GPS: A Method for Data Sharing in Mobile Social Networks

Bo Fan*, Supeng Leng*‡, Kun Yang† and Qiang Liu*
*School of Communication and Information Engineering,
University of Electronic Science and Technology of China, Chengdu, China
†School of Computer Science and Electronics Engineering,
University of Essex, Colchester, United Kingdom
‡Corresponding author, Email: spleng@uestc.edu.cn

***Abstract*—In Mobile Social Networks (MSNs), users with specific relationships are usually treated as a community for data sharing. However, the demand of data sharing among distributed strangers also exists. Those users that have the same interest but do not necessarily know or usually encounter each other can form a *gossip community* and share information. This paper proposes a data dissemination approach, i.e., the *Gathering Point-aided Spreading* (*GPS*) algorithm, which explores the encounter pattern of users and the aid of *gathering points* to facilitate data sharing in the gossip community. Based on the past encounter pattern, the GPS algorithm predicts the encounter probability among users and assigns the best users to carry the data for a wide spreading. Moreover, by storing a copy of data at the gathering points, GPS enables a further sharing of the data even the carriers leave the gathering points. With different utility functions, GPS can be modified into three versions (GPS-DR, GPS-DE and GPS-TR). Simulation experiments show that GPS outperforms SocialCast in both delivery ratio and delay in data sharing. In addition, among the three versions, GPS-DR and GPS-DE perform the best in terms of delivery ratio and delay respectively, while GPS-TR makes a tradeoff between them.**

***Index Terms*—Mobile Social Networks, gossip community, data sharing, gathering point.**

I. Introduction

Mobile Social Network (MSN) is a special type of Delay Tolerant Network (DTN), which explores the social attributes (interest, social relationship, daily schedule, etc.) of users to improve the performance of the services. In MSN, users with strong relationships (such as friends, one-hop neighbors, frequently encountered strangers, etc.) are usually regarded as a community that has a strong inner connection to share or help delivery data. For example, users with common interest keywords, similar location histories and nearby current location are recommended as friends for data sharing [1]. This idea is reasonable but confined. In fact, distributed strangers without the above relationships also have the demand for data sharing. For instance, the housewives in a town are usually interested in the discount in the supermarket and are willing to share the discount information if possible despite the weak relationship among them.

We define such a group of people that have common interest but are usually weak in relationship as a *gossip community*. They are located usually within the same piece of area but do not necessarily know or encounter each other, such as the housewives mentioned above. Different from traditional social communities [2][14] that have strong inner connection (social relationship, co-location, etc.), the gossip community has a loose connection among the community members. The only relationship among them is the common interest. Gossip communities widely exist in our social life, however, few proposal in MSN has focused on this type of community for efficient data sharing. In literature [3], a data sharing method is proposed. However, when the users are multiple hops away from each other, the sharing of data is based on the Ad hoc On-demand Distance Vector (AODV) [4] routing protocol, which is infeasible in MSN for the intermittent connection among users.

This paper proposes *Gathering Point-aided Spreading* (*G-PS*) algorithm, a data dissemination approach that adopts the idea of gossip spreading, namely first letting some gossipers get the information, and then the gossipers spreading it out quickly. The gossiper is assumed to be selfless to carry the data. In order to facilitate the sharing of data, GPS explores the past encounter pattern of users and the aid of Gathering Point (GP) for help. Based on the past encounter pattern among users, GPS predicts their future encounter probability and assigns the best users as data carriers (gossipers) that take the data for a wide spreading. Data carriers are helpful for the sharing of data, for they can take data to the users that are not going to encounter the data source and overcome the shortcoming of loose connection among members of the gossip community. The assignment of data carriers is based on the utility function of users. In this paper we establish three utility functions based on the predicted probability to select the best data carriers respectively aiming at three performance objects in data sharing: maximizing delivery ratio, minimizing delay and balancing the tradeoff between them.

The GP is the popular place that usually attracts a large number of users, such as a bar. It is very helpful in facilitating the spreading of data as it brings an opportunity for some interested users to share data directly and for the data source to select the excellent data carriers due to the large number of users gathering. In GPS, it is assumed that each GP has

ISBN 978-3-901882-58-6 © 2014 IFIP

the ability to store data, and a copy of each data is stored at each GP when the carriers get there, which ensures that the data can be further shared at the GP even though the carriers leave.

To the best of our knowledge, this is the first work that focuses on data sharing among the gossip community. Comparing with the existing work, the main contributions of this paper can be summarized as follows.

