Time Critical Content Delivery using Predictable Patterns in Mobile Social Networks

Fawad Nazir*‡, Jianhua Ma†, Aruna Seneviratne*‡
*Electrical Engineering Department, University of New South Wales (UNSW), Sydney, Australia
†Faculty of Computer and Information Sciences, Hosei University, Tokyo, Japan
‡Networked Systems Research Group, National ICT Australia (NICTA), Sydney, Australia
fawad.nazir@student.unsw.edu.au, jianhua@hosei.ac.jp, aruna.seneviratne@nicta.com.au

***Abstract*—In Mobile Social Networks (MSN) individuals with similar interests or commonalities connect to each other using the mobile phones. MSN are special kind of Ad-hoc Networks in which wireless connectivity (i.e. encounters and re-encounters) with social peers is predictable. Most of the recent research proposes routing framework to exploit these predictable patterns to identify the best information carriers. The best information carriers are selected based on the high probability of encounter with potential information recipients. In these approaches an important variable is ignored i.e. time of encounter. Considering encounter time in the protocol gives time assurance of message delivery for time critical application. Therefore in this paper we address the research question, how to exploit people's predictable social patterns to improve the content delivery performance and lower end-to-end delay in time critical applications? Our assumption is that the people follow similar mobility patterns daily (i.e. Monday to Friday). In this paper we verify our research question and assumption using real trace data of 100 users carrying Nokia 6600 smart phones over the course of nine months. Furthermore we model, analyze and propose algorithms for social encounter based content delivery system for time critical applications. The simple heuristics and initial study presented in this paper achieve a timely content delivery using predictable patterns while lowering system wide traffic flooding.**

I. Introduction

At the end of year 2007 there were 3.3 billions mobile users as half of the world has a mobile phone [1]. This shows that mobility is becoming an integral part of human lives. Moreover, mobility is not just impacting communication services as the primary service of the historical Internet but its current and future services, especially information/data/content sharing [2]. This opens up a new and challenging area of research i.e. "Mobile Content Delivery". The advancement and popularity in social networks is raising new research questions e.g. how will Internet look like if it was designed around social networks? The study of social networks can be broadly defined as computational facilitation of social studies and human social dynamics as well as design and use of information and communication technologies that consider social context.

Social networks have recently become one of the central themes across a number of information and communication technology fields and attracted significant interest from not only researchers in computing and social sciences, but also software and online game vendors, web entrepreneurs, political analysts, digital government practitioners. Interlinking the usage of mobile phones and social networks, gives birth to Mobile Social Networks (MSN). In MSN individuals with similar interests or commonalities, connect to each other using the mobile phones. Much like web based social networking, mobile social networking occurs in virtual communities. A current trend for Internet social networking web-sites such as MySpace[3] and Facebook[4] is turning mobile. There are two basic types of MSN's. The first is an extension of web-based social networks on mobile devices and they distribute their communities via the default start pages on mobile phone browsers, an example is JuiceCaster[5]. The second type is pure MSN, that are build purely on ad-hoc bases based on the social encounters and re-encounters [1]. In this paper we concentrate on the second type of social network and use predictable social encounters for time critical content delivery. Our assumption is that the people have repeated mobility patterns and user's mobile device can predict these patterns over time using the history data. The current research in MSN is also using this information to develop routing framework by identifying the best information carriers. These carriers are selected based on the high probability of encounter with potential information recipients. However, an important variable is ignored thats is exact "time of encounter". Considering encounter time in the protocol gives time assurance of message delivery for time critical application. Therefore in this paper we address the research question, how to exploit people's predictable social patterns to improve the content delivery performance and lower end-to-end delay in time critical applications?

The contribution of this paper can be summarized as follows:

- We verify our research question and assumption using real trace data of 100 users carrying Nokia 6600 smart phones over the course of nine months[2].
- We propose a technique to create a backbone MSN for content delivery
- We developed algorithms for interest exchange, next hop

[1] http://www.zdnet.com.au/news/communications/soa/3-3-billion-mobile-users-as-half-the-world-gets-a-phone/0,130061791,339289294,00.htm
[2] http://www.ipoque.com/news_&_events/internet_studies/internet_study_2007
[3] www.myspace.com
[4] www.facebook.com
[5] www.juicecaster.com

selection and message delivery in MSN
- Verification of our algorithms using MSN simulation environment [3] and discussion of results

The rest of the paper is organized as follows. The next section reviews related work. In Section 3, we discuss the system model of mobile social content delivery in detail. Section 4 covers user mobility behavior study. In Section 5, we discuss the formation of backbone MSN. Algorithms for time critical content delivery in MSNs are discussed in Section 6. In Section 7, we apply these algorithms on backbone MSN. Section 7 shows the simulation results, and finally we conclude the paper.

