Exploiting Regularity of People Movement for Message Forwarding in Community-based Delay Tolerant Networks

Long Vu, Quang Do, Klara Nahrstedt
Department of Computer Science, University of Illinois
Email:{longvu2,quangdo2,klara}@illinois.edu

***Abstract*—It is well known that the daily movement of people exhibits a high degree of repetition in which people usually stay at regular places for their daily activities [1]. It is also believed that people usually stay in a social community [2]. In this paper, we exploit the regular movement patterns of community members to design a new routing protocol called UIMR, which forward messages of mobile nodes carried by community members. Particularly, UIMR utilizes the regular movement patterns of community members to construct the routing table and provides a decentralized solution to maximize the message delivery probability while preseving the message delivery deadline. We evaluate and compare the performance of UIMR protocol with Prophet routing protocol and Epidemic routing protocol over the real data set of 100 mobile nodes collected by 9 participants in the same research group at the University of Illinois campus. The experiment results show that UIMR outperforms other alternatives by improving the message delivery considerably while reducing the message overhead significantly.**

I. Introduction

It is well known that the daily movement of people exhibits a high degree of repetition in which people usually stay at regular places for their daily activities. [1]. It is also believed that people usually belong to certain social communities [2], where they have social relationships and social contacts with other community members. Therefore, exploiting these two observations to expedite message forwarding in Delay Tolerant Network (DTN) has drawn significant attention from DTN research community [3]. As a result, there have been two major approaches [1] to improve message forwarding of routing protocols in DTN research: (1) learning the past contact patterns resulted from regular patterns of people movement to predict future contacts, and (2) leveraging the social contacts among members within the same community.

For the first approach, the regular patterns from the previous contact history of mobile nodes was extracted and learnt to predict the future contacts for data forwarding [4], [5], [6], [7], [8], [9], [10], [11], [12]. The selected forwarder was the one who was likely to deliver the message or bring the message closer to the receiver. The best forwarder of the message is selected based on the contact probability between node pairs. However, the contact probability calculated by previous works [4], [5], [6], [7], [8] essentially represented the summarized contact between a node pair, which was similar to the contact summary oracle in the seminal DTN routing paper [3]. By using the summarized contact information, previous works did not fully leverage the regularity of people movement for message forwarding [13]. This is because people usually follow their daily schedules and thus they might repeatedly meet each other during a certain time period of a day, rather than everyday. For example, if the summarized contact probability between a node n_1 and a node n_2 is very low, n_1 is not selected as the message forwarder for n_2. However, in reality n_1 might have contacts with n_2 only during a certain time period τ of a day. As a result, although n_2's summarized contact probability with n_1 is low, the contact probability of n_1 and n_2 during the period τ might be still higher than those of other nodes during this period τ. In this case, n_2 should be considered as the potential message forwarder of n_1 for the period τ.

The second approach in DTN forwarding was to leverage the social relationships, social contacts, and centrality of people within the same community to expedite data forwarding among community members [14], [15], [16], [17], [18], [19]. Since in reality the members of the same community have a rich set of contacts during their daily activities, the forwarding performed by community members can be expedited by this set of contacts. In these previous works, the community was viewed as a graph of vertices, which were community members. The edge between a vertex pair exists if the two corresponding community members have social relationship. Each vertex has a centrality (i.e., node degree), which measures the popularity of a member in his community. The node with the highest centrality is selected as the best message forwarder. Similar to the summarized contact probability in the first approach, centrality essentially is a compression of the time-related contact information. In other words, regardless how often a pair of community members were in contact, there exists only one edge between them. As a result, like the first approach, the second approach did not fully exploit the regular movement patterns of community members for data forwarding.

It is clear that the summarized contact probability and centrality of mobile nodes did not exploit fully the time-related contact information. In this paper, we exploit the

[1]Besides other major approaches

time-related contact information in a more detailed level to capture the regularity of people movement. We focus on the community since we believe in reality people form social community.