- We first focus on such a group of people that have common interest but are usually weak in relationship in data sharing and define them as a gossip community.
- A data dissemination method, GPS algorithm, is proposed for data sharing which explores the encounter pattern of users and the aid of gathering points to facilitate the spreading of data.
- A method to achieve different performance objects in data dissemination is proposed and three objects are achieved, i.e., maximizing delivery ratio, minimizing delay and balancing the tradeoff between them.

The rest of this paper is organized as follows. Section II provides a review of the related work about data delivery in mobile social networks, followed with the system model and basic assumptions of this paper presented in Section III. In Section IV, the detail of GPS algorithm is proposed, and utility functions are designed to guide the assignment of the data carriers in data dissemination aiming at different performance objects. Then, the theoretical analysis and simulation experiments are presented in Section V and Section VI. Finally, Section VII concludes this paper.

II. Related work

The sharing of data is based on data delivery techniques. In MSN, data delivery methods are widely studied which can be divided as single-recipient delivery and multiple-recipients delivery. The first category is also known as socially-aware routing [5][6][7][8] for the data is forwarded towards a specific destination node based on opportunistic routing techniques. A simple approach of this category is *Epidemic* [5]. In the approach, the source broadcasts data to all the users in connection and the users receiving the data then store and broadcast it to all the users encountered. In this way the data spreads until it reaches the destination node. It makes the biggest effort to achieve successful delivery at the cost of a large amount of network resource consumed. While in *Spray&Wait* [6], the number of copy of each data is restricted by the source who creates a specific number of copies for each data and sprays them into the network. Then each copy is carried by a relay node and waits to encounter the destination node during which no more copies are produced. *CAR* [7] is another socially-aware routing approach in which copies of data are carried by some active users to seek for the destination node. The active users are selected based on two user metrics, collocation and degree of connectivity, which respectively indicate the number and varying pattern of the one-hop neighbors. Before finding the destination node, if a more active user is encountered, the current holder of data forwards the data to the user to earn a larger delivery chance. A recently proposal [8] explores user gathering point to help routing in which data is also first disseminated to several relays similar with Spray&Wait. But in the next phase, instead of waiting for the encounter with the destination node, the method tries its best to forward the data towards the user gathering points to seek a larger delivery chance.

The multiple-recipients case is also known as *data dissemination* [1][3][9][10][11], for the data is disseminated to all the users who are interested in it. *SocialCast* [9] is a typical example in which the service provider publishes update data of the services to the subscribers via the help of data carriers. The assignment of data carriers is based on the knowledge of the current neighbors of users and their varying pattern. In literature [10], the service provider first publishes the update data to a single user which is selected at a certain probability. Then the update data is shared by users during which the newer data takes the place of the older one in the buffers of users. The publishing rate allocated to each user is optimized to make the data of the users as “fresh” as possible. The method in [11] focuses on improving delivery ratio. In the method, a constrained Markov decision process is adopted to model the data transmission in the network. Transmission policy is optimized to maximize the number of users that receive the data. The author of [1] proposes a method for data sharing in which users with common interest keywords, similar location histories and nearby current location are recommended as friends and connect for data sharing. While in [3], data is shared directly between two users as long as they are one-hop neighbors regardless of their relationships. However, when the users are multiple hops away from each other, the sharing of data is based on Ad hoc On-demand Distance Vector (AODV) routing protocol which turns out to be invalid in mobile social network due to the intermittent connection among users in the network [12].

Among the above approaches, socially-aware routing approaches are inefficient to support data sharing due to its end-to-end communication mode. Data dissemination methods are feasible, but the existing proposals neither consider the property of the gossip community, nor leverage the gathering habit of users, thus are also poor for data sharing in gossip communities. Therefore, we propose GPS to tackle this issue.

III. System model

This paper considers a distributed MSN (Fig. 1), in which data is delivered based on the carrying and forwarding of users without the help of base stations or APs. Gossip communities are identified by interests and data related with the interest is generated (randomly by a member) and shared among the same gossip community. Gossip communities are formed once the users set their interests in their social profile. They just exist potentially and users need not conduct community detection to discover it out before data sharing. After all, the relationship among the gossip community members are usually very weak to rely on.