II. Related Work

Mobile content delivery is a well studied area of research. As a result, there exist today a multitude of solutions aimed at managing this problem. These solutions are at varying stages of deployment, from purely analytical research, to experimentally validated proposals, right through to fully standardized and commercially available systems. The solutions from mobile networking community address this issue through mobility management [4] [5]. These protocols focus on keeping the connection active during the user mobility. The solutions from Mobile Ad hoc Network (MANET) community are mainly focused on packet routing in the fully connected graph such as AODV[6], DSR[7] and DSDV[8]. These protocols take an assumption that network graph is fully connected and fail to route messages if there is no complete route from source to destination at the time of sending a message. In all of the above-mentioned content delivery approaches an important characteristic of mobile systems is ignored, that is, mobile device social encounters/re-encounters. To exploit this characteristic there are set of algorithms that focus on forwarding messages to encountered nodes unless it reaches its destination [9] [10] [11]. These protocols use device encounters for routing, we call these encounters as blind encounters. The reason is that these encounters dont take into account essence behind these encounters. If we research and study the reason behind these encounters then we can predict these encounters and make more robust and efficient forwarding decisions. Furthermore, we can give message delivery assurance to the user. Therefore in this paper we ask a research question i.e. how to exploit people's predictable social patterns to improve the content delivery performance and lower end-to-end delay in time critical applications?. A fair amount of study has gone into social network analysis [12] [13] [14]. These research articles study the structure of social networks and conclude interesting observations about social network in general. Another related area of research is, using social network analysis for routing in delay-tolerant network [15] [16] [17] [18]. These algorithms use social network analysis techniques to enhance routing using encounters based social networks. These algorithms utilize information like node encounter, inter encounter times, encounter duration to develop delay-tolerant routing algorithms. In these algorithms the assumption is that, if two nodes encounter each other then they have some sort of

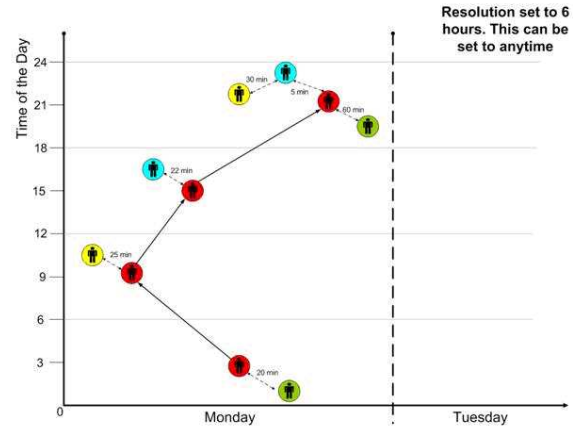

Fig. 1. Social encounter/re-encounter patterns of the red user on every Monday.

social connection with each other. In our proposed approach we argue that this is not a correct assumption as you may encounter a large number of people with whom you have no social connection. Therefore, in our proposed approach we propose the two user can have a social group if their encounter patterns are consistent and repeatable. In this paper we analyze time of the day, hourly, daily and weekly data of people social encounters to form social groups. After forming a social network using the above-mentioned method, then we optimally deliver the time critical content. Furthermore, we have simulated our algorithms on a participatory mobile social network simulation environment[3]. As the connectivity graph in our approach is based on the underlying mobility model therefore we used patterns based mobility model which is close to real life working day movement models[3].

III. Model of Social Content Delivery System

Internet supports many different types of information sharing systems e.g. web, peer-to-peer systems, publish subscribe models, social network websites etc. Recently social networking websites have gained popularity and are among the most visited sites on the web. However from an abstract point of view the web (www) is largely organized around content and social networks are organized around users. Users join social networks, publish their profile, content and create links to other users they know. These social networks are used to maintain social relationships, finding users with similar interests and locating related content. The popularity of mobile phones gives birth to MSN and major research focus is on content delivery in MSN [19]. The study of MSN require in-depth understanding of the dynamic graph structure. Moreover, to understand the probabilistic model of social encounters a user has on his/her day-to-day life.

In this section we will discuss the concept of improving content delivery using these social encounters/re-encounters and present our system architecture. Furthermore we will also model our problem mathematically.

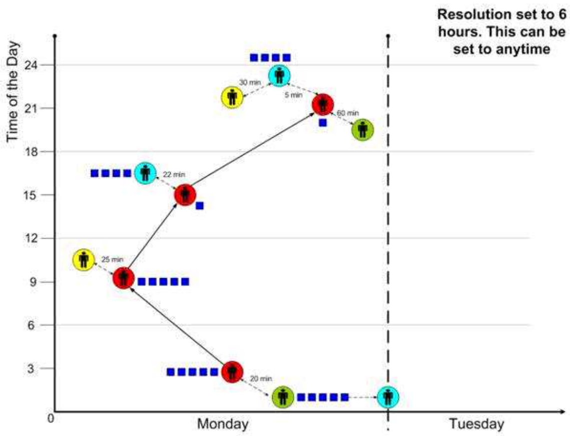

Fig. 2. Data Transfer from green-user to blue-user using social encounters of the red-user.

