specifying the interface to be used for a particular IP. However, it was difficult to sustain the connectivity of the cellular connection since the Wi-Fi interface is the default network to be used. The increased drain of the battery power and possible interference between the two transmissions was one of the reasons for us to switch to a sequential download process as opposed to a concurrent one. We envision a change in the hardware of mobile devices which will allow us to use simultaneous network interfaces similar to the design mentioned in [39]. The addition of the WifiP2PManager class to the Android SDK starting from Android 4.0 will enable us to use the Wi-Fi interface for peer-to-peer ad hoc connections while simultaneously maintaining our cellular connectivity [9].

2.3 Related Work

There have been many hybrid models that have been suggested in order to overcome the challenges faced by a pure client/server model over a single network interface. The hybrid model takes advantage of the fact that there are multiple peers interested in the same file content and uses a peer-to-peer network to augment the client/server model. Rimac et al. [42] develop a mathematical model to evaluate a hybrid model on the basis of churn rate, object size and upload bandwidth. The evaluations show that the hybrid model is sensitive to churn rate but performs better than the traditional model in terms of resource allocation and time convergence.

There have been many suggestions for using the base station to assist in the scheduling of P2P wireless networks. One of most common approaches, as mentioned by Hsieh and Sivakumar [24] is to use the base station for deciding the scheduling of the P2P networks which will offer better throughput due to diminished packet loss. The ICAM(Integrated Cellular and Adhoc Multicast) architecture proposed by Bhatia et al. [12] puts forth the concept of mobile proxies with the highest downlink rate that are used to forward the data to other mobile peers. The problem of fairness is addressed by UCAN (Unified Cellular and Ad hoc Networks. UCAN, proposed by Luo et al. [37], introduces the concept of secure crediting. While selfish hosts are not assumed, there is a crediting system that is used to allow proxies and relays transmitting information to be credited for their relaying work. Augmenting the existing cellular network with peer-to-peer communication is also suggested in Madsen et al. [38]. The paper proposes cooperative network of cellular and P2P networks called Cellular controlled P2P communication networks. The mobile hosts use both the cellular network for communication with the base station and a Bluetooth connection for communicating with peers. An implementation of a file download process on Nokia phones N70 running the Symbian operating system posted decreased energy consumption of about 44% with the download time being half the original time.

Sangam uses multiple interfaces on group-based mobile devices in conjunction with peer-assisted data dissemination. There have been several approaches that have used file segmentation at the server side. Hanano et al. [22] and Kang et al. [27] propose a hybrid network architecture consisting of cellular and Wi-Fi networks. The papers describe a file exchange algorithm in a peer-to-peer network with the help of a central server. Another proposed solution by Dorial et al. [17] to deal with P2P downloads is by using multicast groups. A single node is responsible for distributing a given range of file segments to mobile nodes. The file is not striped across all mobile nodes in this proposed approach. Huguenin et al. [25] propose an approach for peer-to-peer video on demand. The file segments are downloaded in such a way that the less advanced mobile devices get the latest file segments from the server which is sent to the most advanced peer. In return, the least advanced peer downloads the file segments from the most advanced peer in order. This approach ensures that all mobile devices take part in the file download process and that deadlines are met.

The concept of file segmentation and P2P download has been implemented since the start of Napster, Gnutella and BitTorrent. However, since we are using mobile devices that bring into consideration the challenges of mobility, resource constraints and time, the implementation of BitTorrent directly on mobile devices may not be practical. Ekler et al. [18] present a comparison study by implementing the BitTorrent application on Nokia's Symbian platform and the Java Micro Edition platform. The aim of the authors was to examine the realistic possibility of constructing a peer-to-peer network using mobile phones and investigating the challenges faced. Though Sangam uses the concept of file segments and peer-to-peer networks, the idea of chunk assignments based on resources of a group of devices and multiple radio interfaces is not mentioned by authors. Claudino et al. [15] implemented a file sharing application over Wi-Fi interface using the Android 1.6 platform and an existing Bluetooth file sharing application. The authors conclude that

Wi-Fi is the preferred interface to transfer files in a peer-to-peer network. These evaluations allow us to compare our results with [15] over the Wi-Fi interface.

Tysowski et al. [46] implement a file sharing protocol on the Blackberry and Windows Mobile. The mobile nodes download a fraction of the file from the server using the cellular connection and then distribute file segments among themselves using the Bluetooth interface. The results show decreased download time and throughput for a large number of peers. Another implementation is seen in the Push and Track system by Whitbeck et al. [52] for disseminating content to a large number of mobile nodes with a defined time deadline. The system makes use of the ad hoc connections in the peer-to-peer network and reduces the load on the wireless infrastructure. The push and track system does not deal with large video files but is more tuned towards traffic updates and news.

2.4 Models and Assumptions

Network Model: The network model as seen in Figure 2.1 assumes a group of co-located and cooperative mobile devices that are equipped with cellular and Wi-Fi interfaces. We call this a *group of peers* $G^w = (V, E^w)$ where $V = 1, 2...., N$ with group size $|V| = N$ and $E^w = \{(i, j)|\forall i, j \in V\}$ where (i,j) are peers connected via Wi-Fi network.[1]

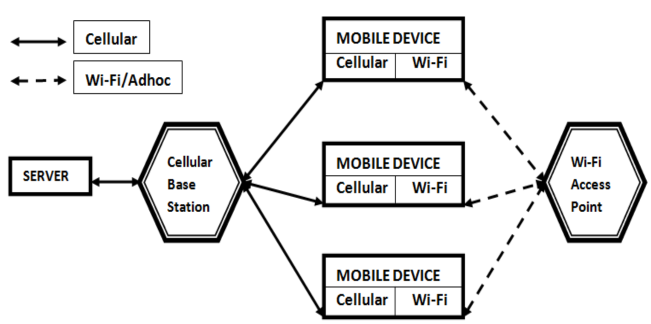

Figure 2.1: Network Model

Connections between all group peers $\{1, 2, .., N\}$ and the server S will

[1]This P2P group can be connected using an AP in ad hoc mode. It does not affect the overlay P2P connection.

be over the cellular network. The communication connections between the server S and the group devices represent edges $E^c = \{(S, i)|\forall i \in V\}$ of the network group $G^c = (S \cup V, E^c)$. At this point we assume G^c represents the group of co-located peer devices that are interested in downloading and sharing a common file. The Sangam system is represented by $G^{Sangam} = G^w \cup G^c = G$.

Data Model: Let the file of interest F be divided into m chunks represented by $\{f_1, f_2..., f_m\}$. Each of these chunks resides on at least one peer once the chunk assignment and download from S is complete.

Resource Model: The battery level and CPU speed of the device are queried and stored at the server S to assist with the chunk assignment and scheduling. The server assigns chunks based on all resources of the entire group of peers.

Table 2.1: Notations

Symbol	Description
T	Minimum File Download Time
$F = f_1, f_2, f_3, \ldots, f_m$	File of Interest
m	Maximum number of chunks in the file of interest
f_{m_i}	Index of chunk assigned to Peer i
n	Size of registered Group of Peers, $n \leq N$
N	Maximum Size of Group of Peers
E_i	Battery level of peer i (in percentage)
B^{max}	Maximum cellular and Wi-Fi bandwidth of a group of peers(incoming and outgoing)
B	Total cellular and Wi-Fi bandwidth of a group of peers (incoming and outgoing)
P_i	Number of chunks assigned to a peer i based on a given policy
R_i	Ratio of chunks assigned to a peer i based on a given policy

Mobility and Access Model: We assume that the nodes of the group are static or are moving slowly, ensuring that all peers in the group stay co-located. In this paper, we do not deal with dynamic group formations.

The model assumes that a group of peers requests a common file of interest (F) from the server S within a registration time interval $t_{arrival}$ which is approximately a few seconds long. It means, within our design, we currently do chunk assignment and scheduling to peers for a group of peers $G' \subset G$ with $V' = \{1, 2, ..., n\} \subseteq V = \{1, 2, .., N\}$ who register with the server S within an interval $t_{arrival}$. After this interval $t_{arrival}$, a new request to join the group of peers and download a file F is treated as a new group request, where server S groups all new requests from a new group of peers $G'' \subset G$ and derives new chunk assignments and schedule. The scheduled chunk assignments are derived and disseminated to all peers in the group of peers once at least two peers have registered with the server S.

Table 2.1 summarizes the notation above.

2.5 Problem Description

The problem is to download a file F from server S to all peers V of group G, since all members of the group want to share the same file F at the same time. This is an optimization problem of group file sharing. The overall goal of the group's file sharing is to minimize the download time $T(F)$ of the shared file F over group G of peers, under the constraints of bandwidth availability (B), file size (m), energy of the device i (E_i) and group size (N).[2]

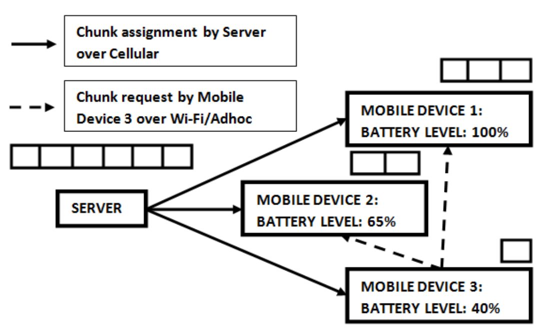

Figure 2.2: Sangam File Download

The optimization problem for minimizing the download time of a file F over a group of peers G:

[2]Bandwidth availability includes cellular and Wi-Fi bandwidth availability within the group G.

$$\textbf{min } T(F)_{\{G\}} \textbf{ subject to} \tag{2.1}$$

$B \leq B^{max}; \quad \sum P_i \leq m; \quad n \leq N; \quad E_i \leq 100$

The problem is NP complete as shown in [47], [14]. We will apply heuristic approach as follows: A peer in group G registers with the server S, specifies the file of interest F within interval $t_{arrival}$. During the registration phase, each peer declares resource information such as CPU speed (in MHz), battery level (in %) and available bandwidth (B^{cell}) to the server S.

1. The server S performs coordinated chunk assignment for group G of registered peers on the file F, based on the received resource information from a group of registered peers in the interval $t_{arrival}$; i.e., the server splits the file F into m chunks at the server S.

2. During download time over the cellular network, server S stripes file chunks $f_{m_i} \in F$ across the group of peers according to the coordinated policies, as seen in Figure 2.2 and downloads the chunks to an assigned peer.

3. During download time over Wi-Fi, peers exchange chunks $f_{m_i} \in F$ in P2P fashion, according to Bandwidth (B^{wifi}), energy (E_i) and other constraints. Peers request chunks from the P2P network using a pull based mechanism as seen in Figure 2.2.

2.6 Sangam Architecture

The design of Sangam architecture encompasses registration of the group of peers at the server S, chunk assignment and schedule by the server based on the resources constraints at each peer. The registration process involves the exchange of information between the server and group of peers. The server performs coordinated chunk assignment on the file of interest based on the received resource information from each peer. Some of the policies are as follows:

2.6.1 Coordinated Chunk Policies

The server upon completing the registration process for a group of peers G begins assigning the chunks to different peers. The common file of interest F is divided into m chunks $F = \{f_1, f_2, \ldots, f_m\}$. Each peer is assigned a range of chunks P_i based on different policies (see below) as a result of the heuristic approaches to solve the optimization problem in Section 2.5. Using the cellular connection between the server and G group of peers as a control channel, the chunk assignment and peer list is sent to all peers to assist in the file download process. Note that the chunks are scheduled at S in the FCFS registration order for all scenarios described below. We propose three different approaches to chunk assignment for the registered group of peers. We present (1) equal chunk assignment, (2) chunk assignment based on battery level, and (3) chunk assignment based on CPU speed.

1. *Equal Chunk Assignment - Homogeneous system:* This approach serves as the baseline against which we compare other approaches. The server S calculates the total number of chunks and divides them equally among the co-located group of peers. This chunk assignment is what we would expect in a homogeneous environment where all mobile phones not only have the same underlying hardware but also the same version of the operating system and radio interface optimizations. In the following equations, R_i^{equal} is the fraction of chunks assigned to each peer, P_i^{equal} is the number of chunks assigned to each peer, n is the size of the registered group of peers and m is the total number of file chunks.

$$R_i^{equal} = \tfrac{1}{n} \qquad P_i^{equal} = R_i m = \tfrac{m}{n} \tag{2.2}$$

2. *Chunk Assignment based on Battery Level - Heterogeneous system:* [11] gives us an energy model where the amount of energy used to transfer 50 kB of data over cellular network is almost twice the amount of energy used to transfer the same data over Wi-Fi network. This is the basis for the server to allocate chunks based on the amount of energy E_i in each device. For a mobile node j, whose battery power E_j is relatively lower than the power E_i of other mobile nodes i in the group G, it might be beneficial to conserve the energy and be able to view the video instead of completely draining its battery power. Another reason for other

nodes i to support this unequal chunk management is to ensure that all file chunks are downloaded and present in the G group of peers. If a mobile device j loses its battery power in the middle of the file download, all peers will be forced to request all chunks assigned to the peer j from the server S over their cellular interface.