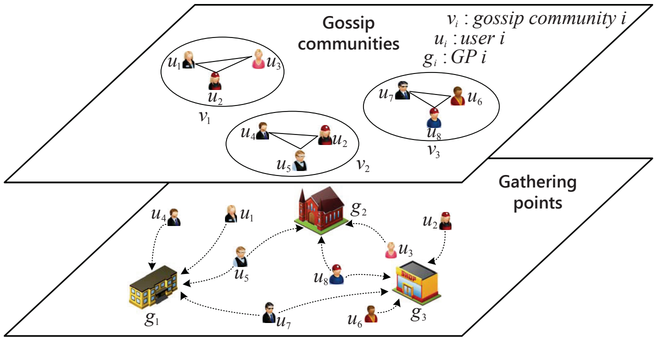


Fig. 1. Gossip communities and gathering points

Users tend to stay at several specific places and spend most of their time there in their daily life. This is confirmed by the conducted experiments [13] and is widely assumed in related studies [8][18]. Such gathering habit of users is greatly helpful for data delivery in MSN for it brings more forwarding chances among users. In this paper, we call the place that usually attracts a large number of users *Gathering Point* (*GP*). It is assumed that each GP supports a virtual throwbox [16], a mechanism to store data at a local storage device, or device of a user currently at the GP.

More explicitly, the network contains α users $U = \{u_1, u_2, \ldots, u_\alpha\}$ that form β gossip communities $V = \{v_1, v_2, \ldots, v_\beta\}$ with an interest topic shared in each community. In addition, there are β' GPs $G = \{g_1, g_2, \ldots, g_{\beta'}\}$ in the network that usually attract a large number of users. Each user belongs to at least one gossip community, namely has at least one interest. Data related with different interests is shared independently. In the remainder of this paper, unless otherwise specified, "interest user" indicates the user who wants to receive the interest data.

IV. Objects Modelling and Data Dissemination Algorithm

In this section, we first modify CPEM [15], a prediction mechanism proposed in our previous work to predict the future encounter probability among users. Based on that we establish three utility functions to assign the best users as data carriers respectively aiming at three objects in data sharing: maximizing delivery ratio, minimizing delay and balancing the tradeoff between them. After that, we propose GPS algorithm for data sharing among the gossip community.

A. Contact Probability Estimation Model (CPEM)

The basic idea of CPEM is based on the observation that human live with a relatively constant habit and tend to repeat some behaviors. For example, Catholics tend to go to the church on Sunday and students go to school on the weekday. With these habits, people tend to encounter the same group of people at specific places in particular time. Thus with the knowledge of the past encounter pattern of users, it is able to predict their future encounter pattern. In CPEM, the past

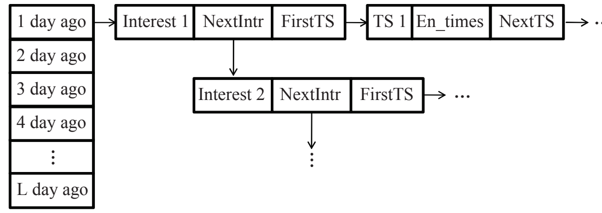


Fig. 2. Encounter Log

encounter pattern of the user is recorded in the *Encounter Log* as shown in Fig. 2. In the log, one day is divided into several time slots and the encounters happening in each time slot are recorded respectively in the corresponding entries of the log. The encounter with the same user will not be repetitively recorded during γ time slots, where γ is the reference size we are going to discuss in detail in IV-*B*. In addition, the encounters with people that have different interests, namely members of different gossip communities, are also recorded separately. In this paper, we modify the model as follows.

$$\begin{aligned} P_{ij} &= \frac{\sum_{k=1}^{L} \omega_k \cdot N_{ij,-k}}{|S_i|} \\ &\omega_k \in (0,1), \\ &\sum_{k=1}^{L} \omega_k = 1 \end{aligned} \tag{1}$$

For a given interest i, P_{ij} indicates the probability of encountering the interested users in time slot j of the day. $N_{ij,-k}$ is the number of interested users encountered in time slot j of k day ago and ω_k is the weight implying the contribution $N_{ij,-k}$ makes to the probability estimating. S_i denotes the amount of interested users in the network which can be learned from the Encounter Log during the initialization.

B. Utility functions

Since the members of the gossip community are usually weak in relationship, the sharing of data is usually aided by data carriers. The assignment of data carriers has a great influence on the performance of data sharing. For example, an active user who often meets a large number of persons is more capable to disseminate the data out if he is the carrier. In this case, to facilitate data sharing, we only need to optimize data carriers. In this paper, we focus on improving the performance of data sharing in terms of delivery ratio, delay and the tradeoff between them. To achieve the best performance in each of them, three utility functions are designed to optimize the assignment of data carriers.