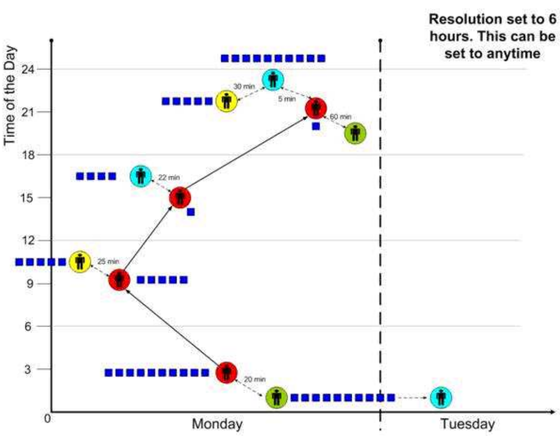

Fig. 3. Scenario for Accumulated multi-hop encounter duration

A. Proposed Model Overview

This paper exploit two defining characteristics of mobile system i.e. mobility and social encounters. This can be achieved after forming reliable social groups/communities of mobile users. Reliable social groups can be established by studying and deducing consistent/repeatable patterns. In a consistent/repeatable mobile social group, user might not be connected all the time. However, they have a high probability of encounters/re-encounters for a given average of inter-encounter time (i.e. time period between encounters) thus enabling the delivery of content in a more predictable manner.

This concept is illustrated in the scenario shown in Figure 1, 2 and 3. These figures highlight the mobility patterns and inter-encounter times of the red user with other social actors. In this scenario we assume that over the period of time we can predict the mobility and inter-encounter time pattern of the red user on every Monday for multiple weeks. The graphs in Figure 1, 2 and 3 shows time of the day on $y-axis$ and days of the week on $x-axis$. The time intervals of these graphs is set to 6 hours. Figure 1, shows that every Monday from time 0-6 hrs the red user encounters green user for 20 min, from 6-12 hrs he encounters the yellow user for 25 min, from 12-18 hrs he encounters the blue user for 22 min, from 18-24 hrs he encounters the green user for 60 min and the blue user for 5 min. We also have the information that the blue user from 18-24 hrs encounters the yellow user for 30 min.

Given this information if the green user (Figure 2) in time 0-6 hrs wants to send a file of 25 MB to the blue user, then what will be the path that will deliver the data with the highest probability and with time assurance having minimum usage of resources? Assume that 1 MB takes 1 min to be transferred. As the green user encounters the red user in time 0-6 hrs. Therefore, the green user will have the information that red user will interact with the blue user later in the day for 27 min. Having known this green user will give the data to red user in time 0-6 hrs.

In 12-18 hrs the red user will meet the blue user for 22 min. During this time he will transfer 20 MB of data and the rest of the 5 MB data will be transferred to the blue user when red user meets him during 18-24 hrs for 5 min.

In Figure 3 we extend our scenario to discuss a more complicated case. In Figure 3 the green user wants to send the data of 50 MB. To transfer 50 MB data from green user to the blue user the red user should encounter the blue user for at least 50 min. However the red user only encounters the blue user for 27 min. In this case we will calculate its accumulated multi-hop encounter duration. This means that red user will have the information that he will meet the yellow user from 6-12 hr for 25 min, who will then meet the blue user from 18-21 hr for 30 min. Therefore the red user will deliver 25 MB of data to yellow user during 6-12 hr. This data will then be given to blue user when the yellow user meet blue user for 30 min from 18-24 hr. With the help of the above scenarios, we have demonstrated how we can take advantage of social encounters/re-encounters for content delivery in MSN.

B. Problem Definition and Notations

To model the above scenario, let $G = (V, E)$ be a finite connected undirected graph (which represents a consistent/repeatable social network), where vertex V represents the set of user devices in a MSN and E represents a probabilistic link having high probability of encounter/re-encounter. If $x, y \in V$ then $l(x, y)$ is a boolean function representing the high probability of encounter/re-encounter of x and y with a poison random delay X of inter-encounters times.

$$P(X = x) = \frac{\lambda^x}{x!} e^{-\lambda}, x = 0, 1, 2, ... \text{ where } \lambda > 0$$

$$l(x, y) = \begin{cases} 1, & \text{x and y high probability of encounter/re-encounter i.e. } t_s \\ 0, & \text{Otherwise} \end{cases}$$

Without loss of granularity, we assume that a user device in a social network acts as a transceiver, producing and consuming messages, as a router, forwarding messages on behalf of other devices, and as a buffer, storing messages for destinations with high probability of encounters. The user devices are also given numeric identifiers, so hereafter we let

$V_n = \mid V \mid, V = 1, 2, 3, ..., V_n$, here V_n is total number of user devices.

We define neighbors in our system as if x_i and y_i are neighbours then $l(x, y) = 1$ and we denote neighbors relationships as $x_i \longleftrightarrow y_i$.

$$\forall x_i, y_i \in V, \text{ if } l(x, y) = 1 \text{ then } x_i \longleftrightarrow y_i$$

We also define the number of neighbors of user device in a social netwrk as N_{v_i}. Note that there can be as many as $V(V-1)$ neighbors and as less as 0.

$$N_{v_i} = \sum_{x \in V, x \neq y} l(x, y)$$

These user devices also have a neighbor table $N_{v_i}, v_i \in V$ of frequently encountered user devices. This information can help other users to use these devices as a bridge between devices in their neighbor table. Using these devices as a bridge will provide content delivery assurance with a predictable delay. The structure of neighbor table is given in $(eq.7)$. $N_{v_i}(x_i)$ represent the entry for v_i in the neighbor table of x_i. Rest of the fields in the equation below are explained in $Table - 1$.