Figure 2.2 shows a possible scenario with peers having different battery levels. Here $R_i^{battery}$ is the fraction of chunks assigned to each peer i, E_i is the battery level of peer i and $P_i^{battery}$ is the number of chunks assigned to peer i. [3]

$$R_i^{battery} = \frac{E_i}{\sum_{k=1}^{n} E_k} \qquad P_i^{battery} = R_i^{battery}\, m \tag{2.3}$$

3. *Chunk Assignment based on CPU Speed - Heterogeneous system:* In order to account for heterogeneity, each peer sends its maximum CPU speed (e.g., 800 MHz, 1 GHz) to the server during the registration process. A faster processor might be handed a larger chunk assignment from the server S as per this algorithm. Though there is an uneven chunk assignment among the peers, there is a benefit for faster mobile nodes to download a larger number of chunks from the server. The faster peers either wait for the slower peer to complete the file download of assigned chunks or end up requesting the chunks from the server directly. Here R_i^{cpu} is the fraction of chunks assigned to each peer i, C_i is the CPU speed of the peer i and P_i^{cpu} is the number of chunks assigned to peer i.

$$R_i^{cpu} = \frac{C_i}{\sum_{k=1}^{n} C_k} \qquad P_i^{cpu} = R_i\, m \tag{2.4}$$

2.7 Sangam File Download Protocol

The download protocol takes place once the peer registration and chunk assignment are completed. In order to download the entire file F, the peer has

[3] $R_i^{battery}$ is a simplified equation and an approximation, but it can be more complex by taking into account powerful models of network energy, CPU energy, application energy etc. This equation is under the assumption that cellular network energy is more costly than Wi-Fi network energy.

to go through two download phases. In the first phase, each peer i requests the server S for the chunks assigned to it in the FCFS registration order. Between the first and second phases the communication interface is changed from cellular to Wi-Fi. In the second phase, each peer i requests the G group of peers for the remaining chunks as per peer list and chunk assignment handed down from the server at the end of the registration process. In order to avoid overloading any one peer in a given time interval by requesting the same set of chunks, we use a round robin scheduling policy locally at each peer. Each peer begins phase two of the download process by requesting the peer whose ID is $(peerID+1)\ mod\ n$. The peer ID is issued by the server S at the time of registration.

A peer is unable to download all chunks from the server S and G group of peers in the first attempt for two reasons. Firstly, there is an interval of time during which the mobile node might be unreachable due to switching between cellular and Wi-Fi interfaces. Since we assign chunks based on the resource constraints of the nodes, all mobile nodes will not complete phase one at the same time. In this case, a mobile node requests the peer for the chunk that it was unable to download in phase two. Secondly, a chunk may not be present among the group of peers. The mobile node then requests the server for any chunks that might not have been downloaded by the group of peers.

2.8 Implementation

Sangam has been implemented on the Android operating system version 2.1 and above. The file download application on the client side is present in the user space and uses the Java API for socket connections. The turning off and on of the cellular and Wi-Fi interface is carried out using the *WifiManager* class of the Android API. The phones used in the experimentation are Droid, Nexus S and Nexus One. In Figure 2.3 we highlight the main modules involved in the implementation of the client Sangam framework. The server consists of a C program that accepts client connections and responds with the requested file chunk. The server handles the registration of the group of peers and the chunk assignment based on the received resource information from the group of peers.

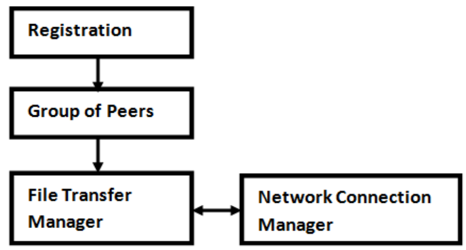

Figure 2.3: Sangam Client Architecture

1. **Registration:** Registration of a peer at the server S allows the user to specify the file of interest F, IP address (cellular and Wi-Fi interface) and port number. The registration process also involves the mobile node sending information about the battery level and CPU speed to the server. We set up TCP connections over the cellular and Wi-Fi interfaces to communicate with the server and group of peers respectively. Each peer is also assigned an ID that is used in the download process.

2. **Group of Peers Manager:** After registration with the server S, the mobile node receives the list of peers interested in the same file F along with their IP address (Wi-Fi interface) and port number. Peer management handles the storage of the IP address, port number, and P_i chunks assigned to each of peers who are interested in the same file F. The peer list includes an array of peers in G and their chunk assignment. The peer list is used to request the chunks from the group of peers in phase 2 of the download process.

3. **Network Connections Manager:** There are three TCP connections that the peer sets up during the file download process. The first two are used to request file chunks from the server and group of peers. The third one is for handling incoming file chunk requests from other peers. The connection to the server takes place over the cellular network. The other two network connections to the peers take place simultaneously over the Wi-Fi interface. The current cellular and Wi-Fi connection status is maintained by the Network Connections Manager module.

4. **File Transfer Manager:** The File Transfer Manager is responsible

for carrying out the download protocol. It is responsible for switching interfaces between cellular and Wi-Fi in a timely fashion. It keeps track of the number of chunks received from the group of peers and services requests according to the chunk assignment handed down from the server. It also ensures mutual exclusion when writing and reading chunks from a given output file. The video file is present at the server S and divided into chunks of size 2048 bytes.

2.9 Evaluation

The evaluation of the Sangam framework was carried out on Android phones having an operating system version of 2.1 or greater. We used five Droid phones which use the Verizon network for cellular connectivity [4] to the server. To demonstrate the heterogeneity of devices we also used the Nexus One device with a T-mobile cellular connection. The Verizon network uses CDMA technology and offers an average speed of 848 kbps [7]. T-mobile uses GSM technology with an average speed of 7.2 Mbps [1][5] . A maximum of five phones were used during this evaluation and files of size 2.2 MB and 3.9 MB were requested. We used 802.11x Wi-Fi access point for connectivity among the mobile devices. The Wi-Fi network used WPA2 protocol for authentication. We carried out file download evaluations with chunk sizes varying from 256 bytes to 4096 bytes. We found 2048 bytes to be the optimal size which balanced the limited memory requirements of the mobile device and the IP packet size.

In our evaluation, the **File Download Time** $T(F)$ is the most important metric. Our goal is to minimize the time required by all peers to download the complete file by solving the optimization problem in Section 2.5. In some cases, when we assign chunks based on battery level, our aim is to minimize file download time under the constraint $E_i \leq E_{available}$ where E_i is the energy required by phone i to complete the file download process and $E_{available}$ is the phone's total energy at the time of the file download

[4] At the time of this experiment, only 3G cellular connectivity was possible.

[5] The average download speed on a 4G/LTE network is 2.2 Mbps [2] which is lower than a Wi-Fi network (6 Mbps and above). In this case, using the hybrid network will results in greater cellular bandwidth and energy savings as compared to the decrease in download time.

process. Each resulting plot has the baseline cellular download time, server download time over the cellular network and peer (P2P) download time over the Wi-Fi interface. The *Baseline Download - Cellular* is the time taken to download the entire file from the server over the cellular network. The *Server Download - Cellular* is the time taken by a peer to download the assigned chunks from the server over the cellular link and *Peer Download Wi-Fi* is the time taken to download the remaining chunks from the group of peers over the Wi-Fi interface (P2P download). The average download time for Baseline measurements where 5 peers simultaneously request the entire file of size 2.2MB from the server S over the cellular network was found to be 551.375 seconds with a standard deviation of 38 seconds over three trials. We have conducted baseline measurements where 5 peers simultaneously request the entire file of size 2.2 MB from the server S over the cellular network. The average download time was found to be 551.375 seconds with a standard deviation of 38 seconds over three trials. The download time per file chunk of size 2048 bytes was 0.494 seconds.

2.9.1 Scenario 1 - Equal Chunk Assignment

In our first set of experiments we demonstrate the advantage of the hybrid network using the Sangam framework. We assign file chunks equally to all the peers in the group G. The group of peers G request a file F of size 3.9 MB from the server S. We carry out this request for 5 registered peers. In Figure 2.4, each peer is downloading a smaller number of file segments over the cellular interface. The majority of the chunks are now downloaded from the group of peers over the Wi-Fi interface. The time taken to download the file over the hybrid network is 445 seconds and the time to download the same file over the cellular network is 1007 seconds. Hence the download time decreases by 55.8%.

2.9.2 Scenario 2 - Chunk Assignment based on Battery Level

In our next experimental scenario we consider chunk assignment based on the battery level. The peer with the greater battery level is assigned a larger share of file chunks. In this case the metric that we choose to minimize is

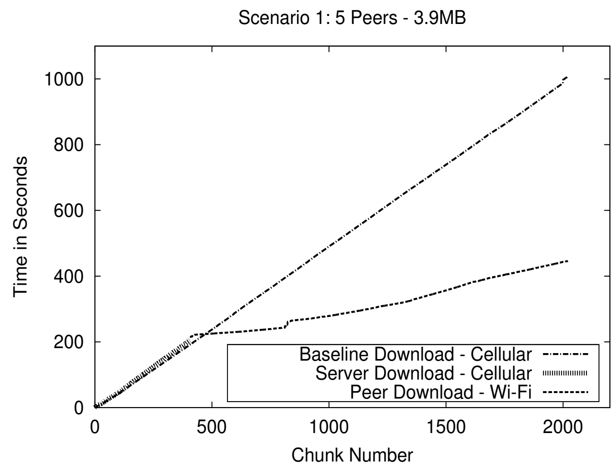

Figure 2.4: Scenario 1: File Download Time with Equal Chunk Assignment Approach

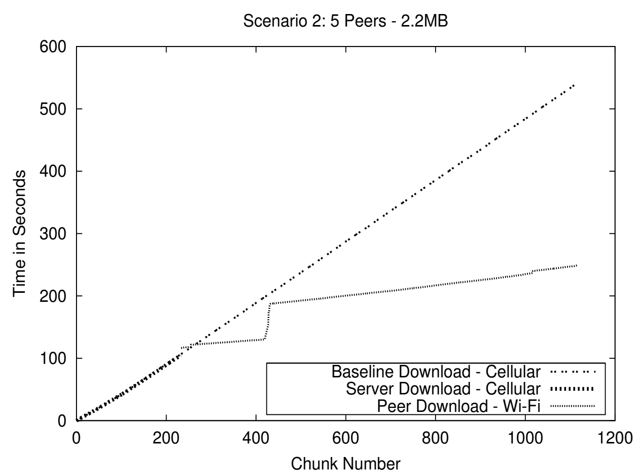

Figure 2.5: Scenario 2: File Download Time for Chunk Assignment Approach based on Battery Level

the file download time under the constraints of the battery consumption of the peers. We consider a hybrid network with 5 peers of unequal battery levels requesting a file of size 2.2 MB as seen in Figure 2.5. Table 2.2 also demonstrates that peers with a lower battery level suffer a greater decrease in battery power than the peers with a higher battery level over the course of the file download process.

In [11] we see that the energy required to transfer 50 kB over 3G is 12.5J as compared to 7.5J over Wi-Fi. The drop in energy if the entire file is

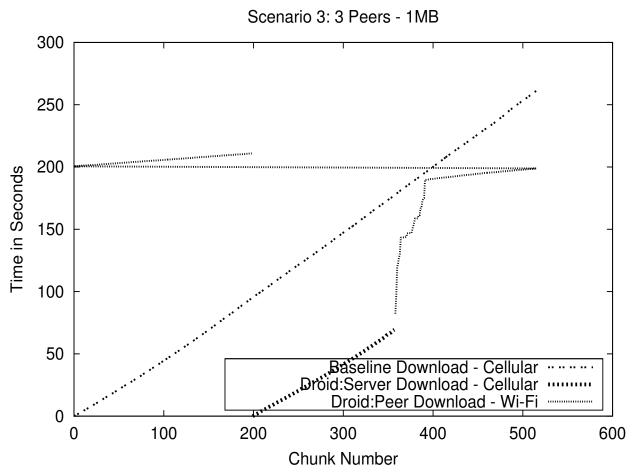

Figure 2.6: Scenario 3: File Download Time for Chunk Assignment Approach based on CPU Speed for the Droid

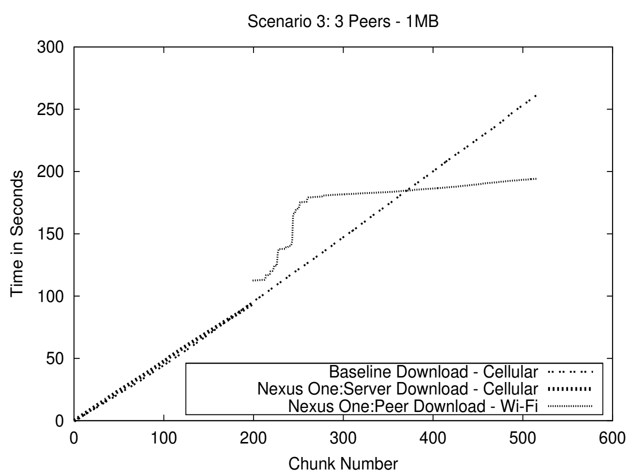

Figure 2.7: Scenario 3: File Download Time for Chunk Assignment Approach based on CPU Speed for the NexusOne

downloaded over the cellular network is 563.2J. However if the hybrid network is used, the drop in energy is 384.32J. In Figure 2.5, the peer with a battery level of 40 is assigned a smaller share of chunks that need to be downloaded from the server as compared to the chunk assignments handed out to peers with higher battery levels. Due to the limited time spent downloading the chunks over the cellular interface, peers with a lower battery level do not suffer from a large drop in power level. Using the hybrid network we get a decrease in download time of 53.7%.