1) Delivery Ratio (DR): Delivery ratio indicates the ratio of interested users that receive the data, which is greatly influenced by the social property (occupation, interest, etc.) of data carriers. For instance, a football coach who often gives lesson to the football lovers is more likely to disseminate football related data out. From the point of view of probability, a good carrier should have a large probability to encounter the interested users. To this end, we try to maximize the probability of data carriers to encounter the interested users. Namely

$$\max \frac{1}{c}\sum_{m=1}^{c} Prob_m \tag{2}$$

where c is the number of data carriers which is decided by the number of copies created by the source for each data. $Prob_m$ indicates the probability of the m-th carrier to encounter the interested users. It can be calculated with CPEM.

To realize the optimization of carriers shown in formula (2), an intuitive idea is to make $Prob$ as the utility function and assign the user with the largest utility as the data carrier. Based on CPEM, for a given interest i, the utility function is designed as follows.

$$Utility_{i,DR} = \sum_{cur<j<cur+\gamma} P_{ij} \tag{3}$$

cur is the index of current time slot and γ indicates the number of future time slots taken into account. Namely the probability of users to encounter the interested users within the following γ time slots is considered to decide the assignment of data carriers. We call γ *reference size* and are interested to study the influence of γ on the performance of the method.

2) DElay (DE): Delay of data sharing is the interval between the time when the data is generated and received. It is essential for data sharing especially when the shared data is time-sensitive. For instance, when sharing information about discount, a large delay in data delivery may make the information become useless, for the deadline of the discount comes before the information reaching the interested users. Delay of data dissemination is mainly caused by the wait before encountering the interested users. Therefore, the user who can encounter the interested users with the shortest wait without doubt is the best carrier to minimize delay. In this case, we try to minimize the average waiting time of the carriers to achieve the smallest delay in data sharing. Namely

$$\min \frac{1}{c}\sum_{m=1}^{c} |t_{m,next} - t_{cur}| \tag{4}$$

where $t_{m,next}$ is the time of the m-th data carrier to encounter the next interest user with a probability more than $P_{threshold}$ and t_{cur} is the current time. To minimize the average waiting time of the carriers, we set the reciprocal of the waiting time as the utility function. For a given interest i, $Utility_{i,DE}$ is designed as follows.

$$Utility_{i,DE} = \frac{1}{|TS_{i,next} - TS_{cur}|} \tag{5}$$

For simplicity, in $Utility_{i,DE}$ we explore the time slot in which encounters happen instead of the exact encounter time. $TS_{i,next}$ indicates the time slot in which the next encounter with the interested user happens, while TS_{cur} is the current time slot.

3) TRadeoff (TR): Delivery ratio and delay are two essential performance factors of data sharing. Unfortunately, it is usually impossible to achieve the optimum of both them at the same time. A pursuing of delivery ratio may lead to too much emphasis on a large delivery chance even it is in the far future and lead to a large delay. Also, in order to get a small delay, one may focus on the delivery that can happen soon in spite of the small probability of success. In this paper, besides delivery ratio and delay, we are also interested in the tradeoff between them for a balance performance. Based on $Utility_{i,DR}$ and $Utility_{i,DE}$, we make the tradeoff as follows.

$$Utility_{i,TR} = \sum_{cur<j<cur+\gamma} \frac{P_{ij}}{|TS_j - TS_{cur}|} \tag{6}$$

TS_j denotes time slot j, while P_{ij} and TS_{cur} are the same as before. $Utility_{i,TR}$ makes the user with both large probability and short waiting time stand out which is beneficial for both delivery ratio and delay.

Algorithm 1 *Gathering Point-aided Spreading*

Data Spray

for all $c_i \in C$ **do**
  if $encounters\ u_i$ and $CopyOf(u_i) = 0$ **then**
    give $\lfloor CopyOf(c_i)/2 \rfloor$ copies to u_i and
    assign u_i to be a new carrier;

Data Dissemination

for all $g_i \in G$ **do**
  if $u_i\ arrives$ and u_i is the interested user **then**
    send the data to u_i;
for all $c_i \in C$ **do**
  if $arrives\ g_i$ and $CopyOf(g_i) = 0$ **then**
    store an copy at g_i;
  if $encounters\ u_i$ and $u_i \notin C$ **then**
    if $Utility(c_i) < Utility(u_i)$ **then**
      assign u_i to be the new carrier;
    else if u_i is the interested user **then**
      send the data to u_i;

C. Data dissemination algorithm

Now we present the Gathering Point-aided Spreading (GPS) algorithm which can be modified into three versions (GPS-DR, GPS-DE and GPS-TR) for different performance objects using the established utility functions. Particularly, GPS-DR, GPS-DE and GPS-TR respectively adopt $Utility_{i,DR}$, $Utility_{i,DE}$ and $Utility_{i,TR}$ as $Utility$ in Algorithm 1 to achieve the maximum delivery ratio, minimum delay and a balance between them. For each data of the source, the sharing contains two phases, *Data Spray* and *Data Dissemination*.