Table -1 : Notation Description	
$l(x, y)$	link between x and y
V_n	total number of user devices
$x_i \leftrightarrow y_i$	x_i and y_i are neighbors
$N(v_i)$	number of neighbors
$N_{v_i}(x_i)$	neighbor table of device v_i
$T_E(v_i, x_i)$	time left for v_i to encounter x_i
$T_D(v_i, x_i)$	upcoming encounter duration of v_i with x_i
t_s	encounter threshold to form a social group
D_i	i^{th} day of the we where $i = 1, 2, 3, 4, 5$
$T_{i \to j}$	time duration from ihrs to jhrs
Q_{v_i}	query based profile of device v_i
m	message format $(m.header, m, m.ttl, m.timeuout, m.size)$
$E_n(x_i \leftrightarrow y_i)$	number of encounters between x_i and y_i
C_{x_i}	amount of content in the buffer of device x_i

We explained the MSN structure and related functions. Now we will talk about notations and functions associated with mobile content delivery.

Every user device in a social network will have its own profile. This profile is a set of keywords and is represented by Q_{v_i}. This profile represents the interest of the user using device v_i.

v_i receives m, if $Q_{v_i} \Rightarrow m$, where $Q_{v_i} \Rightarrow m$ means Q_{v_i} matches the keywords in m.

Therefore:

$$Q_{v_i}(m) = \begin{cases} 1, & \text{if } Q_{v_i} \Rightarrow m \\ 0, & \text{Otherwise} \end{cases}$$

The message m has a format of $(m.header, m, m.ttl, m.timeuout, m.size)$. Here, $m.header$ is set of keyword(s) representing the content of the message, m is the actual message, $m.ttl$ is the number of hops the message can penetrate in the MSN, $m.timeout$ is the expiry time of the message in seconds and $m.size$ is the size of the message in bytes.

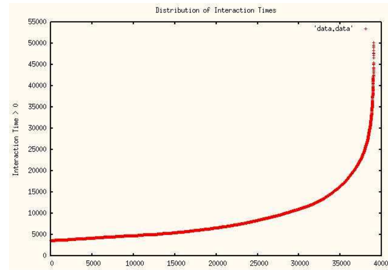

Fig. 4. Distribution of encounter duration.

IV. User Mobility Behavior Study

This section describes the analysis of the statistical properties of traces from Reality Mining project at MIT Media Lab [2]. The experiment in this project used 100 Nokia 6600 smart phones with pre-installed software especially developed for this experiment. Seventy-five of these devices were distributed to student or faculty members in the MIT Media Laboratory, and the other 25 were given to students at the MIT Sloan business school, close to the laboratory. The data generated by the software included call logs, cell tower IDs, Bluetooth devices in range, phone status, and application usage. This experiment ran for almost nine months during the course of $2004 - 2005$ academic years. Our analysis of this data set is mainly focused on user encounter patterns. The user encounters patterns are studied according to the days of the weeks and times of the day. The two important parameters studied for each pair of users in our data set are: average encounter duration and probability of encounter (eq.6). A detailed analysis was done on the data for Monday's, Tuesday's and Wednesday's between $2004 - 09 - 01$ and $2004 - 11 - 30$. The users selected for the study have highest encountered time with each other as opposed to the highest encounters, as highest encounter number does not fulfill the purpose because the encounter time can be as low as approximately 0 seconds. Figure 4, shows a distribution graph of the users and their encounter times. This figure shows an exponential graph from which we selected the users that have $encounter - duration > 4000 seconds$. The encounter-duration between two users gives us an important measure for our study that is average encounter duration. $L_w(i, j) = \{E_{avg}, P_e\}$ $(eq.6)$. The second parameter studied was the probability of encounters and re-encounters. The graph in Figure 5 shows the encounters and re-encounters between a pair of users for six months having time of the day on $y-axis$. From this encounter data, we can calculate the probability of interactions between two users for a particular day of the week and the time of the day. This data was used in combination with the encounter duration for our proposed time critical content delivery algorithms.

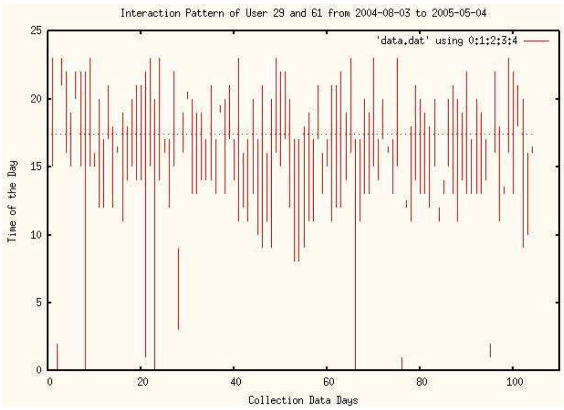

Fig. 5. Encounter Patterns on the Time-Series of two Users from the Reality Mining Data Set.