Table 2.2: Scenario 2: Download Time for 5 Peers

Peer ID	Initial Battery Level	Final Battery Level	Download time(s)
0	50	50	251.351
1	40	20	238.686
2	80	80	285.429
3	40	20	288.402
4	30	30	241.465

2.9.3 Scenario 3 - Chunk Assignment based on CPU Speed

In order to account for the heterogeneity of nodes we assign chunks to the group of peers based on the CPU speed level of the mobile devices. We use two Droids that use the Verizon cellular network and one Nexus one that uses the T-Mobile cellular network. The CPU level of the Droid phones are 800 MHz and that of the Nexus One is 998.4 MHz. All 3 peers request a file of size 1 MB from the server. The Nexus One is assigned a greater share of the file segments due to its higher CPU speed. Figure 2.6 of the Droid shows a certain delay in switching interfaces and starting the download from the group of peers. We notice the speed of download of the remaining chunks from the group of peers by the Nexus One device is extremely fast (refer Figure 2.7). The Droid (Figure 2.6) is able to leverage the faster Nexus One device in order to complete its download. However, the faster CPU power of Nexus One was not able to overcome the bottleneck of the throughput of the cellular network. As we can see in Figure 2.7, the cellular download line of Nexus One mimics the baseline plot in spite of having a greater processing power. Using this chunk assignment scheme we are able to get a decrease in download time of 22%.

2.9.4 Conclusion

We recognize the need to exploit the multiple interfaces on a mobile device to improve its download capacity. The Sangam framework uses the cellular and Wi-Fi links to improve the download time and reduces congestion of cellular networks. Using efficient file segmentation techniques, the Sangam framework aims to optimize the file download time and resource usage of the mobile devices in the P2P network.

The Sangam framework assumes that the groups of mobile devices are present in one location and have multiple radio interfaces. However we know that users move from one location to another over the course of a day. In order to account for the mobility of groups of users, we need to study different user mobility patters. In the next chapter, we use the mobility traces obtained from the UIM tool to help us understand group mobility characteristics. We apply grouping algorithms to the UIM data set to discern physical temporal and physical spatial behavior. Based on this observation we refine the existing socio-physical Situated Wireless Communities model and attempt to infer fine-grained social and physical context of the users. Using this knowledge of group mobility, we attempt to design a framework for content dissemination for dynamic P2P groups.

CHAPTER 3

PHYSICAL REFINEMENT OF SOCIO-PHYSICAL MODELS FOR MOBILE LEARNING COMMUNITIES

In this chapter we attempt to refine the socio-physical SWC model using low-level physical UIM mobility traces. Using this refined socio-physical model, we infer higher level social group behavior. In Section 3.2 we present the assumptions and methodology used in refining the SWC model and in Section 3.3 we present our inference framework and results.

3.1 Introduction

Traditional virtual social communities consist of people connected online unimpeded by physical barriers (e.g., Facebook communities). However, as smartphones are becoming ubiquitous, social scientists and computer scientists are interested in social interactions/relations of users and efficient content delivery to users who use smartphones and tablets and create wireless communities. We observe that students tend to meet their class project groups at the computer lab or homework study groups at the student lounge, go to class, etc., at certain time intervals during the day. We also observe that groups of students moving together from one location to another due to common class schedules or group projects exhibit dynamic group membership behavior. Due to this grouping of students, a number of locations may experience an unusually large number of Wi-Fi users during certain time intervals during the day. Hence the design of Wi-Fi infrastructure and internet content distribution protocols would greatly benefit from understanding the group gathering patterns as well as group mobility patterns.

The social scientists, based on surveys, are interested in classifications of these wireless communities, and devise social models based on their social interactions. Social scientists use **a coarse physical and social classification** to create socio-physical models. An example of the socio-physical model is Situated Wireless Communities [44]. On the other hand, computer scientists are interested in physical properties of phones such as time, location, movement and mobility management of phone users in order to better understand mobility patterns and hence improve content distribution and delivery to these wireless communities. Computer scientists use **fine-grained sensory features classification** to create physical temporal and spatial mobility and activity models of individual users. Hence, the main problem addressed in this paper is how to refine the existing socio-physical models using the fine-grained sensory information from phones, and discover new socio-physical group behaviors.

There have been mobility models in the past that have attempted to classify user behavior based on physical and social attributes. A mobility model based on common social or physical context is proposed by Sun et al. [44] called Situated Wireless Communities (SWC). Wireless communities are formed by wireless networks ranging in size from small local communities using LANs to digital cities such as a Helsinki village. Based on physical and social contextual information that is provided by the mobile device of the user, there are four major classifications [44] - Nomadic Clan, Nomadic Mix, Settled Clan, and Settled Mix as seen in Figure 3.1. This socio-physical model has very coarse physical domain granularity as mentioned above. Hence it is not possible to accurately define and label group behavior.

Hui and Crowcroft [26] present a set of mobility models based on social relationships among users derived from real life Cambridge data sets. Social relationships between people are measured by how often two people meet and how long they stay with each other. Four relationships are defined based on this: Community, Familiar Strangers, Strangers and Friends. Though this model makes use of real traces, the granularity of physical location is very coarse. The classification is based on two parameters, familiarity and regularity, which cannot be used to conclusively conclude the given classification tags.

The authors in [13] present the Home-cell Community-based Mobility Model (HCMM) where people are attracted towards their community's location. A node is attracted to an external cell based on the social connections that the node has with the home-cell nodes. The mobility classification of users in this model considers only the social domain. The physical location is very coarse where each user is present in a grid with several other users.

PHYSICAL \ SOCIAL	STABLE	DYNAMIC
DYNAMIC	NOMADIC CLAN	NOMADIC MIX
STABLE	SETTLED CLAN	SETTLED MIX

Figure 3.1: Situated Wireless Communities

In the past low-level sensors have been used to detect precise physical movement of users. Some of the sensors that can be used to gather information about the user's activity are microphones [45], accelerometers, compass measurement [16], magnetometers [23], Wi-Fi, Bluetooth [48], [36], cameras, ambient light, proximity, GPS, gyroscope, etc. By using a combination of the previously mentioned sensors, the activity of the user can be inferred based on patterns detected in the collected sensor data [34], [40]. Based on this activity inference, a number of suggestions regarding the health of the user, shopping preferences, etc., are made through mobile phone applications [30]. In the above mentioned models, the user's location was detected and activity could be inferred, but the social domain classification of groups of users was absent. The mobility models developed tried to classify the movement of users based on previously traversed paths, aggregated path suggestion from a central server, predictive path and contact based on past history of the user. In summary, none of the above social mobility models or sensing inference models are able to present the refinement of socio-physical models of mobile learning communities.

3.2 Assumptions and Methodology

3.2.1 Assumptions

Online social communities such as Facebook and Twitter communities lack physical contextual information while physical wireless communities such as smartphone mobile communities lack social contextual information. However, it is necessary to share both social and physical contextual clues to have (a) an effective social interaction, (b) an efficient content distribution among groups, and (c) an effective mobility management of groups and crowds.

In this chapter, we assume the socio-physical model of *Situated Wireless Communities (SWC)* [44] as shown in Figure 3.1. The physical location context is identified by the user's connectivity to an area of interest. Settled and Nomadic user groups refer to the static and dynamic physical location context of the user. Settled users will be in one area of interest such as university campus, where nomadic users will move from one area to another area of interest. Clan and Mix user groups refer to the static and dynamic social context of the users. Clan refers to a group of users who know one another; users are acquaintances or friends. Mix refers to a group of users who do not know one another. However, these users may be in the process of getting acquainted or may have a common social context such as being students at the same university, or local community.

To achieve refinement of socio-physical models using low-level physical phone features, we will use the socio-physical SWC model. We will show that using fine-grained cyber-physical sensory information, we can identify four new instances of social group behavior not identifiable in the SWC model. The four instances are: *Physical Stable Temporal*, *Physical Stable Spatial*, *Physical Dynamic Temporal* and *Physical Dynamic Spatial* behaviors. In order to observe *Physical Stable Temporal* and *Physical Stable Spatial* behavior, we need to observe the density of users (group size) at different locations. Identifying the locations that are frequented by a large number of users specifies *Physical Stable Spatial* behavior. In the case of *Physical Dynamic Temporal* and *Physical Dynamic Spatial* group observation, we need to identify proximity and movement of groups of users.

For example, identifying the locations that users frequent before coming together characterizes *Physical Dynamic Temporal* behavior. Identifying movement of groups of users in a given time interval defines *Physical Dynamic Spatial* behavior. We propose to use mobility traces that have been collected using Bluetooth and Wi-Fi scanning to refine the SWC model. These traces provide fine granularity of physical movement and identification of proximal users. Using this physical location and proximity information, we extend the SWC model classification. By refining the SWC model we attempt to observe and infer the above instances of social group behavior.

Sample Bluetooth Trace

```
t=03-25-2010 18:01:22
MAC:m_i, cls=5767168, rssi=-78, name=T-Mobile myTouch 3G, devtype=other
```

Sample Wi-Fi Trace

```
t=05-12-2010 12:07:01
SSID: UIUCnet, BSSID: a_i, capabilities: [WPA2-PSK-CCMP], level: -38, frequency: 2412
```

Figure 3.2: Sample Bluetooth and Wi-Fi Trace

For identification of the four new instances of group behavior in the SWC model, we will use sensory information such as Wi-Fi access point BSSIDs, Bluetooth MAC addresses and timing information mapped to location, contact duration and contact time. These sensory information is taken from mobility traces acquired by the University of Illinois Movement (UIM) [49] scanning tool of the Mobile Learning Community at the University of Illinois at Urbana-Champaign. The mobility traces were collected using smartphones carried by 123 undergraduate students, graduate students and faculties over the course of 6 months from May 2010 to August 2010.

The UIM tool enabled Bluetooth scanning every 60 seconds and Wi-Fi scanning every 30 minutes. Each Bluetooth trace as shown in Figure 3.2, has a timestamp t_i followed by the device name, device MAC address m_i and the signal strength of each phone present in the vicinity of the user. Each Bluetooth MAC m_i represents one user and each user is identified by the integer u_i mapped to its Bluetooth MAC m_i. The Wi-Fi scan was used to detect the MAC address of the Wi-Fi Access Point. The Wi-Fi trace as shown in Figure 3.2 contains a timestamp t_i followed by the SSID, BSSID a_i, signal strength and frequency of the Access Point. We restrict our analysis to on-campus locations and participants of the UIM tool experiment. We apply the clustering algorithm presented in [48] to assign a location L_i to each group of Wi-Fi BSSID addresses logged at each t_i by the UIM tool. Hence, given a time t_i, we are able to infer the location L_i of the user u_i provided the user has a Wi-Fi trace at time t_i. Using this physical location and proximity information, we refine the SWC model in an attempt to observe and infer the above instances of social group behavior.

3.2.2 Methodology

In this chapter, we take the mobility traces collected by the UIM tool and refine the SWC socio-physical model by Sun et al. [44], shown in Figure 3.1, in order to introduce physical spatial and temporal aspects of socio-physical group behavior. To refine the SWC model using fine-grained cyber-physical sensory information and infer new socio-physical group behavior, our approach uses the following methodology. In **Phase 1**, we assume clean records from phones but we still need to clean up (filter) the trace data for our socio-physical grouping analysis. We filter and remove Wi-Fi traces which contain access point BSSIDs located off campus since we are interested in socio-physical on-campus group behavior of mobile learning communities.[1] In **Phase 2**, we attempt to assign a location to each Wi-Fi record using clustering techniques similar to [48]. Since any location (classroom, student lounge, etc.) contains multiple Wi-Fi access points, Wi-Fi scanning results in recording a location's multiple access points in a single record. Each Wi-Fi record consists of a timestamp followed by a list of access point information (SSID, BSSID, signal strength, etc.).