In Data Spray, the source creates c copies of the data and sprays them to the carriers. Comparing with the adaptive spray [19], we adopt *Binary Spray* [6] scheme for a larger spray rate. The basic idea is that when the carrier/source that has more than one copy encounters a user with zero copy, he gives half of his copies to the user. The user receiving the copies becomes a new carrier (step $2-4$). The c copies are finally sprayed to exactly c carriers with each carrier taking one copy.

In Data Dissemination, the copies are carried by the carriers to wait for a delivery chance (an encounter with the interested users) during which one extra copy is generated and stored at each GP when the carriers get there (step $5-6$). This ensures that the data can be further shared at the GP even though the carriers leave. Interested users can receive the data from the GPs storing the data once they get there (step $2-3$) or from the carriers when they encounter each other (step $10-11$). When the carrier encounters a better carrier candidate who has a larger utility than itself, the carrier delivers the copy to the candidate and assigns it to be the new carrier. The carrier losing the copy become a normal node (step $8-9$). Then the data is taken by the new carrier to seek a delivery chance till the end of its set life (TTL). Control messages are sent before data delivery which contain interests, encounter information and recently received data of users. To avoid needless resource consumption, the GPs and carriers only forward data to the user that has not received it yet. The optimizing of carriers not only improves delivery efficiency, but also pushes the data towards the GPs for the user with larger utility is more likely to appear at the GPs. The maximum number of copies stored in the network is $c+\beta'$ (respectively at c data carriers and β' GPs). The two phases are not necessarily executed one after the other. In fact Data Dissemination begins once a carrier has exactly one copy of the data.

Algorithm 2 *Compute the expected delay*

1: Construct the transition graph;
2: Determine the state set S;
3: Determine $\phi_{s,s'}(t)$ for each pairwise $s, s' \in S$;
4: Set $f_{s_0,s} = 0$ ($\forall s \in S$);
5: Set $d_{in}(s) =$ in-degree of s ($\forall s \in S$);
Input: s_a;
6: **while** $S \neq \emptyset$ **do**
7: **for each** $s' \in S$ that $d_{in}(s') = 0$ **do**
8: $S = S - \{s'\}$;
9: **for each** $s \in S$ that $\phi_{s',s}(t) \neq 0$ **do**
10: **if** s' is s_0 **then**
11: $f_{s_0,s}(t) = \phi_{s',s}(t)$;
12: **else**
13: $f_{s_0,s}(t) = f_{s_0,s}(t) +$
14: $\int_0^t f_{s_0,s'}(\tau)\phi_{s',s}(t-\tau)d\tau$;
15: **if** $d_{in}(s') = 0$ **then**
16: $d_{in}(s) = d_{in}(s) - 1$;
17: **if** $d_{in}(s_a) = 0$ **then**
18: **Return** $f_{s_0,s_a}(t)$;
Output: $f_{s_0,s_a}(t)$;

V. Theoretical Analysis

In this section, we attempt to derive the delay in data sharing and study how the GP facilitate data sharing. A continuous Markov chain is applied to model data sharing and the delays in data sharing that leverages/do not leverage the GP for help are respectively computed.

A. Continuous Markov chains

We consider data sharing in a gossip community with 1 data source and n interested users that usually visit m GPs. To simplify the analysis, the optimization of data carriers is not considered. Namely the data is always taken by the initial carriers. We assume that the inter-meeting time between any two users and between a user and a GP follow exponential distributions with parameters λ and Λ ($\lambda \ll \Lambda$), respectively. This is widely adopted in many researches [8][17].

Firstly, we establish the states of the Markov chain. Each state is denoted as $s_{ij} = \langle i, j \rangle$ in which i ($0 \leq i \leq m$) and j ($0 \leq j \leq n$) respectively indicate the number of GPs and interested users that receive the data. The initial state is $s_0 = \langle 0, 0 \rangle$. Then we discuss the transition between any two states. State s_{ij} can only transfer to state $s_{(i+1)j}$ or $s_{i(j+1)}$ and the transition is irreversible. The transition stop at state s_{mn} where all the m GPs and n interested users have received the data. Transition probability is denoted as $\phi_{s,s'}(t)$ which indicates the probability density functions for the transition from state s to state s' at time t. We build the transition graph as Fig. 3.