V. FORMATION OF A BACKBONE MSN

In this paper we propose to identify a backbone MSN comprising of the most active social users. The active users are selected based on the information in the neighbor tables and the depth of profile. Neighbors table contain the information about daily encounters/re-encounters and calculate the probability of the next projected encounter at a particular time of the day. User profile defines the type of message a user device can receive or store to forward. The depth of profile means generality of the user profile. The more devices the user encounter the more generic its receiving/forwarding profile will be. The structure of neighbor table is given in (eq.7). $N_v(x_i)$ represents the entry for node v_i in the neighbor table of node x_i. Here $T_E(v,x)$ means time left for an upcoming encounter with device x_i and $T_D(v,x)$ means upcoming encounter duration. As total number of the encountered devices are given by V_n. Therefore, the encounter nodes with higher V_n and deeper Q_v becomes the backbone nodes.

$N_{v_i} = (x_i, T_E(v_i, x_i), T_D(v_i, x_i))\ (eq.7)$

Figure 6-(a) shows the social encounter/re-encounter graph of users. Here the encounter links between users have been plotted. These link represent the high probability of encounter/re-encounter. This clearly shows 7 users are more connected as compared to other users. In order to identify the major social actors and backbone MSN these social links will be filtered using two important parameters discussed in (eq.6) i.e. average encounter duration and probability of interaction.

Figure 6-(b) shows the filtered social graph with only the backbone MSN and gateway users (i.e. directly connected users) to the backbone. In this network the nodes in blue boxes represent the backbone nodes and the thicker links show the core links between the backbone nodes. The rest two nodes in white circles 19 and 24 are the nodes that are well connected to the backbone network and can utilize it. Section 7 illustrated how these two nodes can utilize this backbone social network for content delivery. It is explained using a scenario in which node 19 sends content to node 24 on Monday. This time critical content delivery is analyzed in detail for different times of the day using real trace data.

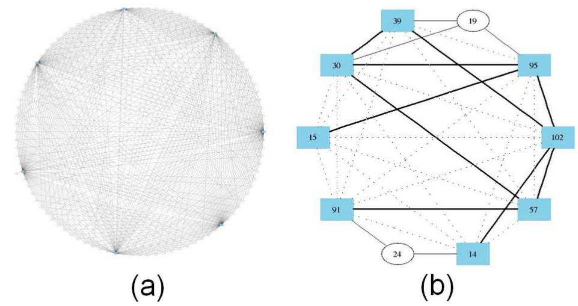

Fig. 6. Backbone MSN (a) All Social Links to Identify Backbone MSN. (b)Filtered Social Links

VI. CONTENT DELIVERY ALGORITHMS

For the purpose of simplicity we assume that all the user devices have similar technical features. These devices can exchange data when they are in each other's range using blue-tooth protocol. In the proposed protocol the key idea/assumption is that the users with common interest encounter/re-encounter each other often and these encounters are predictable. The location and handling of these devices depend totally on the behavior of the user carrying that device.

Message delivery is driven by the content of the message. Each node subscribes to interests e.g. "Events in Sydney", "Cricket Match Results", "John's Concert tickets", Rock music etc. The goal of the protocol is to deliver the message that matches the subscribed keyword(s). There are three algorithms in our proposed approach i.e. interest exchange, next hop selection and message delivery. Furthermore, the algorithms are based on predictable encounters/re-encounters. Unlike conventional mobile publish-subscribe models, in our approach mobile phones only exchange profile when they have a high re-encounter probability i.e. higher value of t_s (Table-1). Each mobile device keep record of the devices it encounters and discard the information if the re-encounter does not happen within a timeout.

A. Interest Exchange

If the re-encounter counter between two devices is greater or equal to t_s, then these devices are considered to have a social connection and they can exchange profiles for future content exchange. The algorithm for interest exchange is explained in Algorithm 1.

B. Next Hop Selection

The backbone social network is used to route the packets. This is discussed in detail in section 4. In this section we will more formally give the algorithm for next hop selection. Basically, in the next hop selection algorithm we determine whether the neighbors of the current device are better carrier of the message than the current device.

C. Message Delivery

A device passes a message to the neighboring device if any one of the following case is true:

Algorithm 1: Interest Exchange Algorithm

Interest Exchange of x_i with its social groups
Neighbours of x_i = $N(x_i)$
for $v_i, v_i \in V_n$ **do**
  if $E_n(x_i \longleftrightarrow v_i) \geq t_s$ **then**
    $N(x_i) \leftarrow N(x_i) \bigcup v_i$

Interest Profile of x_i is Q_{x_i}
1. Aggregate the profile of all neighbours $N(x_i)$ in the neighbour table.
for $n_i, n_i \in N_{x_i}$ **do**
  send Q_{x_i} to n_i

Method called on receipt of a profile exchange message from neighbouring nodes. Receive message m from v_i
for $n_i, n_i \in N_{x_i}$ **do**
  if $n_i = v_i \wedge E_n(n_i \leftrightarrow v_i) \geq t_s$ **then**
    $Q_{x_i} \leftarrow Q_{x_i} \bigcup Q_{v_i}$
  else if $n_i = v_i \wedge E_n(n_i \leftrightarrow v_i) < t_s$ **then**
    $E_n(n_i \leftrightarrow v_i) \leftarrow E_n(n_i \leftrightarrow v_i) + 1$

Algorithm 2: Next Hop Selection Algorithm

Current carrier of the message m is x_i. The format of message is defined in Table-1. Now we will check whether the neighbors of x_i are better carrier of message m or not.
for $n_i, n_i \in N_{x_i}$ **do**
  if $m \subset Q_{n_i} \wedge T_e(x_i, n_i) \leq m.timeout \wedge isTransferable(T_D(x_i, n_i), m.size)$ **then**
    Choose n_i as a mext hop for m.