Using the procedure in [48] we assign locations to groups of frequently occurring Wi-Fi access point BSSIDs. In **Phase 3**, we apply mining techniques such as density based clustering (DBSCAN) [19] and Apriori algorithms [10] to modified Wi-Fi traces post location assignment and modified Bluetooth traces post clean up. We then categorize the group behaviors inferred into four different physical classes according to the socio-physical model: *physical dynamic temporal* (social groups are moving in time), *physical dynamic spatial* (social groups change in space), *physical stable temporal* (social groups stay together in time) and *physical stable spatial* (social groups stay together in space) behavior. In this section we provide a detailed explanation of the procedure adopted in Phase 1 and Phase 2 of our methodology. Table 3.1 summarizes the notations used in this section. Section 3 will provide a detailed explanation of the procedure followed in implementing Phase 3 of our methodology.

1. **Phase 1**: The input for Phase 1 are all Wi-Fi and Bluetooth traces

[1] This phase can be expanded towards any data filtering algorithms and inclusion of relevant physical phone features and contextual information that are of importance to inference of users' activities, group behaviors and/or crowd formations.

Table 3.1: Notations and Terminology

Notation	Terminology
t_i	record timestamp of record i
u_i	Identification of user i
m_i	Bluetooth MAC address of the phone i
a_i	BSSID of the access point i
F	Set of good Wi-Fi records
W	Set of all Wi-Fi records
GR	Good Ratio metric
B	Set of BSSIDs
G	Similarity graph
C	Candidate Cluster Set
CF	Final Location Cluster
L	Location Map

as shown in Figure 3.2. We create a file that maps each access point BSSID to an integer value. This enables easy recognition and analysis of groups of access points during the mining process. We only map the access points on campus whose SSID is “UIUCnet”. We extend this mapping process to all the Bluetooth traces.Each MAC address represents one user and is mapped to an integer u_i.[2] The output of Phase 1 are all modified Wi-Fi and Bluetooth traces.

2. **Phase 2**: In the next step we assign a location to each Wi-Fi record following steps similar to [48]. The input for Phase 2 are all modified Wi-Fi traces from the output of Phase 1. The following steps explain the location assignment procedure in more detail:

3. **Obtain a good record set:** To obtain the set of good Wi-Fi records F from the set of all Wi-Fi records W, we apply the Frequent Item Set algorithm [10] to calculate the mean and standard deviation of support values[3] of the BSSID pairs present in each record. All records W are listed in decreasing order of their good ratio GR (ratio of standard deviation to the mean of support values). The higher the good ratio

[2]There can be multiple Bluetooth devices carried by a single user which may lead to the over estimation of the density of users at a given location. To prevent this we only record the MAC address of devices which have a device name such as “T-Mobile”, “G1”, etc.

[3]The support value of each candidate item set is the frequency count of occurrence of the given candidate item set in a global item list.

GR value, the greater the frequency of occurrence. From the ordered list of records W, we add a record to the empty set F till the size of BSSID's recorded $|B|$ is equal to the total number of BSSIDs recorded.

4. **Construct a similarity graph:** In the similarity graph $G(V, E)$ to be constructed, each good record F_i is a vertex v_i. If the Tanimoto coefficient [8] (refer Equation 3.1) of similarity between good records F_i and F_j is greater than 0.1, an edge e_i is drawn between the pair of vertices v_i and v_j. The coefficient of similarity between two records F_i and F_j represents the number of common Wi-Fi BSSIDs and suggests that both Wi-Fi records were collected at the same location.

$$T(F_i, F_j) = \frac{|F_i \cap F_j|}{|F_i| + |F_j| - (|F_i \cap F_j|)} \tag{3.1}$$

5. **Apply the Star Clustering Algorithm:** From G constructed above, order the set of vertices v_i in decreasing order of edges. Cluster each vertex v_i along with all vertices v_j that it has an edge forming a cluster C_i. Remove all vertices v_j repeat this clustering procedure till the ordered list is empty. The list of clusters is called the Candidate Cluster Set C. By collapsing clusters (records) with overlapping BSSIDs, we obtain a single list of BSSIDs that represent a location.

6. **Merge and Assign:** Each cluster from the Candidate Cluster Set C_i is merged with another cluster C_y if all the records in cluster set C_i are a subset of the records contained in cluster set C_y. Each cluster CF_i in the final set CF is a set of records and is stored as the set of BSSIDs present in each of the records. Each cluster CF_i in the final cluster set CF is assigned an integer representing a location L_i. To assign locations to each Wi-Fi record, BSSIDs of each Wi-Fi record are compared with each cluster CF_i and the cluster with the highest Tanimoto coefficient of similarity is assigned to it.

In the following section, using the location assigned Wi-Fi records from Phase 2 and modified Bluetooth records from Phase 1, we apply mining techniques to identify new socio-physical grouping behavior in the spatial and temporal domain of static and dynamic groups.

3.3 Inference Framework

In this section we describe the grouping algorithms applied to the UIM data set to discern physical temporal and physical spatial behavior in Phase 3. Based on this observation we refine the SWC socio-physical model and attempt to infer fine-grained social and physical context of the users. Refer to table 3.2 for a list of notations used in this section.

Table 3.2: Notations and Terminology

Notation	Terminology
D_{temp}	Set of k largest groups identified at a given location L_i on a given month m_i and day d_i
d_{k_i}	Day on which group D_{temp_i} occurs at location L_{k_i}
h_{k_i}	Hour interval (e.g 11AM - 12PM) during which group D_{temp_i} is present at location L_{k_i}
BT_{i_j}	Set of MAC addresses detected by user j of group D_{temp_i} during the hour interval h_i
BT_{temp_i}	The intersection of BT_{i_j} of different users of group D_{temp_i} during the hour interval h_i
D_{spa}	Set of k largest groups identified on a given month m_i, day d_i and hour h_i
L_{spa}	Set of locations visited by groups in D_{spa}
$U_{L_{j\,i}}$	Set of users present at location L_{spa_j} during the hour h_i
BT_{spa_i}	The intersection of BT_{i_j} of different users of group D_{spa_i} during the hour interval h_i
U_{dyn}	Set of users identified at a given location L_i on a given month m_i, day d_i and hour h_i
CI_i	Candidate item set of size i where i is 2,3,4 and 5
O_{s_i}	List of locations, months, days and hours of the occurrence of each of the candidate item sets in C_{s_i}
CI_{sub}	Subgroups derived from the main candidate item set CI_{s_i}

3.3.1 Physical Stable Temporal Behavior

To explore the physical stable group behavior, we identify and study static groups at different locations over time. In the three dimensional space of location, time and density, we set the location to be a constant (Figure 3.3). We observe the density of users over the course of a day at different

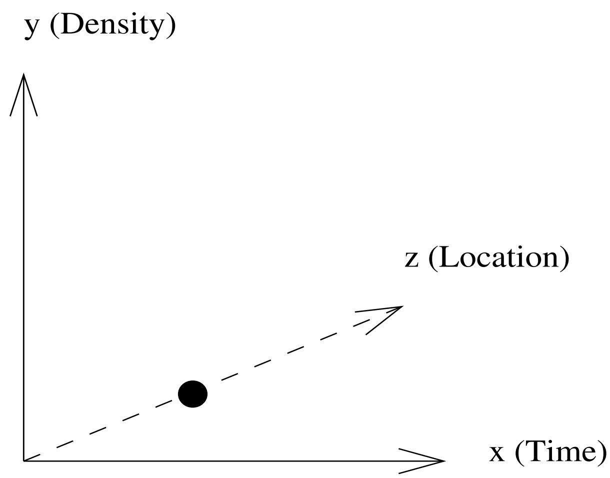

Figure 3.3: Physical Stable Temporal Behavior

locations. Using this graph, we try to understand the time intervals during which content must be disseminated in order to reach the maximum number of people. The high density points at different locations can be recognized to facilitate efficient content dissemination. For example, when we observe the density of users at a university coffee shop, we notice that the largest density of users is observed at 12PM and 5PM. This corresponds to lunch time and end of classes for university students.

Algorithm

We find the density of users at a given location during a given interval of time using the density based clustering algorithm (DBSCAN) [19]. The following steps detail the procedure:

1. **Step 1:** Our first step in applying DBSCAN is to define the Tanimoto Similarity Coefficient [8] as the distance metric for any two users. We define four attributes for each user - location L_i, month m_i, day d_i and hour h_i. The greater the similarity coefficient between two users, greater are the number of common attributes and hence lesser is the distance.

2. **Step 2:** To identify a cluster in DBSCAN, we decide if users belong to the same cluster using the previously defined distance metric and if

the size of the cluster is large enough to be relevant in our analysis. To record the size of groups at a given location over the course of a day, we find all users present at a given location L_i on a given month m_i and day d_i without setting the hour h_i attribute.

3. **Step 3:** We find k largest groups of users D_{temp} and each of these D_{temp_i} groups visit a location on a given day d_{k_i} of the experiment. On each of the d_{k_i} days of the experiment, we find the location L_{k_i} which is visited by the largest group of users in the set D_{temp}.

4. **Step 4:** To find the density of users present at a given location L_{k_i}, we use the Bluetooth traces to query the MAC addresses of all mobile devices seen by each of the users in a given group. The hour h_{k_i} during which each user j is present at the given location L_{k_i} is found using the Wi-Fi traces. We look at each hour h_{k_i} in the Bluetooth traces during which the user is present at the given location l_{k_i} on the given day d_{k_i}. The set of Bluetooth MAC addresses recorded by each user j in the hour h_{k_i} is given by BT_{i_j}. The intersection of all Bluetooth MAC addresses recorded by each user j in the group D_{temp_i} during the time interval h_{k_i} is given by BT_{temp_i}. The size of BT_{temp_i} is the density of users present at the given location L_{k_i} during a given time interval h_{k_i}. By finding the density of users detected by different members of the group D_{temp_i} at different hour intervals h_{k_i}, we are able to trace the number of users visiting a given location over the course of a day.

Inference

The density based clustering algorithm can be used to obtain a list of days, where the total density of users at certain locations is greater than a given threshold. In Figure 3.4 we observe that the density at location 182 is highest at 5PM and 7PM which suggests that location 182 is probably a student lounge. Using this data we can identify locations where there is a congregation of a large number of users at certain hours during the day. At university campus locations, it might be useful to observe locations where delay tolerant content such as university news updates can be propagated at opportune

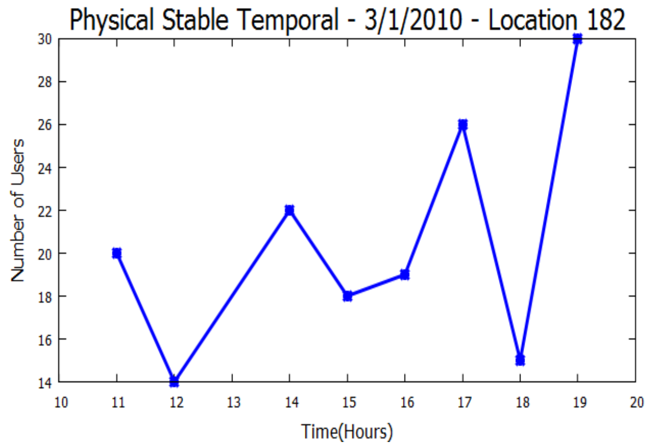

Figure 3.4: Physical Stable Temporal Behavior

moments over an interval of time to enable maximum content reception and propagation.

3.3.2 Physical Stable Spatial Behavior

To observe physical stable spatial behavior, we observe static groups at different locations at a given time. In the three dimensional space of location, time and density, we set time to be constant (Figure 3.5). By observing the density of users at different locations, we try to understand the deficiencies in internet infrastructures. It is often the case that certain locations at a given hour of the day are subject to a large number of people. For example, campus coffee shops are generally crowded during the lunch hour. Internet users are forced to deal with heavy network congestion during these time intervals. We can identify locations where users experience heavy network congestion by observing physical stable spatial behavior.

Algorithm

To observe physical stable spatial behavior of groups, we look at the density of groups at a given time and location. We use the DBSCAN algorithm as described above with a few modifications as detailed below:

1. **Step 1:** Follow steps 1 and 2 from Section 3.3.1.

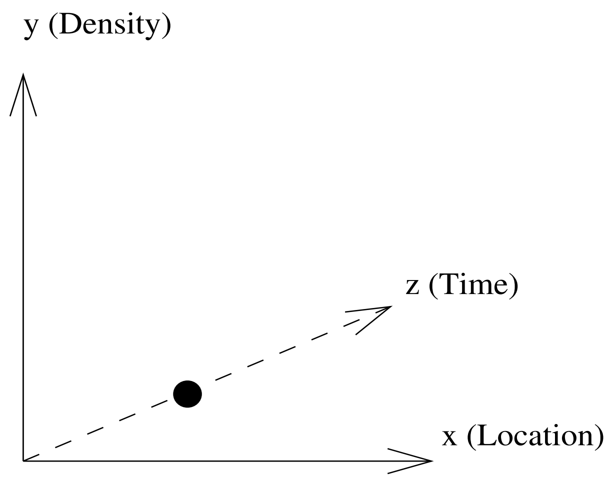

Figure 3.5: Physical Stable Spatial Behavior

2. **Step 2:** In order to view spatial domain behavior of groups, for any given month m_i and day d_i, we fix an hour h_i in the day during which we observe the density of users at different locations. The k largest group observed is denoted by the set D_{spa}.