In our previous paper [20], we presented the UIM framework to collect the joint Wifi/Bluetooth trace. The collected Wifi/Bluetooth trace was then used to construct a predictive model, which included location predictor, stay duration predictor, and people predictor [21]. In this paper, we exploit the time-related contact information by dividing time into type of day and time slot. Then, for each pair of type of day and time slot, we use the people predictor to predict the time-related contact probability [2], which then is used to design a new routing protocol called UIMR [3]. By dividing time into type of day and time slot, UIMR exploits the regularity of people movement to forward messages since people exhibits different movement patterns for different days and time slots. By leveraging the regular movement patterns of community members, UIMR maximizes message delivery probability while preserving message delivery deadline. We compare the performance of UIMR with Prophet routing [4] and Epidemic routing [22] over the real data set of 100 mobile nodes collected by 9 participants from the same research group in the Department of Computer Science, University of Illinois in March 2010. The results show that UIMR outperforms other alternatives by considerably improving the delivery time and reducing message overhead. In summary, this paper has following contributions:

1) We present a new routing protocol UIMR, which forwards the data messages for mobile nodes within the same community. Our protocol exploit the past contact history and construct the finer grain contact probability using type of day and time slot to improve data forwarding
2) We evaluate UIMR from real Bluetooth trace collected from 9 participants in the same research group at the University of Illinois. The evaluation results show that UIMR outperforms Epidemic and Prophet considerably.

This paper is organized as follows. We first present the Community-based Delay Tolerant Network in Section II. Then, we present the the design of UIMR in Section III. In Section IV, we compare the performance of UIMR with Epidemic routing and Prophet routing protocols. Finally, we conclude the paper in Section V.

II. A Community-based Delay Tolerant Network

We focus on a community whose community members have a rich set of social contacts due to their daily activities. In other words, a community member may have a high chance to have contacts with a certain portion of members of his community during his daily activities. We can find the community in numerous scenarios where people move within the same area of a university campus, an office, a cooperation building, etc. As a result, the community can be viewed as a "closed world" of mobile nodes. Because nodes in the same community usually have a rich set of contacts, in this paper we exploits this characteristic of community to forward the message among community members.

In reality each mobile node is carried by one person, we thus use the terms "community member", "mobile node", "node", "person", "Bluetooth MAC", "BT MAC" interchangeably. We consider the forwarding of the message m between two community members s and r, in which s is the sender and r is the receiver of m. A mobile node n follows its regular movement patterns thus n has regular contacts with other nodes within its community. Since the direct path between the sender s and the receiver r is not always available, our network belongs to the Delay Tolerant Network. In our context, the contact patterns between community members are represented by the contact probability, which is calculated by the people predictor in our previous work [21]. Basically, our people predictor used the collected Bluetooth contacts from Bluetooth trace to derive the contact probability. Then, we use the contact probabilities to construct the routing table of mobile nodes, which is used to forward the message m towards the receiver r.

III. UIMR: The UIM Routing Protocol

This section presents the design of UIMR protocol. First, we present the objectives of the protocol. Second, we present how to construct the routing table by using the people predictor in [21]. Finally, we present how the message is forwarded among mobile nodes in the network.

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

Table I
NOTATIONS USED BY UIMR PROTOCOL

[2] We use the terms time-related contact probability and contact probability interchangeably.
[3] UIMR stands for UIM Routing protocol.

A. Protocol Design Objective

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

$$\max \quad {P^n}_t$$
$$s.t.\ T_m \leq D_m \qquad (1)$$

UIMR obtains the objective function for the constraint of Equation 1 as follows. At the time t in the routing process, assuming that m is carried by a mobile n_1 and n_1 meets n_2, if ${P^{n_2}}_t > {P^{n_1}}_t$, then n_1 transmits m to n_2, which will carry and forward m towards r. n_2 will then find a node with better delivery probability to transmit m towards r. Our UIMR protocol keeps only a single copy of the message m among all the nodes in the network during the routing process to reduce message overhead. Since all the nodes in the network perform the same way, we are maximizing the delivery probability of m by the delivery deadline D_m. The constraint of Equation 1 is satisfied because T_m is taken into account in the calculation of ${P^n}_t$. For a mobile node n, we have $0 \leq {P^n}_t \leq 1$, which is derived from the routing table R_n of n. The routing table R_n is constructed by using the people predictor [21]. Since the mobile nodes are within a community, it is likely that the receiver r has contacts with a certain portion of community members. These contacts will help in forwarding m to r. In the following sections, we present the detail of UIMR protocol.

B. Constructing the Routing Table

With UIMR protocol, each node n has a routing table R_n, which is used to forward the message m during the routing process. R_n is basically a table with multiple columns and rows, which is similar to a relation in the relational algebra. Thus, in order to ease the presentation of the following sections, we use Relational Algebra [23] to represent and manipulate the routing table R_n. Henceforth, we use the terms routing table and relation interchangeably.