Here:

- $m \subset Q_{n_i}$ means if message is a subset of the current neighbours profile.
- $T_E(x_i, n_i) \leq m.timeout$ means if time left for n_i to encounter the destination is less than or equal to the m.timeout.
- $isTransferable(T_D(x_i, n_i), m.size)$ means can the profile be transfered in the duration of encounter.

- The message matched the interest of the neighboring device(s).
- Neighboring device(s) are better carrier of the message than the current device

The second case is already discussed in 6(B). This section presents an algorithm for the first case and methods that are invoked when a device receives a message from its neighboring nodes.

VII. Content Delivery Using Backbone MSN

This section illustrates the methodology of content delivery using the backbone social networks (Figure 6-(b)) applying the algorithms discussed in the previous section. We will limit our discussion to all Monday's between $2004-09-01$ and $2004-11-30$. Same methodology can be applied to other days of the week. The backbone social network in Figure 6-(b) is represented in a more readable form in Figure 7. In this figure each link is labeled with a pair of values discussed in $(eq.6)$. The weights are calculated using the accumulated data for all Monday's between $2004-09-01$ and $2004-11-30$. The average time of encounter is given in seconds. The average time of encounter is proportional to the content size. If the content can be exchanged in the time of encounter then we will check the probability of encounter. This is important because the time of encounter should be long enough to exchange the content, even if the probability of encounter/re-encounter is high still if the duration of encounter is low then that link can not be used for content delivery.

Algorithm 3: Message Delivery Algorithm

Algorithm to check if the message m matched the interest of the neighboring nodes N_{x_i} of the current node x_i.
for $n_i, n_i \in N_{x_i}$ **do**
  for $c_i, c_i \in C_{x_i}$ **do** `/* checking all the data in the buffer of` x_i `*/`
    if $Q_{n_i} \subset c_i$ **then**
      send c_i to x_i

Methods invoked by x_i on receipt of a data message m. The following are the checks that are made when a device received a data message

- Do I already have this data?
- Do i need this data?
- Does anyone else whom i will encounter need this data?

Case : 1
for $c_i, c_i \in C_{x_i}$ **do** `/* checking all the data in the buffer of` x_i `*/`
  if $m.header \subset c_i$ **then**
    Discard this message.

Case : 2
if $Q_{x_i} \subset c_i$ **then**
  send c_i to the application.

Case : 3
for $n_i, n_i \in N_{x_i}$ **do**
  if $Q_{n_i} \subset m.header$ **then**
    Keep m in the message buffer.

The scenario shows in Figure 6-(b) is that node 19 has to send content to node 24 on Monday with the highest probability. From the graph in Figure 6-(b) it is obvious that the content delivery route should be from $19 \longrightarrow 95 \longrightarrow 102 \longrightarrow 14 \longrightarrow 24$ or $19 \longrightarrow 39 \longrightarrow 102 \longrightarrow 14 \longrightarrow 24$. However, considering the encounter duration and probability this route is not even a possible route. Figure 8, 9 and 10 shows the segregated encounter data for different times on Monday.

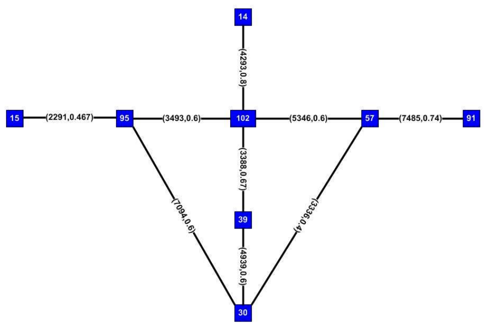

Fig. 7. Aggregated Backbone MSN Link Weights (Data of Mondays for 6 months).

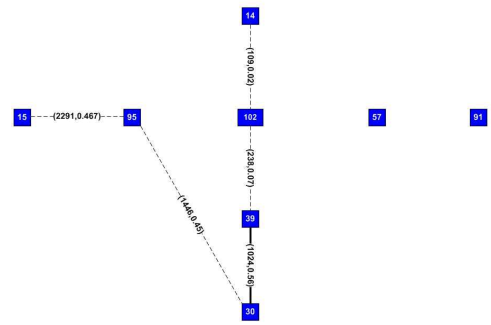

Fig. 8. Backbone MSN Link Weights (Monday Data for 6 months from 0-6 hrs)