3. **Step 3:** We find all locations L_{spa} that are visited by each set of users in D_{spa} on a fixed month m_i, day d_i and hour h_i. Each location L_{spa_i} has a set of users U_{L_i} associated with it. To find the density at the given location L_{spa_i}, we employ the method described in Section 3.3.1. We use the Bluetooth traces of each of the j users present in set U_{L_i} to detect the number of Bluetooth MACs recorded during the fixed hour h_i. The set of Bluetooth MACs detected by each user j is given by BT_{i_j}. The intersection of all Bluetooth MACs detected is given by BT_{spa_i}. The size of this set gives us the density of users present at a given location L_{spa_i} on a given month m_i, day d_i and hour h_i.

Inference

The results in Figure 3.6 show increased density of users at location 185 and 552. It is often the case that in crowded locations, wireless network performance decreases because the wireless infrastructure has not been constructed to withstand high density of users. For example, by looking at this data, we

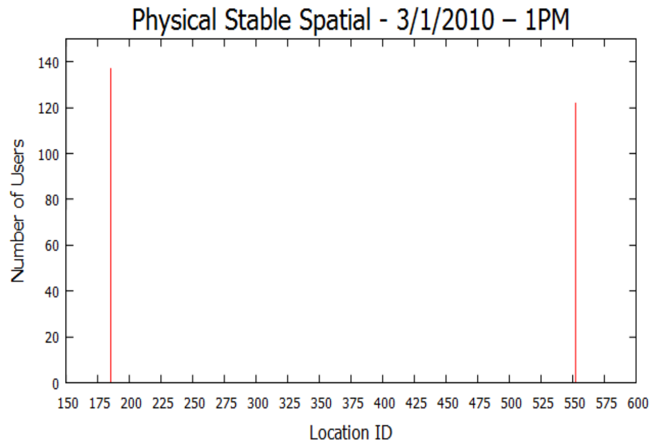

Figure 3.6: Physical Stable Spatial Behavior

know that Location 552 needs to be able to withstand 122 users. By analyzing these models, we are able to better anticipate user's needs. We can also infer the type of location given the density of users. For example if the density of users at a location does not increase beyond 30, it may be a classroom or a conference room. If the density of users is above 50, it is most probably a student lounge, classroom, computer lab, etc., in educational environments.

3.3.3 Physical Dynamic Temporal Behavior

Users move from one location to another either as a single entity or as part of a group. The association of a user with a given group changes with time. In order to observe physical dynamic temporal behavior, we identify the different mobility paths of users belonging to a group. The "hot mobility paths" of the users over time and location help us determine the different locations that users pass through before forming a group. By observing physical dynamic temporal behavior of different groups, we are able to design efficient content dissemination protocols. Based on the different locations that users pass through, the content dissemination server is able to schedule distinct chunks for downloading and sharing among the different group members. For example, a group of students working on a common class project may have a few classes in common. If some members of the group have a common "hot mobility path" for some part of the day, the server can schedule exchange of content between some members of the group before the entire group is formed at a location in the future.

Algorithm

To observe physical dynamic temporal behavior of groups, we use the Apriori algorithm [10] to identify locations called Points of Interest (POI) where two or more groups meet and record the time interval during which group membership stays constant. The procedure followed is detailed below:

1. **Step 1:** To apply the Apriori algorithm we list the set of users U_{dyn_i} present together at a given location L_i, on a given month m_i, day d_i and hour h_i of the experiment. We define the global list of occurrences to be the list of all users U_{dyn}. The Apriori algorithm involves generating candidate item sets CI and removing infrequent candidate item sets using the minimum support value.

2. **Step 2:** We start with candidate item sets CI_2 of size 2 and iterate through each user set U_{dyn_i} in the global list. From each set of users U_{dyn_i} we find all possible non-duplicate two-user candidate item sets. If the item set has already been recorded, we increment the occurrence count else we add the new item set CI_{2_i} to the candidate item set CI_2.

3. **Step 3:** We find candidate item sets for three users CI_3, four users CI_4 and five users CI_5. For each group such as CI_{2_i}, we list the location, month, day and hour of each of its occurrences represented by O_{2_i}.

Inference

In Figure 3.7, users 7,45 and 98 are present at location 66 between 11AM and 12PM. Users 45 and 7 move to location 182 where User 98 joins them at 3PM from location 171. Since the three users meet only after 3PM, it is possible for users 45 and 7 to exchange content before the arrival of User 98. Since user 98 visits location 171, he/she may provide an information update or delay tolerant content to users 45 and 7. By predicting movement of users in a group, we can schedule content dissemination of files with predictive deadlines.

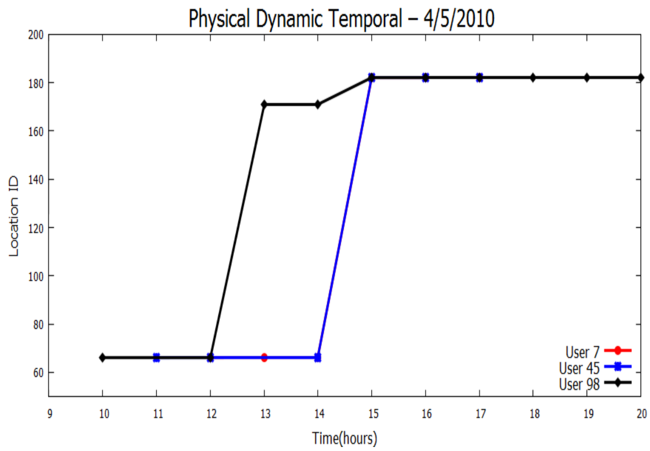

Figure 3.7: Physical Dynamic Temporal Behavior

We notice that movement of users 45 and 7 from location 66 to location 182 represents a possible social relationship. The users may be working on the same project and hence be present at location 182. Location 66 may be a classroom since all three users spend almost an hour at that location. Location 182 may be a computer lab or student lounge. If we are able to identify the location and time interval during which users meet, we may be able to infer the degree of familiarity between users.

3.3.4 Physical Dynamic Spatial Behavior

To observe physical dynamic spatial behavior, we identify and track different groups of users over time. We identify that path taken by members of different groups over time and locations. We attempt to identify "hotspots" - which are meeting points of different groups. In observing the above group behavior, we attempt to answer questions relating to the average amount of time that a group stays together and locations visited by the group during this time. By understanding group behavior, we are able to identify locations where maximum number of groups meet and hence result in efficient content dissemination. For example, a computer lab is a "hot spot" where students who are working on the same project meet to discuss class assignments. A number of groups meet at the same location periodically.

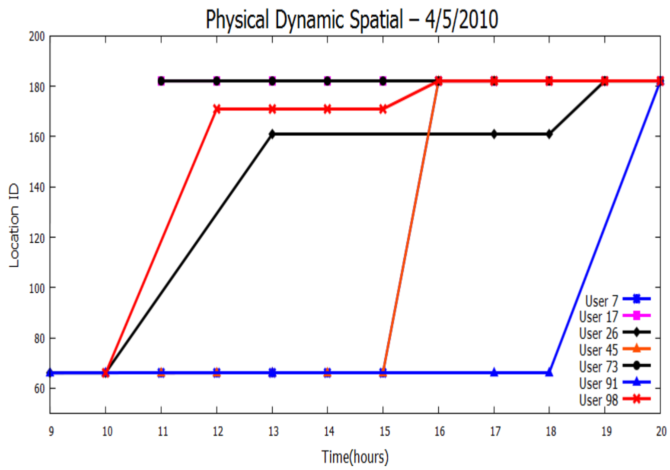

Figure 3.8: Physical Dynamic Spatial Behavior

Algorithm

To observe physical dynamic spatial behavior, we find the intersection of different occurrence sets O_{2_i} with O_{2_j} or O_{3_k} to identify multiple points of interest on a given day. The procedure is detailed as follows:

1. **Step 1:** We begin with the occurrence set O_{5_i} of the largest group CI_{5_i}. Each instance of the occurrence set O_{5_i} represents a location L_i, month m_i, day d_i and hour h_i where a group of 5 users are present. For each instance O_{5_i}, we note the month m_{5_i} and day d_{5_i}.

2. **Step 2:** For each occurrence set considered O_{5_i}, we find different subsets of CI_{5_i} that are present in different candidate sets CI_2, CI_3 etc. and look at the corresponding candidate occurrence sets. We represent the subgroups that occur on the given month m_{5_i} and day d_{5_i} as CI_{sub}.

3. **Step 3:** We also scan through the candidate item sets of sizes 2,3 and 4 that have users present in the larger group CI_{5_i}. The candidate item set found is added to CI_{sub} if it occurs on the given month m_{5_i} and day d_{5_i}. The candidate item set need not be a subset of CI_{5_i}

4. **Step 4:** Using the Bluetooth traces of each of the users present in the set CI_{sub}, we plot the movement of each user over the course of the day d_{5_i} in the month m_{5_i}.

Inference

In Figure 3.8, the candidate item set of 5 users are 17, 73, 26, 98 and 91. Users 7 and 45 are derived from this core group. We plot the movement of all 7 users on 4/5/2010. All users are present at location 182 between 8PM and 9PM. Users 26 and 91 are present together at location 66 between 9AM and 10AM. Users 91 and 7 stay at location 66 for an extended period of time from 10AM and 3PM. All users join users 17 and 73 at location 182 at different points during the day. From this model. we can develop a content dissemination protocol for different groups of users given the approximate time spent together at a given location. Since user 7 moves from one group to another at 3PM, we can use user 7 to transmit file chunks intended for users 98, 17 and 73. Since we know the locations visited by users, we can look into caching content at locations 66 and 182.

We can also estimate the social relationships between users given the extended periods of time that users spend together. Location 182 could be a computer lab or a student lounge. All users meet at this location at the end of the day which is often the behavior of students working on common group projects. Location 66 can be a classroom given that students are present in that location between 8AM and 3PM. We also notice that users 91 and 7 may share similar class schedules while users 17 and 73 may be close to a project deadline. This would explain their presence in location 182 for the entire day.

3.4 Conclusions

In this chapter we identify that the socio-physical model developed by social scientists is incomplete in the physical domain and the physical mobility model developed by computer scientists is incomplete in the social domain. We use data mining techniques on user mobility traces collected using the UIM tool. By analyzing the mobility traces we refine the SWC socio-physical model to answer questions about user mobility and group behavior.

In the next chapter we use the social contextual information obtained from the refined socio-physical SWC model and combine it with the content dissemination Sangam framework in order to construct a Dynamic Group Based Content Distribution Framework for mobile P2P networks. We present the design and partial implementation of the group based content dissemination framework on Android phones and evaluate the implementation of the Dynamic Group Based Content Distribution Framework using a model scenario.

CHAPTER 4

DYNAMIC GROUP BASED CONTENT DISTRIBUTION FRAMEWORK FOR UNSTRUCTURED P2P NETWORKS

In order to form a social P2P network and schedule content dissemination efficiently, we need to define different metrics based on which P2P networks will be formed. In this chapter we present the Dynamic Group Based Content Distribution Framework which uses social contextual information obtained from mobility traces (Chapter 3) for content dissemination among a group of users (Chapter 2).

4.1 Introduction

Peer-to-Peer (P2P) networks have gained immense popularity since the time of its inception [6] since users are able to share content in a distributed fashion without a central server and are able to download files from collocated peers and achieve bandwidth savings. All peers in a P2P network form an overlay network over the underlying network infrastructure. Each peer may have a single peer link or multiple peer links depending on the type of P2P network constructed. The two types of P2P networks - Structured and Unstructured have their own advantages and disadvantages. Structured P2P networks involve peers connected to one another over a structured topology such as the Chord P2P network [43]. The distributed hash table (DHT) overlay topology is efficient for large scale implementation and offers certain bounds on routing a query. However, the disadvantage is the maintaining the structure during churn. Unstructured P2P networks do not use any algorithm or topological rules for constructing a P2P network. The links between peers are formed in an ad hoc fashion. There are three types of unstructured P2P networks: 1) Pure P2P - no centralized control and all peers connect to one another in a distributed fashion, e.g., Gnutella, Freenet, 2) Centralized P2P - a central node is used for bootstrapping and indexing files, e.g., Napster, 3) Hybrid

P2P - bootstrapping nodes are deployed in the P2P network, e.g., Kazaa. The advantage of unstructured P2P networks is the construction of P2P networks in an ad hoc fashion and the disadvantage is the large number of flooding messages that may result due to it. In designing the Dynamic Group Based Content Distribution Framework we consider unstructured P2P networks since users are able to form groups in an ad hoc fashion without maintaining any structure.