We exploit the regularity of people movement and the contacts among community members to construct the routing table R_n by classifying time into type of day and time slot. Particularly, we use the people predictor in [21] to calculate the probabilities that n has contact with one particular person for a type of day and during a time slot. These probabilities will be used to make the forwarding decision of the routing process. The intuition of classifying time into type of day and time slot is that people usually follow their scheduled routines for the daily activities. Therefore, a person may meet the same set of people for a specific time period in a particular day. So, we classify time of a day into time slot to capture this regular pattern. Also, the movement behavior of people may be different for the weekday and weekend, thus we classify time into type of day: weekday and weekend. These regular patterns of people movement exist in numerous scenarios such as university campus, workplace, etc.

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

Table II
AN EXAMPLE OF A ROUTING TABLE (OR A RELATION) R_n FOR TIME SLOT SIZE $\rho = 1$ HOUR.

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

The predictive model presented in [21] outputs three predictors including location predictor, duration predictor, and people predictor for a mobile node n. Let ${P^n}_p$ be the people predictor of the mobile node n. In this section, we leverage the people predictor to construct the routing table R_n.

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

Then, the set of queries χ is used as input for the people predictor ${P^n}_p$. For each query $X_k \in \chi$, ${P^n}_p$ returns a tuple $e_k =< \nu_k, \tau_k, {p^c}_{u_1}, {p^c}_{u_2}, ..., {p^c}_{u_j}, ..., {p^c}_{u_{|U|}} >$, in which ${p^c}_{u_j}$ is the contact probability between n and u_j for the type of day ν_k and during the time slot τ_k. Here, $1 \leq j \leq |U|$ and

$0 \le p^c{}_{u_j} \le 1$. Although the people predictor $P^n{}_p$ in [21] only returns the list of mobile nodes, which have a non-zero contact probability with n for a particular value of ν_k and τ_k, we can easily modify $P^n{}_p$ so that it returns the list of all mobile nodes by giving the zero probability for the nodes that do not have contact with n for ν_k and τ_k.

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

1) An Extensible Routing Table: The more n moves, n may encounter more mobile nodes in n's community. Therefore, R_n should be an extensible routing table that can grow when more mobile nodes are added into the set U. In other words, the relation R_n should be added more attributes or the table R_n should have more columns when n meets new community members. However, since in reality the community is usually a "closed world", the set of members n encounter will be eventually stable.

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

Since we have a collected trace of 20 days, in Section IV, we use the classification with type of day (not day of week) and time slot to construct the routing table and evaluate the UIMR protocol. The trace of 20 days may not be enough to construct the routing table for a finer grain classification with day of week and time slot.

C. Message Forwarding Decision

Given the routing table (or the relation) R_n of the mobile node n, R_n is used to route the message m from the sender s to the receiver r. This section presents how the node n_1 decides whether it should transmit m to another node n_2 when n_1 and n_2 encounter. Notice that since n_1, n_2, r are in the same community, they encounter each other with high probability.

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

As presented in Equation 1, during the routing process we prefer the node, which provides a higher delivery probability of the message m to the receiver r. Therefore, at time t when n_1 meets n_2, n_1 calculates its delivery probability $P^{n_1}{}_t$, which is the probability n_1 delivers m to r during the time period $[t, D_m]$. Similarly, n_2 calculates its delivery probability $P^{n_2}{}_t$. The two nodes n_1 and n_2 then compare their delivery probabilities. If $P^{n_1}{}_t < P^{n_2}{}_t$, m is transmitted from n_1 to n_2 and then n_2 becomes the only carrier of m in the network from that time. Otherwise, n_1 continues to be the only carrier of m in the network. The next question is how to calculate $P^n{}_t$ for a mobile node n.

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

$$E = \sigma_{\varphi=\{\nu=m_\nu, \tau\in\Sigma'\}} R_n \tag{2}$$

In Equation 2, φ is the condition of the selection operation over R_n, which basically filters out irrelevant tuples. In particular, relation E consists of only the tuples of R_n that have the type of day m_ν and the time slot in the set Σ', which is created as follows. For a time slot τ_i, let $\tau^s{}_i$ be the starting time of τ_i and $\tau^e{}_i$ be the ending time of τ. For example, if $\tau_i = (08:00, 10:00]$, then we have $\tau^s{}_i = 08:01$ and $\tau^e{}_i = 10:00$. For the duration $[t, D_m]$, we have $\Sigma' = \{\tau_i : \tau^s{}_i \ge t, \tau^e{}_i \le D_m\}$. For example, in Table II, if $m_\nu = weekday$, $t = 08:00$, and $D_m = 11:00$, then the relation E consists of the first three tuples as shown in Table III.