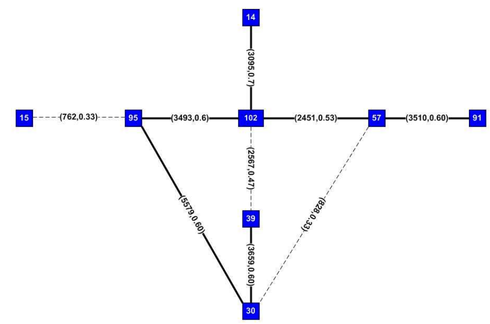

Fig. 9. Backbone MSN Link Weights (Monday data for 6 months from 6-12 hrs)

The steps of content dissemination from node 19 to 24 using the information of social encounters for different times of the day are illustrated. Node 19 only meets node 95 in the morning from 0 to 6 hrs. Therefore, the only option for node 19 is to transfer the data to node 95. During 0 to 6 hrs node 95 has a probability of 0.45 to meet node 30 that is the nearest to the destination (Figure 8). Having known this information node 95 will deliver the content to node 30.

During 6 to 12 hrs (Figure 9) node 30 has two options either to give the content to node 39 or node 57. Apparently node 30 has a better link with node 39 with the probability of 0.60 and average encounter time of 3659 sec. However, considering 12 - 18 hrs (Figure 10), in which data has to be delivered to the final destination. The combined probability of interaction of the nodes in the path $39 \longrightarrow 102 \longrightarrow 57 \longrightarrow 91$ is 0.06 and the combined probability of the path $30 \longrightarrow 57 \longrightarrow 91$ is 0.17. Therefore from 6-12 hr node 30 will deliver the content to node 57.

In the final time span from 12 to 18 hrs (Figure 10) the content will be delivered to 91 from node 57 and then node 91 will deliver the content to the final destination i.e. 24 (as 91 is connected to 24 in Figure 6-(b)). Finally the optimal path with highest probability and average time of encounter for Monday is $19 \longrightarrow 95 \longrightarrow 30 \longrightarrow 57 \longrightarrow 91 \longrightarrow 24$.

In the above scenario using real trace data [2] and with the help of simple example we have demonstrated our proposed approach of content delivery using MSN for time critical application.

VIII. Simulation Results

Simulation of MSN is still a topic of active research. MSN simulation environment is essential, as validation of protocols for these networks relies almost exclusively on simulations. Thus a simulation using a mobility model that captures the behavior of nodes in the real world is needed. The current simulations techniques use random models to generate dynamic MSN. However, the random models are not suitable for MSN simulations. In this paper we use Virtual Social Simulated Environment (VSSE) [3] a Second Life(SL)[6] based simulation environment, which can support dynamics in simulation models by allowing real users to participate using their avatars. SL is a virtual world simulator accessible via the Internet having more than five hundred thousand active users. SL can capture dynamics of movement models as avatars have different movement speed, different movement patterns and different neighbors.

In this paper we discuss the preliminary results of our analysis of two important parameters of the system, i.e. message carrier changes and message latency. Message carrier changes is a key parameter because it has a large impact on the network traffic. Network traffic is directly proportional to message carrier changes (hop count). In Figure 11 we compared the message carrier changes to the number of messages in the network. The proposed algorithm depends on the accuracy of the predicted patterns and the generality of the device profile. Therefore, if the network is not populated with messages the prediction algorithm will not be accurate and user profile will be specific to the user needs. Our results in Figure 11 confirm this effect. The career changes decrease with the increase in number of messages. This shows that our system can scale while lowering system wide traffic flooding.

Figure 12 shows a comparison between latency and number of messages. Latency depends on, the strength of the backbone social network, accuracy of time-stamps and most importantly the synchronization of the system clocks. These parameters are essential to find the best message carrier inside the mobile social network and a carrier selection can considerably decrease the message latency and assure timely delivery of the message.

[6]http://www.secondlife.com/

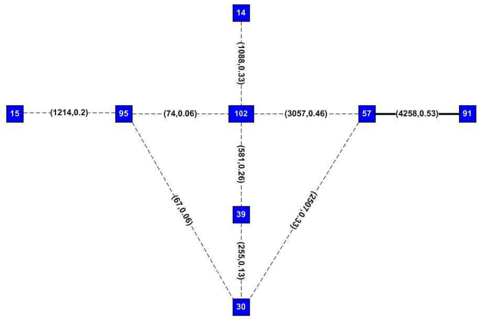

Fig. 10. Backbone MSN Link Weights (Monday data for 6 months from 12-18 hrs)

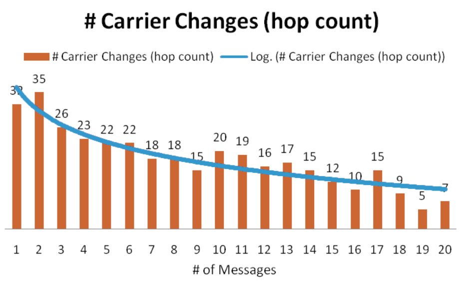

Fig. 11. Analysis of message carrier changes (hop count) in MSN for content transfer.

Figure 12 shows that latency is inversely proportional to number of messages. In our simulation we select one random node that forms the backbone mobile social network. As the simulation progress and message are exchanged the backbone social network adjusts its self. In our preliminary simulation based study we analyzed these two parameters to validate our claims of time assurance of message and lowering system wide network traffic. In future we plan to have a more in depth analysis of our proposed algorithms.