The sudden infusion of mobile devices into the wireless ecosystem has resulted in a surge of new ideas and implementations of mobile P2P networks. As the number of mobile device users has increased, the demand for applications that can enable content distribution in a local and global fashion has also increased. P2P networks have been a popular method of sharing content among users without the efforts of a central scheduler. Napster [6], Gnutella [21] were some of the initial P2P file sharing services that gained immense popularity during the early 1990s. Users formed P2P networks based on their common interest in a file. With the advent of mobile devices, the number of different P2P services only increased.

Mobile devices users formed P2P networks with other users who were present in their vicinity based on their common interests. However P2P networks have a number of drawbacks such as fairness, churn, etc., which have carried over to the mobile P2P networks. Churn is characterized by new peers joining or leaving the P2P network, multiple failures occurring or any event that leads to a change in the membership of the P2P network. Churn in P2P networks results in an increased number of membership messages, missed download deadlines, inefficient scheduling of file segments, etc. The initial formation of groups is important in order to prevent premature leaving of nodes in a P2P network. Studying Churn and its effects is an important part of P2P network analysis as it is an indicator of the stability and resilience of the P2P network. In this chapter we attempt to counteract churn by using addition social contextual information about the user to form a Dynamic Group Based Content Distribution Framework for mobile P2P networks.

In order to counteract churn, age of nodes as a link metric, structured P2P networks etc have been proposed in order to improve the stability of P2P groups. In recent years, with the introduction of social networks and pervasive use of mobile devices, we are able to obtain a wealth of social

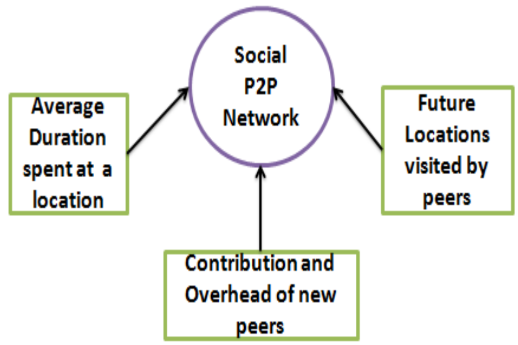

Figure 4.1: Possible Link Metrics for a Social P2P Network

and physical contextual information about the mobile user. In addition to finding out more about the people that a mobile user may be acquainted with, we are also able to pinpoint the current and future location of the user using applications such as four square, personal calendars, etc. Recent P2P networks [32], [29] have started to introduce these additional social contextual information when determining the P2P links and the corresponding weights on the edges. These P2P networks are called social P2P networks. The common preferences of one user with another increases the P2P link weight and hence the probability of link formation between the two users. The links can be based on common interests of the two users, common likes/dislikes about different subjects/products, etc. The weight of these links can be a counter or a weighted sum of the above mentioned possible link metrics.

In order to form a social P2P network and schedule content dissemination efficiently, we need to define the different metrics based on which the P2P networks will be formed. The Dynamic Group Based Content Distribution Framework works in a scenario where a large number of users are interested in the same content and the content has a relaxed download deadline. For example, in university campuses it may be beneficial to disseminate content about events held at the university to all students. This information needs to reach all students by the event time but it is not necessary for all users to download it in the shortest amount of time possible. The decision to form mobile P2P groups at a given location is based on the average duration spent by a user at a given location, future locations visited by the user, density of users at different locations and contribution and overhead of new users

(Figure 4.1). The Dynamic Group Based Content Distribution Framework attempts to use all three inputs to decide group membership at a given location and schedule file segments based on the Sangam framework (refer Chapter 2).

4.2 Related Work

In order to design the Dynamic Group Based Content Distribution Framework, the related work in this section is focused on the different link metrics that have been proposed to overcome churn in P2P networks. There are two social P2P networks - Socionet [32] and Prometheus [29] which both focus on constructing P2P links based on common user preferences and social network data. The two other P2P networks described in this section - Age-based protocol [33] and Churn Resistant Protocol (CRP) [31] - are based on the age of a node and network characteristics of the P2P network.

The Prometheus P2P network is a structured P2P network. Its underlying structure is a Distributed Hash Table (DHT). Socionet and Age-based P2P networks are unstructured P2P networks. The CRP is implemented on a hybrid P2P network, a blend of structured (base ring with a successor list) and unstructured (overlay links) network. Unstructured P2P networks are preferred when dealing with large churn scenarios since maintenance and repair of structured P2P networks introduces considerable overhead.

The CRP protocol [31] uses network latency and node degree to construct stable and churn resilient P2P networks. The metrics used to evaluate this system are average delivery delay, average control messages, average message replication ratio etc. However the authors do not evaluate query hit or other metrics which evaluate the efficiency of finding and retrieving data in a P2P network. The CRP protocol's aim is to disseminate data and it assumes all users are interested in this data. This might be the case for news broadcast information but for a user trying to find an audio file of a certain genre, flooding the entire P2P network may not be useful. The metric used for constructing and maintaining the link needs to be better defined in order to construct meaningful overlay networks.

The age-based overlay [33] network moves in that direction by proposing P2P links based on the age of the node. A simple monotonically decreasing function of age in the network is proposed to characterize a node's reputation. The authors assume that nodes will stay longer in the P2P network since an increase in reputation directly translates into more node connections. This may not always be true when users move from one location to another as is the case in mobile P2P networks. The Dynamic Group Based Content Distribution Framework accounts for the average duration of time spent by users in different locations using mobility traces of the user that have been collected in the past. The two metrics used to evaluate this protocol are average path length and clustering coefficient. Though the links are constructed using a metric other than network characteristics, it does not indicate the interest/preference of the user.

Socionet [32] constructs an overlay graph based on the user preferences mined from social networking data. Each link is weighted using a similarity metric. In Socionet, the authors use a music database to mine user preferences and construct the overlay network. The metrics used to evaluate the system now focus on query success ratio, recall rate, precision rate, number of successful matches during different churn scenarios, etc. The authors build and analyze P2P networks based on how efficiently a user is able to query data items given the additional input of a social graph.

Prometheus [29] builds on this idea and uses data from multiple sources such as Online Social Networks, collocation, viewing the same website etc. It uses a Distributed Hash Table based overlay and replicated storage for collecting and storing social sensor data. Since most of this data is sensitive, the authors introduce public key cryptography and access control lists to control the access of each user's information. The user decides the location and access of its online sensor data. The applications using this data create a geo-social graph with labeled interactions and their corresponding weights. The metric used to evaluate the system is average response time. Since some amount of time is spent querying user's social graph information from Prometheus, a certain delay is associated with the system. However, the user can query a range of information and use it to build a number of P2P applications.

Both Socionet and Prometheus compare their P2P overlay links based on social contextual information as opposed to random links. In both systems, socially aware overlays perform better than random overlay networks. Only Socionet is evaluated under churn and it performs better than a random network. Though the CRP and age-based overlays were not constructed randomly, and link weight calculation involved some metrics related to network latency, node degree, etc., user preferences were not considered. Hence socially aware overlays provide fast, efficient and relevant data to the user. Table 4.1 compares the four P2P systems based on the metrics used for determining link weight and evaluating the system. The Dynamic Group Based Content Distribution Framework builds on the idea of better performing socially aware P2P overlays by using mobility traces collected using the UIM tool to extract social contextual information about the users in the P2P network. Using the mobility traces, the Dynamic Group Based Content Distribution Framework is able to mine information about the average duration spent by a user at a given location, the possible future locations that will be traversed by the user, possible social acquaintances, density of users at different locations, etc.

Table 4.1: Comparison based on Link and Evaluation Metrics

Protocol	Link Weight Metrics	Evaluation Metrics
Churn Resistant Protocol	Proximity Weight = α* Capacity Proximity Weight(CPW)+(α - 1)* Network Proximity Weight(NPW)	Average delivery delay, Average control messages, Average message replication ratio
Age-based Protocol	age of the node	Average path length, Clustering coefficient
Socionet	User preference based similarity index	Query success ratio, Query recall rate, Query precision rate, Similar hit ratio, Number of successful query matches during churn
Prometheus	Social sensor data from OSNs, contact lists etc.	Average response time

4.3 Dynamic Group Based Content Distribution Framework

The Group Framework assumes that a large number of users are interested in a given content. It does not optimize file download time for any of the users in the P2P network since we assume a scenario of relaxed download deadlines. To determine the location of a user, the Dynamic Group Based Content Distribution Framework assumes that we are able to obtain the location information of a user with very fine granularity. For example, we can distinguish between two users on two different floors of a building. The scenario where the Dynamic Group Based Content Distribution Framework may be useful is in universities where a student may need to be informed of important guest speakers or events that will be held in the coming days. As long as the user receives it before the event, the content is considered to be relevant and hence the Dynamic Group Based Content Distribution Framework does not optimize file download time.

The approach taken by the Dynamic Group Based Content Distribution Framework to schedule the different files is to define time intervals during which segments of a given file of interest are disseminated among the different mobile users. This is similar to the Sangam framework seen in chapter 2. Each file of interest is divided into segments. A given number of file segments are disseminated to a P2P group in one *epoch* of time. An *epoch* of time is defined as the time interval that is required to schedule file segments among a P2P group and the subsequent exchange of these file segments among themselves.

The Dynamic Group Based Content Distribution Framework employs two different models in conjunction with one another. The Client-Server model is used to define the relationship between the central server and all the mobile users in the various groups. The P2P model defines the relationship between the different mobile devices of users who form groups based on the Dynamic Group Based Content Distribution Framework and exchange content. The central server is in charge of group membership and content scheduling while the P2P network is used to exchange content among the different mobile users who are part of the same group.

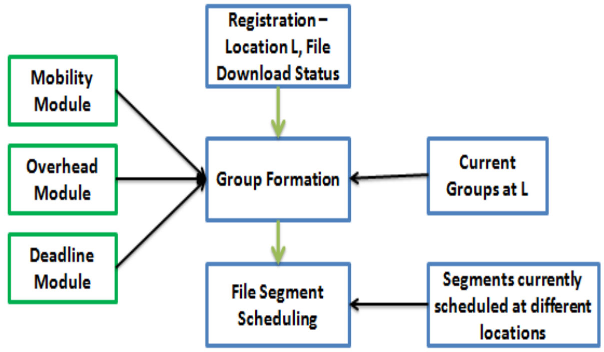

Figure 4.2: Group Decision Framework - Server

4.3.1 Server

The central server contains the logic for group membership management as well the segment scheduling. It obtains social and physical contextual information from the UIM mobility traces [48] and makes a decision regarding the P2P groups. The UIM mobility traces consist of Wi-Fi and Bluetooth traces. A Wi-Fi trace file is a set of Access Point MAC addresses scanned by the user at a given instant of time. Using clustering algorithms described in [48], each set of Access Point MACs are assigned a location. The Bluetooth traces are MAC addresses of the mobile users present in the vicinity of the user collecting the Bluetooth traces. The number of Bluetooth MACs scanned at a given instant of time gives us information about the density of users at a given location. Using these traces the server is able to calculate the average duration of time spent by users at a given location, the list of possible locations that a user may visit in the future, density of users at different locations in the present and future instants of time, etc. Figure 4.2 represents the block diagram of the server.

1. Group Membership Management: Since the server schedules a file segment every *epoch*, mobile users register with the server after the completion of every *epoch* or after moving to a new location. The server keeps track of all users scheduled and to be scheduled at different locations. The server runs the scheduler every Ts and uses three modules to form new groups:

(a) Mobility Module: The mobility module uses the UIM mobility traces to capture important aspects of the user - average duration spent by a user at a given location, density of users at different locations, future locations traversed by the user, etc. Based on the past behavior of the user at a given location and number of peers that a user may come in contact with in the future, the server assigns groups and schedules non-overlapping file segments accordingly. The average duration of time spent by a user at a given location is used to give an indication to the server about the current behavior of the user. The user may be moving through the given location in a transient fashion or may be staying for at least an *epoch* duration of time. By estimating the size of P2P groups and downloaded file segments at future locations, the server can schedule the right file segment at a given location.

(b) Overhead Module: It is often the case in P2P networks that certain peers request more content than they contribute to the P2P network. The overhead module calculates the membership message overhead and file segment contribution of a new peer before including a new peer as a group member. This helps the server prevent unnecessary overhead and bandwidth usage of certain peers.

(c) Deadline Module: Though the Dynamic Group Based Content Distribution Framework allows for relaxed deadlines when scheduling file segments, we still need to disseminate content to registered users in a reasonable time frame. The deadline module keeps track of this final deadline and ensures that the entire file is sent to the mobile user. Each deadline module also maintains a clock on the duration of time spent in a given location. After every epoch registration, the server is able to estimate the amount of time the user will remain at a given location. Using this information, the server will not schedule mobile users if they are at the end of their stay at a given location.