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

Table III
THE RELATION E WITH $m_\nu = weekday$, $t = 08:00$, AND $D_m = 11:00$.

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

$$S = \pi_{u_i=r} E \tag{3}$$

In Equation 3, we obtain the relation S by extracting the attribute $u_i = r$ from the relation E. In other words, the table S has only one column, which consists of the contact probabilities between n and receiver r obtained by projecting the column of receiver r in relation E. For example, in Table II, if $m_\nu = weekday$, $t = 08:00$, $D_m = 11:00$, and $r = u_3$, then $S = \{0.1, 0.6, 0.5\}$ as shown in Table IV. Since S has only one column, we use the term "set S", "relation S", and "table S" interchangeably in following sections. Formally, we have $S = \{p_1, p_2, p_3, ..., p_{|S|}\}$, where $0 \leq p_j \leq 1$, $1 \leq j \leq |S|$. p_j represents the contact probability between n and r during the time slot j^{th} and $p_j = 0$ means n and r have no contact during the j^{th} time slot.

$u_3 = r$
0.1
0.6
0.5

Table IV
THE RELATION S WITH $m_\nu = weekday$, $t = 08:00$, $D_m = 11:00$, AND $r = u_3$.

Notice that the order of elements in S corresponds to the order of the time slot in the relation E. That is, p_1 is the contact probability between n and r during the first time slot after time t, and $p_{|S|}$ is the contact probability between n and r during the last time slot before D_m. The set S is then used to calculate ${P^n}_t$, the probability that n delivers m at r during the period $[t, D_m]$ as follows:

$${P^n}_t = \sum_{i=1}^{|S|} {p^d}_i \tag{4}$$

In Equation 4, ${p^d}_i$ is the delivery probability that n successfully delivers m to r at the time slot i^{th}. ${p^d}_i$ is calculated based on contact probability in the set S as follows. We observe that node n only delivers m to r at the time slot i^{th} if n fails to deliver m to r in the first $(i-1)$ time slots and n successfully delivers m to r at time slot i^{th}. This happens only if n and r have no contacts during the first $(i-1)$ time slots and n and r have contact in the i^{th} time slot. So, we have:

$${p^d}_i = \prod_{j=1}^{i-1} (1 - p_j) \cdot p_i \tag{5}$$

The Equation 5 is used to calculate the delivery probability for each time slot i, and then ${P^n}_t$ is calculated accordingly in Equation 4.

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

2) Discussion: When the two nodes n_1 and n_2 have contact at time t, ${P^{n_1}}_t$ and ${P^{n_2}}_t$ are calculated by Equation 4 and compared, the node with the greater value of delivery probability is chosen as the next forwarder of m. Doing this, we obtain the objective function of the optimization problem in Equation 1 by maximizing the delivery probability of m in the routing process. Notice that the routing decision is made in a totally distributed manner since ${P^n}_t$ is calculated using the local routing table R_n of the node n and does not rely on information from other nodes. Also, since the set E is created based on the message delivery deadline D_m, calculation of ${P^n}_t$ takes into account the message delivery deadline D_m to satisfy the constraint of Equation 1. Therefore, we believe UIMR provides a robust and efficient routing solution for community-based DTNs.

IV. EVALUATION

A. Settings

We select a set Δ of 9 Bluetooth traces collected by 9 participants in the same research group in the department of Computer Science, University of Illinois from March 1, 2010 to March 19, 2010. The set Δ gives us 500 unique BT MACs. From this set of 500 Bluetooth MACs, we select a subset $D \subset \Delta$ of 100 Bluetooth MACs. In the set D, each BT MAC $b \in D$ encounters at least 3 out of 9 participants in the set Δ during the experiment period from March 01, 2010 to Mach 19, 2010. Since the participants in Δ come from the same research group and nodes in D have contacts, D represents contacts of a realistic community. If we select $D = \Delta$, then there exists BT MAC $b \in D$ that meet few other nodes in D, which may not form a realistic community since community members have too few contacts.