IX. Acknowledgment

This project is funded by National ICT Australia (NICTA). NICTA is funded through the Australian Governments Backing Australias Ability initiative, in-part through the Australian Research Council.

X. Conclusion

This paper presented time critical content delivery algorithms using predictable patterns in mobile social networks. The approach assumes that people follow similar patterns daily (i.e. Monday to Friday) and these patterns can be exploited to deliver the time critical content to interested subscribers. The social groups can be established by studying and deducing consistent/repeatable patterns based on the mobility history data. We have evaluated our approach and validated our assumptions using the real trace data of 100 users carrying Nokia 6600 smart phones over the course of nine months. Furthermore, we have simulated our algorithms on a participatory mobile social network simulation environment[3]. As the connectivity graph in our approach is based on the underlying mobility model therefore we used patterns based mobility model which is close to real life working day movement models[3]. The simple heuristics and initial study presented in this paper achieve a timely content delivery for time critical application using predictable patterns while lowering system wide traffic flooding.

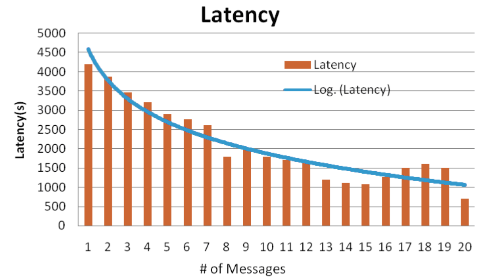

Fig. 12. Message latency analysis

References

[1] Paolo Costa, Cecilia Mascolo, Mirco Musolesi and Gian Pietro Picco *Socially-aware Routing for Publish-Subscribe in Delay-tolerant Mobile Ad Hoc Networks* In IEEE JSAC. Volume 26 June 2008.
[2] N. Eagle and A. Pentland , *Reality Mining: Sensing Complex Social Systems*, Personal and Ubiquitous Computing, Vol 10, 4, 255-268.(2006)
[3] Nazir. F, Prendinger. H, Seneviratne. A *Participatory Mobile Social Network Simulation Environment*, NICTA Tech Report ATP-1823 (2009).
[4] Ian F. Akyildiz, Xie. J, Mohanty. S, *A Survey of Mobility Management in Next-Generation All-IP-Based Wireless Systems* IEEE Wireless Communications, (August 2004). pp. 16-28.
[5] Nazir. F, Seneviratne. A, *Towards Mobility Enabled Protocol Stack for Future Wireless Network* Ubiquitous Computing and Communication Journal, (August 2007). Volume 2 Issue 4.
[6] Perkins, C.E, Royer, E.M, *Ad-hoc on-demand distance vector routing.* In proc. WMCSA '99 (1999), IEEE, pp. 90-100.
[7] Johnson, D., Maltz, D. *Dynamic source routing in ad-hoc wireless networks.* Mobile Computing (1996), pp 152-181.
[8] Perkins, C.E, Bhagwat, P. *Highly dynamic destination-sequenced distance vector routing (dsdv) for mobile computers.* In proc. SIGCOMM '94 (1994), vol. 24, ACM Press, pp. 234-244.
[9] Burgess, J., Gallagher, B., Jensen, D. Levine, B.N. *Maxprop: Routing for vehicle-based disruptive-tolerant networking.* In Proc. Infocomm 2006 (2006), vol. 4 IEEE, pp. 1688-1698.
[10] Grossglauser, M. Vetterli, M. *Locating nodes with ease: last encounter routing in ad hoc networks through mobile diffusion.* In proc. Infocomm '03 (2003), vol. 3, IEEE, pp. 1954-1964.
[11] Khelil, A., Marron, P.J, Rothermel, K. *Contact-based mobility metrics for delay-tolerant ad-hoc networking*, In proc. MASCOTS '05 (2005), IEEE, pp. 435-444.
[12] Mislove. A, Marcon. M, Gummadi, K. *Measurement and Analysis of Online Social Networks.* In proc. IMC '07 ACM (2007)
[13] Wasserman S, Fausr K, *Social Network Analysis*, Cambridge Press, Publisher, (1994).
[14] Nazir. F, Takeda. H, *Extraction and Analysis of Tripartite Relationships from Wikipedia*, In proc. IEEE International Symposium on Technology and Society (2008)
[15] Daly. E, Haahr. M, *Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs.* In Proc, MobiHoc '07 (2007)
[16] Miklas. A, Gollu. K, Chan. K, *Exploiting Social Interactions in Mobile Systems.* In proc. Unicomp 2007 LNCS (2007), Pp 409-428.
[17] Vahdat. A, Backer, D. *Epidemic routing for partially connected ad hoc networks.* Technical report cs-200006, Duke University (2000)
[18] Lindgern. A, et al. *Probablistic routing intermittently connected networks.* Lecture Notes of Computer Science 3126 (2004), pp 239-254
[19] Borcea. C, Gupta. A, Kalra. A, *The MobiSoC Middleware for Mobile Social Computing: Challenges, Design and Early Experiences*, In Proc. Mobilware '08 ACM (2008)