2. Segment Scheduling: Once the groups are formed based on the input of the above described decision modules, the server allocates the file segments to be sent to each P2P group. This function of the server also uses input from the mobility module. In order to schedule chunks for

each P2P group, the server finds the list of possible future locations of each peer and ensures that non-overlapping file segments are scheduled for peers in future P2P groups. This ensures the formation of large P2P groups with non-overlapping file segments and reduces usage of excessive server bandwidth.

4.3.2 P2P Network

The P2P network handles the initial peer registration and the subsequent file segment exchanges (refer Section 2.7 and 2.8). The server sends each mobile user a list of peers who are part of the same group. The peers exchange group registration messages and request file segments from the peers in the group. They also service file segment requests and enable segment exchanges among all peers in the P2P network.

4.4 Implementation

In order to evaluate the Dynamic Group Based Content Distribution Framework, a centralized scheduling server and a mobile P2P network has been implemented (refer Section 2.8). The centralized scheduling server was implemented using JAVA on a linux machine. Mobile clients were implemented as Android applications on Nexus S, Nexus One and Samsung Galaxy devices. Each file of interest F is divided into segments ($f_1,\ f_2,\ f_3,\ \ldots, f_m$) of fixed size. Each *epoch* e_i is responsible for the dissemination of multiple file segment $f_i,\ f_j,\ f_k\ \ldots$. Each epoch e_i begins with the registration of old and new mobile users with the central server. The central server then forms the P2P groups and schedules the file segments. The final step in the *epoch* is the mutual exchange of file segment in the P2P network. The four main steps in each *epoch* is explained in detail below.

4.4.1 Registration

Each mobile user u_i registers with the central server with its current location l_{u_i}, file status as a bit array b_{u_i}, duration of time already spent at this location d_i and current group id g_i. Upon registration of different users, both old and new, the server waits for a given time interval before beginning the group formation process. This is to ensure that it is able to group as many users as possible. The server begins the group formation and file segment scheduling process every Ts.

4.4.2 Group Formation

The server maintains a map for each user which lists the average duration spent by a user at a location (l_{u_i}, t_{av}). This duration is obtained after processing the mobility traces of the UIM tool. Every Ts the server initiates group formation. For the set of registered users that the server has at the beginning of the Ts interval, the server finds all users at a common location. If the average duration of all the users at a location is within the *epoch* interval and the peer overhead is within a given threshold, the users are assigned the same group. The final deadline of each peer is also taken into consideration. If the deadline of the file download is less than two epochs, the peer directly downloads the remaining file segments from the server.

4.4.3 File Segment Scheduling

The server processes the UIM tool mobility traces to also obtain the possible future locations traversed by a user. Using the mobility traces, we obtain a list of future locations that a user visits given the current location. Each future location is associated with a probability value based on the number of times that the user visits the location in the future after staying in the current location. Hence the server maintains a second map which lists the set of future locations and associated probability values given the current location l_{u_i} of the user $(l_{u_i}, l_{j1} = p_{j1}, l_{j2} = p_{j2})$. The server uses the list of future locations for each group to schedule non-overlapping file segments. This ensures that when peers meet at future locations, there are sufficient non-overlapping file segments to form P2P groups with low bandwidth and processing power overhead.

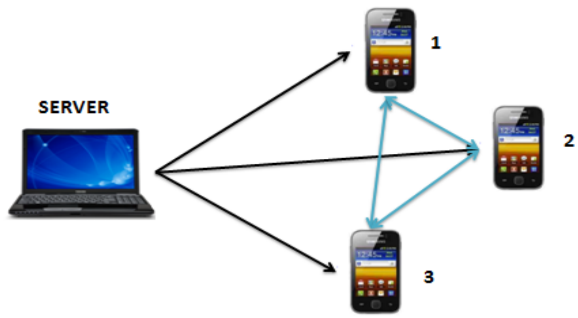

Figure 4.3: Experimental Setup

4.4.4 P2P File Segment Exchange

The P2P network is formed once the mobile users register with the server and receive their group assignment and file segments. The peers in the network request file segments from the other peers in the network. Each peer also sends requested file segments to different peers in the network. In the current implementation, each peer is allowed to request all required chunks from the P2P network in a round robin fashion. However if there are too many peers in the system, we can limit the number of file segments to be requested by each peer. This can prevent multiple requests to a single peer for any given file segment.

4.5 Evaluation

In order to evaluate the dynamic group based content distribution framework implemented using a central server and group of mobile devices, we consider a simple scenario of three mobile users moving from location 0 and location 1 to location 2.

4.5.1 Experimental Setup

The experimental setup (Figure 4.3) consists of a linux laptop running 10.04LTS as the server. Samsung Galaxy, Nexus S and Nexus One are the three mobile devices running Android OS 2.1 and higher. The Wi-Fi network is used to communicate with the server and among the mobile devices. For the P2P

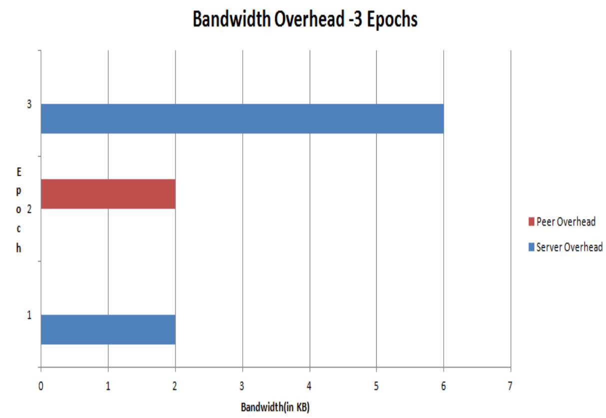

Figure 4.4: Bandwidth Overhead

network, the access point is used to connect the devices as opposed to any ad hoc connections. A file of size 120 kB with fixed segment sizes of 2 kB is used. Each *epoch* interval is set to be 12 s. The interval is set at 12s since the average time taken by the central server to form groups and schedule segments is about 6-8 s. We consider a scenario where three peers are interested in downloading a file of 6 segments. At *epoch* 1, peer 1 and peer 2 are present at location 1, Peer 3 is present at location 0. At *epoch* 2, peer 1, peer 2 and peer 3 meet at location 2. The group formation and segment scheduling is evaluated under two instances: 1) the server has no knowledge of the future locations and average duration spent by each peer at a given location, i.e., the Dynamic Group Based Content Distribution Framework is not used, 2) the server uses the Dynamic Group Based Content Distribution Framework to make decisions about group formation and segment scheduling. The bandwidth overhead and download time is evaluated using these two instances.

4.5.2 Results: Bandwidth Overhead

We compare the additional bandwidth that the server and peers utilize when users are grouped and scheduled without using the Dynamic Group Based Content Distribution Framework. In Figure 4.4, we notice the additional bandwidth used in each of the three epochs. In *epoch* 1, the server receives the registration request for users 1,2 and 3. Since users 1 and 2 are in the same location, the server schedules segment 0 and segment 1. User 3 is present at a different location. Since the server does not have any knowledge about the possible future locations of user 3, it schedules segment 0 again. This is represented by the 2kB bandwidth overhead seen in 4.4. In scenario 2, since the server knows that all three users will meet at location 2 in the future, non-overlapping segments are scheduled for all three users.

In *epoch* 2, there is an unequal exchange of file segments since user 3 has lesser number of file segments that user 1 and 2 are interested in. In Figure 4.4, the additional bandwidth that user 1 or user 2 will be subject to due to requests from user 3 is represented by the 2 kB overhead in epoch 2 seen in Figure 4.4.

Without the Dynamic Group Based Content Distribution Framework, some peers will be forced to send more file segments than they receive. This results in bandwidth and processing power overhead for a peer. It also results in the peer remaining in the P2P network for longer than required since it is forced to service all peers in the P2P network. In *epoch* 3, all peers request the remaining file segments from the server since overlapping segments are present in the P2P network and the download deadline has been set to 4 epochs. Each peer has one chunk that it is unable to obtain from other peers. This is directly downloaded from the server. The server sends the same chunk to all three peers. This additional bandwidth of 6 kB during epoch 3 is seen in Figure 4.4.

Without the Dynamic Group Based Content Distribution Framework, additional server and peer bandwidth is used as well as extended file download schedules. Considering the average duration of each user at a location helps in decreasing the rate of churn since peers with low remaining durations at a given location are not included in groups and scheduled accordingly.

4.5.3 Results: Download Time

In Figure 4.5 we see the difference in download times for a given peer when downloading a file using the Dynamic Group Based Content Distribution Framework and without it. When mobile peers use Dynamic Group Based Content Distribution Framework and register with the server, they are able to download a larger number of file segments into the P2P network in a shorter

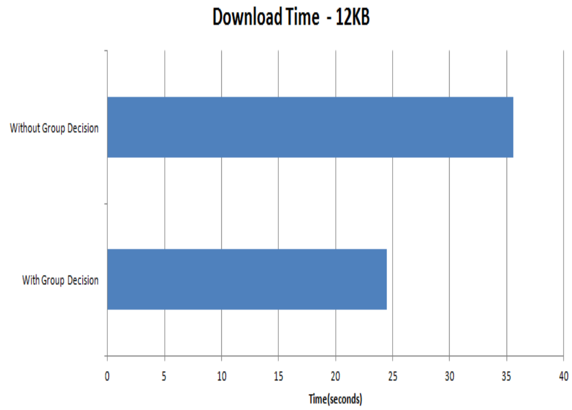

Figure 4.5: Download Time

amount of time since the server schedules non overlapping file segments and keeps track of the mutual file segment exchange. By using the Dynamic Group Based Content Distribution Framework, the server is able to schedule segment 0, segment 1 and segment 2 for peer 1, peer 2 and peer 3 in spite of peer 3 being in a separate location. The predictive knowledge of future locations that will be traversed by a peer ensure this assignment. The number of epochs needed by each user decreases since each peer is able to obtain a larger number of segments from the P2P network in each *epoch*. The difference in download time that we observe is due to the additional *epoch* required by each peer to request the remaining file segments from the server when this predictive knowledge is not used during file scheduling.

4.6 Conclusion

The Dynamic Group Based Content Distribution Framework uses the UIM mobility traces to provide social contextual information and forms P2P groups. Based on the P2P groups formed, content is scheduled and disseminated every epoch using the Sangam based content dissemination framework. Using input from the three decision modules - mobility, overhead and deadline module - the central server uses the average duration of time spent by a user at a given location, list of locations to be traversed by the user in the future, density of users in different locations, file download deadline, peer membership overhead, etc., to schedule peers accordingly.

Future work for the Dynamic Group Based Content Distribution Framework involves testing different scenarios over longer epochs and larger file sizes. The location detection also needs to be designed to provide the indoor location of the user with very fine granularity. The mobility traces can also be updated with new mobility information of each user as the Dynamic Group Based Content Distribution Framework receives mobile user registration data every *epoch*.

CHAPTER 5

CONCLUDING REMARKS

In this thesis we have developed protocols for content dissemination for static and mobile P2P groups. Using existing mobility traces collected by the UIM tool we have refined the existing socio-physical Situated Wireless Communities model and tried to answer questions relating to group mobility and formation.

Recognizing that most mobile devices are equipped with multiple interfaces, in the first part of this thesis, we developed the Sangam framework that would involve downloading a file over both the cellular and Wi-Fi links. In scenarios where there is no infrastructure and co-located peers are interested in fast download of large files, the Sangam framework minimizes the file download time. Using efficient chunk management policies based on resource information from the group of peers, the file chunks are assigned and distributed accordingly among the group of peers.

We implemented the Sangam framework on Android phones in order to compare and contrast the three chunk management policies: 1) equal chunk assignment 2) chunk assignment based on battery level and 3) chunk assignment based on CPU speed. We obtained a decrease in download time of 50 s - 60 s by implementing equal chunk assignment and CPU speed approach. Though the download time was slightly higher when we implemented chunk assignment based on battery level, we were able to conserve the power level of certain mobile devices ensuring the completion of the download process.

The Sangam framework assumed that the groups of mobile devices were present in one location and had multiple radio interfaces. However we know that users move from one location to another over the course of a day. In order to account for the mobility of groups of users, we needed to study different mobility patterns. In the second part of the thesis, we used the mobility traces obtained from the UIM tool to help us understand group mobility characteristics. We applied grouping algorithms to the UIM data set to discern physical temporal and physical spatial behavior. Based on this observation we refined the existing socio-physical Situated Wireless Communities model and attempted to infer fine-grained social and physical context of the users. The refinement of the socio-physical model is crucial for the future management of group mobility and development of mobile content dissemination protocols. Using this knowledge of group mobility, we attempted to design a framework for content dissemination in dynamic P2P groups.