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

Table V
EXAMPLE OF A BLUETOOTH TRACE B COLLECTED BY ONE EXPERIMENT PHONE

Table V shows an example of a Bluetooth trace B collected by our UIM system. Here, each row in the table is the result

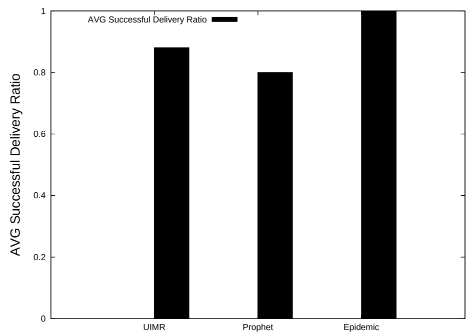

Figure 1. Comparison of Average Successful Delivery Ratio

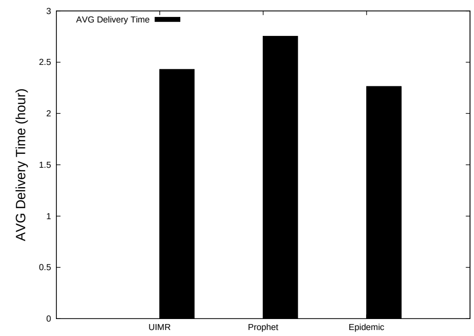

Figure 2. Comparison of Average Delivery Time for Delivered Messages

of one Bluetooth scan. The set D is created by projecting the column "Set of Scan Bluetooth MACs" from B. B is also used to infer contacts among mobile nodes in D to construct the routing table as discussed in Section III.

We then use D to evaluate the performance of UIM forwarding and compare its performance with three different protocols: Epidemic routing [22] and Prophet routing [4]. Epidemic routing is basically a flooding-based scheme which floods the message from the current forwarder to any new encountered nodes, which have not carried the message. This protocol provides a high delivery probability; however, it incurs a high message overhead. Meanwhile, Prophet uses the summarized contact probability between pairs of nodes for the entire time period rather than classifying time into type of day and time slot like what UIMR does. Then, the forwarding decision of Prophet is made based on this long-term summarized contact probability. Since the calculated probability is not for type of day and time slot, it does not capture the time-related contact information and it may not exploit fully the regularity of people movement in forwarding message.

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

There are two motivations to select the receiver r_i as above. First, as presented in a survey of 300 computer science faculty members and students [24], the forwarding delay that people can tolerate is from one to several hours, depending on the delay-tolerant networking applications and services [24]. We also observe that since the set D consists of BT MACs collected by participants in the same research group, nodes in D usually have contacts during the office hour period from 8AM to 6PM. So, by selecting r_i from the last 30 records of ${D^d}_{s_i}$, we basically set the deadline for the message transmission at the end of the office hour (i.e., around 6PM) and thus the delivery deadline is in the range of 8 to 10 hours. We believe this delivery deadline is reasonable in reality. Second, r_i is selected based on the set of ${D^d}_{s_i}$ means that applying Epidemic routing, s_i always can deliver the message m to r_i. Due to the construction of our testing set Γ, we expect that Epidemic routing will have 100% delivery ratio and a shortest delivery time since it floods the network with the message. We then use Epidemic routing as the base line in our performance comparison. In order to compare the performance of UIMR with Epidemic and Prophet routing protocols, we use three metrics Average Successful Delivery Ratio, Average Delivery Time, and Average Message Overhead.

B. Results

Figure 1 shows that Epidemic routing outperforms UIM routing and Prophet routing in terms of Average Message Delivery Ratio since Epidemic routing is essentially a flooding-based routing protocol. In Epidemic routing, a node n_1 which did not receive the message m will take a copy of the message m once n_1 has a contact with n which is carrying m. Since the receiver is chosen in the same day of the sender, epidemic routing will always can deliver the message to the receiver. Meanwhile, UIM obtains 89% of successful delivery since it exploits the regularity of people movement forward the message, which is not exploited by Prophet. This figure also shows that Prophet only obtains 78% successful delivery.

Figure 2 shows that Epidemic routing obtains the shortest delivery time of 2.26 hours. Correspondingly, UIMR obtains 2.42 hours and Prophet obtains 2.75 hours. Notice that this figure is only for delivered messages. In other words,

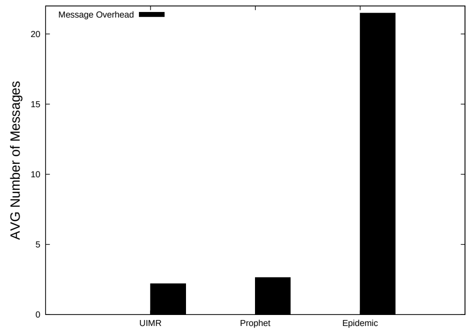

Figure 3. Comparison of Average Message Overhead

89% of messages delivered by UIMR and 78% of messages delivered Prophet are taken into calculation for this plot.