In the third part of the thesis we used the social contextual information obtained from the refined socio-physical SWC model and combined it with the Sangam content dissemination framework for heterogeneous mobile groups in order to construct a dynamic group based content distribution framework for mobile P2P networks. The Dynamic Group Based Content Distribution Framework is a decision framework which is used to determine the formation of groups and schedule content for dissemination accordingly. For a given set of users who are interested in the same content, the decision to form groups is made globally by a centrally located server. The Dynamic Group Based Content Distribution Framework uses the UIM mobility traces to provide social contextual information and forms P2P groups accordingly. The refined socio-physical SWC model is used to provide information about the possible density of users at different locations, movement of users at different times during a given day, etc. The three decision modules - mobility, overhead and deadline - are used by the central server to determine the average duration of time spent by a user at a given location, list of locations to be traversed by the user in the future, density of users in different locations, file download deadline, peer membership overhead, etc., in order to form user groups and schedule file download among the users accordingly. On forming P2P groups based on the above social information, content is scheduled and disseminated using the Sangam framework. The evaluation of the Dynamic Group Based Content Distribution Framework shows increased bandwidth overhead and download time when the Dynamic Group Based Content Distribution Framework is not employed in forming the P2P groups.

5.1 Future Work

In this thesis we have developed and evaluated protocols required for content dissemination in static and dynamic P2P groups. Multiple radio interfaces and mobility traces have aided in developing protocols that can be used efficiently to disseminate content to static and mobile P2P groups respectively. However improvements can be made as part of our future work in the field of group mobility and content dissemination for P2P groups.

The Sangam protocol uses coarse CPU and power calculations. The CPU value sent to the server is the frequency of the CPU of the device. However it might be interesting to observe how scheduling is performed when the current download link capacity value is sent to the server. Accurate power measurements obtained by using more advanced Android API's might lead to a better understanding of the current resource levels of the device. During the development of the Sangam framework Android devices were not designed to communicate over the ad hoc interface. However recently all Android devices having operating systems 2.2 and above have their hardware so designed to enable the use of the WiFi Direct Android API to communicate with other mobile devices over their ad hoc interface. Using the ad hoc interface in conjunction with WiFi and cellular interfaces to evaluate the static P2P scenario will give us a good idea about the combined efficiency of multiple interfaces. Using the Sangam protocol to stream a media file instead of a single file download will also help us evaluate the download time efficiency of the protocol.

The UIM protocol helped us refine the socio-physical SWC model and facilitated the development of the content dissemination protocol for dynamic P2P groups. However one of the issues with large scale data collection is the storing of relevant and continuous data from devices. The process of location assignment can be made to relate to actual locations with the help of tags. In order to use different mobility traces to refine existing socio-physical models the analysis of mobility traces should be designed to be a generalized mobility data management tool. A GUI interface which helps the user specify the different formats will be useful in loading different mobility traces. Using an SQL database to aggregate and analyze different mobility traces to answer queries relating to user movements, density of users at different locations, etc., will help in creating a stronger mobility model. It might also help to assign a relevant probability value to identify the continuity and relevance of the mobility traces collected.

The Dynamic Group Based Content Distribution Framework uses the refined SWC model and content dissemination Sangam protocol to form P2P groups and disseminate content. To improve the Dynamic Group Based Content Distribution Framework it is necessary that all users are able to identify their current location and inform the central server. Hence all locations need to be identified using WiFi AP maps or any technology that provides very fine granularity of location detection. Since we are dealing with mobile P2P groups we can collect mobility traces of users and use it to fine-tune the Dynamic Group Based Content Distribution Framework. We can design the mobility model of the group based content distribution framework to act as a learning model with constant input of new mobility traces. The previously mentioned improvements of the Sangam framework and Mobility model combined with Dynamic Group Based Content Distribution Framework will be useful for designing and developing efficient content dissemination protocols for static and mobile P2P groups in the future.

REFERENCES

[1] Cdma and gsm network speed. http://www.tested.com/news/cdma-vs-gsm-examined-which-3g-network-is-superior/1630.

[2] Cellular speed test. http://www.bgr.com/2011/03/14/pc-world-t-mobiles-4g-network-wins-smartphone-speed-test/.

[3] Cisco newsroom. http://newsroom.cisco.com/press-release-content?type=webcontent.

[4] Connectivity manager android api. http://developer.android.com/reference/android/net/ConnectivityManager.html.

[5] Data transfer over cellular networks. http://www.digi.com/pdf/wp_efficientdatatransfer.pdf.

[6] Napster. http://www.napster.com/.

[7] Speedtest data. http://www.wired.com/gadgetlab/2011/02/iphone-speedtest/.

[8] Tanimoto coefficient. http://www.wikipedia.org/wiki/Jaccard_index.

[9] Wi-fi direct api. http://developer.android.com/resources/samples/WiFiDirectDemo/index.html.

[10] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499. Morgan Kaufmann Publishers Inc., 1994.

[11] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. Energy consumption in mobile phones: a measurement study and implications for network applications. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 280–293. ACM, 2009.

[12] R. Bhatia, L.E. Li, H. Luo, and R. Ramjee. Icam: Integrated cellular and ad hoc multicast. *IEEE Transactions on Mobile Computing*, pages 1004–1015, 2006.

[13] C. Boldrini, M. Conti, and A. Passarella. User-centric mobility models for opportunistic networking. In *Bio-Inspired Computing and Communication*, volume vol.5151 of *Lecture Notes in Computer Science*, pages 255–267. Springer Berlin / Heidelberg, 2008.

[14] Buttazzo and C. Giorgio. *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, volume 24. Springer, 2011.

[15] J. Claudino, N. Shay, and D. Valenzona. An evaluation of peer-to-peer file sharing on the android mobile platform. `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.176.83`.

[16] I. Constandache, X. Bao, M. Azizyan, and R.R. Choudhury. Did you see bob?: human localization using mobile phones. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 149–160, New York, NY, USA, 2010. ACM.

[17] S.S. Doria and M.A. Spohn. A multicast approach for peer-to-peer content distribution in mobile ad hoc networks. In *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, pages 1–6. IEEE, 2009.

[18] P. Ekler, J.K. Nurminen, and A. Kiss. Experiences of implementing bittorrent on java me platform. In *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, pages 1154–1158. IEEE.

[19] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. International conference on knowledge discovery and data mining, 1996.

[20] R. Gass and C. Diot. An experimental performance comparison of 3g and wi-fi. In *Passive and Active Measurement*, pages 71–80. Springer, 2010.

[21] P. K. Gummadi, S. Saroiu, and S.D. Gribble. A measurement study of napster and gnutella as examples of peer-to-peer file sharing systems. *SIGCOMM Comput. Commun. Rev.*, vol.32(no.1):pp.82–82, January 2002.

[22] H. Hanano, Y. Murata, N. Shibata, K. Yasumoto, and M. Ito. Video ads dissemination through wifi-cellular hybrid networks. In *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*, pages 1–6. IEEE, 2009.

[23] S.A. Hoseinitabatabaei, A. Gluhak, and R. Tafazolli. Udirect: A novel approach for pervasive observation of user direction with mobile phones. In *Pervasive Computing and Communications (PerCom), 2011 IEEE International Conference on*, pages 74–83. IEEE, 2011.

[24] H.Y. Hsieh and R. Sivakumar. On using peer-to-peer communication in cellular wireless data networks. *Mobile Computing, IEEE Transactions on*, vol.3(no.1):pp.57–72, 2004.

[25] K. Huguenin, A.M. Kermarrec, V. Rai, and M. Van Steen. Designing a tit-for-tat based peer-to-peer video-on-demand system. In *Proceedings of the 20th international workshop on Network and operating systems support for digital audio and video*, pages 93–98. ACM, 2010.

[26] P. Hui and J. Crowcroft. Human mobility models and opportunistic communications system design. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol.366(no.1872):pp.2005–2016, 2008.

[27] S.S. Kang and M. Mutka. Efficient mobile access to internet data via a wireless peer-to-peer network. In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on*, pages 197–205. IEEE, 2004.

[28] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. In *Proceedings of the 25th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, April 2006.

[29] N. Kourtellis, J. Finnis, P. Anderson, J. Blackburn, C. Borcea, and A. Iamnitchi. Prometheus: User-controlled p2p social data management for socially-aware applications. In Indranil Gupta and Cecilia Mascolo, editors, *Middleware 2010*, volume vol.6452, pages pp.212–231. Springer Berlin / Heidelberg, 2010.

[30] N.D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A.T. Campbell. A survey of mobile phone sensing. *Comm. Mag.*, vol.48(no.9):pp.140–150, September 2010.

[31] Z. Li, G. Xie, K. Hwang, and Z. Li. Churn-resilient protocol for massive data dissemination in p2p networks. *Parallel and Distributed Systems, IEEE Transactions on*, vol.22(no.8):pp.1342 –1349, aug. 2011.

[32] K.C.J. Lin, C. Wang, C. Chou, and L. Golubchik. Socionet: A social-based multimedia access system for unstructured p2p networks. *Parallel and Distributed Systems, IEEE Transactions on*, vol.21(no.7):pp.1027 –1041, July 2010.

[33] H. Liu, X. Liu, W. Song, and W. Wen. An age-based membership protocol against strong churn in unstructured p2p networks. In *Network Computing and Information Security (NCIS), 2011 International Conference on*, volume vol.2, pages pp.195 –200, May 2011.

[34] H. Lu, W. Pan, N.D. Lane, T. Choudhury, and A.T. Campbell. Soundsense: scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 165–178, New York, NY, USA, 2009. ACM.

[35] H. Lu, J. Yang, Z. Liu, N.D. Lane, T. Choudhury, and A.T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 71–84. ACM, 2010.

[36] S. Lu, S. Shere, Y. Liu, and Y. Liu. Device discovery and connection establishment approach using ad-hoc wi-fi for opportunistic networks. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 461 –466, march 2011.

[37] H. Luo, X. Meng, R. Ramjee, P. Sinha, and L.E. Li. The design and evaluation of unified cellular and ad hoc networks. *IEEE Transactions on Mobile Computing*, pages 1060–1074, 2007.

[38] T.K. Madsen, Q. Zhang, F. Fitzek, and M. Katz. Exploiting cooperation for performance enhancement and high data rates. *Journal of Communications*, vol.4(no.3):pp.193–202, 2009.

[39] Y. Tong Man and Y. Qi. Mobile wireless communications device having dual antenna system for cellular and wifi, May 2008. US Patent 7,369,091.

[40] E. Miluzzo, N.D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S.B. Eisenman, X. Zheng, and A.T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, SenSys '08, pages 337–350, New York, NY, USA, 2008. ACM.

[41] G.P. Perrucci, F.H.P. Fitzek, G. Sasso, W. Kellerer, and J. Widmer. On the impact of 2g and 3g network usage for mobile phones' battery life. In *Wireless Conference, 2009. EW 2009. European*, pages 255–259. IEEE, 2009.

[42] I. Rimac, S. Borst, and A. Walid. Peer-assisted content distribution networks: performance gains and server capacity savings. *Bell Labs Technical Journal*, vol.13(no.3):pp.59–69, 2008.

[43] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *ACM SIGCOMM Computer Communication Review*, volume vol.31, pages pp.149–160. ACM, 2001.

[44] J. Sun and M.S. Poole. Beyond connection: situated wireless communities. *Commun. ACM*, vol.53:pp.121–125, June 2010.

[45] A. Szabo, T. Weiherer, and J. Bamberger. Unsupervised learning of propagation time for indoor localization. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, may 2011.

[46] P.K. Tysowski, P. Zhao, and K. Naik. Peer to peer content sharing on ad hoc networks of smartphones. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pages 1445–1450. IEEE.

[47] J.D. Ullman. Np-complete scheduling problems. *Journal of Computer and System Sciences*, vol.10(no.3):pp.384–393, 1975.

[48] L. Vu, Q. Do, and K. Nahrstedt. Jyotish: A novel framework for constructing predictive model of people movement from joint wifi/bluetooth trace. In *Pervasive Computing and Communications (PerCom), 2011 IEEE International Conference on*, pages 54–62. IEEE, 2011.

[49] 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 the 13th ACM international conference on Modeling, analysis, and simulation of wireless and mobile systems*, MSWIM '10, pages 257–265. ACM, 2010.

[50] L. Vu, K. Nahrstedt, I. Rimac, V. Hilt, and M. Hofmann. Ishare: exploiting opportunistic ad hoc connections for improving data download of cellular users. In *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*, pages 1475–1480. IEEE, 2010.

[51] C. Walsh, A. Doci, and T. Camp. A call to arms: it's time for real mobility models. *SIGMOBILE Mob. Comput. Commun. Rev.*, vol.12:pp.34–36, January 2008.

[52] J. Whitbeck, M. Amorim, Y. Lopez, J. Leguay, and V. Conan. Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–10. IEEE, 2011.