Figure 3 shows that while UIMR needs 2.2 messages, Prophet needs 2.65 messages to send one message m from the sender to the receiver (or to confirm that m misses the deadline), Epidemic needs 21.5 messages to send one message m successfully. So, Epidemic incurs 10 times of message overhead. For this plot, all messages are taken into consideration even if the transmission misses the delivery deadline.

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

V. Conclusion

In this paper, we present a new routing protocol called UIMR, which exploits the regular movement patterns of people within the same community to forward data messages. UIMR provides a distributed solution to calculate and utilize time-related contact probability for routing messages. We evaluate and compare UIMR with Epidemic and Prophet routing protocols over the real Bluetooth contact trace collected by 9 participants in the same research group at the University of Illinois campus. The results show that UIMR outperforms other alternatives and efficiently route the data messages to the receivers by the delivery deadline.

In reality, people usually follow their regular schedules to perform their routine works. At the same time, people have social relationships and people form social community. By exploiting regular movement patterns of community members to expedites data forwarding, UIMR is thus widely applicable.

References

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

[2] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Proceedings of ACM/SIGMOBILE Workshop on Multi-hop Ad Hoc Networks: from theory to reality (REALMAN 2006)*, 2006.

[3] S. Jain, K. Fall, and R. Patra, "Routing in delay tolerant network," in *Proceedings of Sigcomm*, 2004.

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

[5] S. C. Nelson, M. Bakht, and R. Kravets, "Encounterbased routing in dtns," in *Proceedings of Infocom*, 2009.

[6] H. Dubois-ferriere, M. Grossglauser, and M. Vetterli, "Age matters: Efficient route discovery in mobile ad hoc networking using encouter ages," in *Proceedings of Mobihoc*, 2003.

[7] E. P. C. Jones, L. Li, J. K. Schmidtke, and P. A. S. Ward, "Practical routing in delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 6, 2007.

[8] Q. Yuan, I. Cardei, and J. Wu, "Predict and relay: An efficient routing in disruption-tolerant networks," in *Proceedings of Mobihoc*, 2009.

[9] C. M. L. McNamara and L. Capra, "Media sharing based on colocation prediction in urban transport," in *Proceedings of Mobicom*, 2008.

[10] Z. Guo, B. Wang, and J.-H. Cui, "Prediction assisted single-copy routing in underwater delay tolerant networks," in *Proceedings of IEEE GLOBECOM*, 2010.

[11] S. Merugu, M. Ammar, and E. Zegura, "Routing in space and time in networks with predictable mobility," Tech. Rep., 2004.

[12] H. Cai and D. Y. Eun, "Aging rules: What does the past tell about the future in mobile ad-hoc networks?" in *Proceedings of MobiHoc*, 2009.

[13] T. Hossmann, F. Legendre, and T. Spyropoulos, "From contacts to graphs: Pitfalls in using complex network analysis for dtn routing," in *Proceedings of First IEEE International Workshop on Network Science For Communication Networks (NetSciCom 09)*, 2009.

[14] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *Proceedings of MobiHoc*, 2008.

[15] T. N. Dinh, Y. Xuan, and M. T. Thai, "Towards social-aware routing in dynamic communication networks," in *Proceedings of the 28th IEEE International Performance Computing and Communications Conference (IPCCC)*, 2009.

[16] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proceedings of Mobihoc*, 2007.

[17] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: A social network perspective," in *Proceedings of Mobihoc*, 2009.

[18] F. Li and J. Wu, “Localcom: A community-based epidemic forwarding scheme in disruption-tolerant networks,” in *Proceedings of SECON*, 2009.

[19] J. Niu, X. Zhou, K. Wang, and J. Ma, “A data transmission scheme for community-based opportunistic networks,” in *Proceedings of WiCOM*, 2009.

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

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

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

[23] “Relational algebra. http://en.wikipedia.org/wiki/relational_algebra.”

[24] E. P. S. Gaito and G. P. Rossi, “Opportunistic forwarding in workplaces,” in *Proceedings of ACM WOSN*, 2009.