B. Compute the expected delay

Now we calculate the expected delay for a particular number x of users to receive the data, namely the average time to transfer from the initial state s_0 to state s_{ix} ($0 \leq i \leq m$). The computing contains two steps. Firstly, the probability density function $f_{s_0,s_a}(t)$ for the state transition from state s_0 to an arbitrary state s_a is computed. Then we treat the states s_{ix} ($0 \leq i \leq m$) as a combined state s_x, compute the probability density function $f_{s_0,s_x}(t)$ for the transition from s_0 to s_x and finally derive the expected delay.

1) s_0 to an arbitrary state s_a: Consider two arbitrary states s and s' and the previous states $S_P = \{s_p | \phi_{s_p,s'}(t) > 0, s_p \in S\}$ of s'. The probability density functions $f_{s,s'}(t)$ for the state transition from s to s' satisfies

$$f_{s,s'}(t) = \sum_{s_p \in S_P} \int_0^t f_{s,s_p}(\tau)\phi_{s_p,s'}(t-\tau)d\tau \qquad (7)$$

which indicates that if the probability density functions for the state transitions from the state s to the previous states of s' have been calculated, then the probability density function for the transition from s to s' also can be derived. Based on that, we propose an inverse algorithm (Algorithm 2) of the one proposed in [8] to calculate the probability density function $f_{s_0,s_a}(t)$ for the state transition from the initial state s_0 to an arbitrary state s_a. In the algorithm, step 1-3 construct the transition graph. Step 4 sets all $f_{s_0,s}(t)$ ($\forall s \in S$) to be 0. In step 5 $d_{in}(s)$ is set as the in-degree of s which is used to decide that $f_{s_0,s}(t)$ is derived (when $d_{in}(s) = 0$) in the following steps. Step 8 deletes the states s' with 0 in-degree, namely the one whose $f_{s_0,s'}(t)$ has been derived. Step 10-14 compute $f_{s_0,s}(t)$. When $d_{in}(s_a) = 0$, $f_{s_0,s_a}(t)$ is derived.

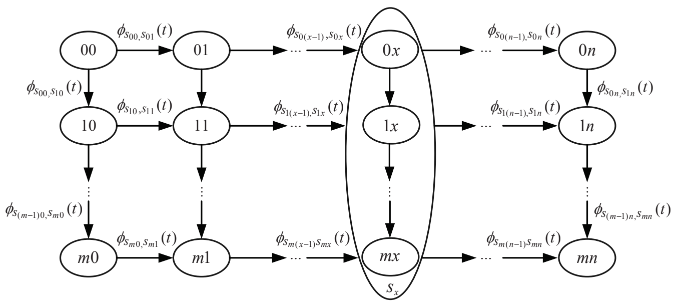
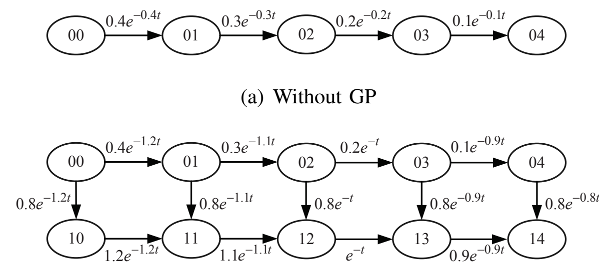


Fig. 3. Transition graph



(a) Without GP

(b) With GP

Fig. 4. Transition graph of the studied case

2) *s_0 to the combined state s_x:* Ignoring the inner transitions, we regard the states s_{ix} $(0 \leq i \leq m)$ as a combined state s_x which indicates that x users have received the data. Consider the previous states $S'_P = \{s_p | \phi_{s_p,s_x}(t) > 0, s_p \in S\}$ of s_x, the probability density function $f_{s_0,s_x}(t)$ for the state transition from s_0 to s_x satisfies

$$f_{s_0,s_x}(t) = \sum_{s_p \in S'_P} \int_0^t f_{s_0,s_p}(\tau)\phi_{s_p,s_x}(t-\tau)d\tau \qquad (8)$$

As each $f_{s_0,s_p}(t)$ $(s_p \in S'_P)$ is calculated in the previous step, $f_{s_0,s_x}(t)$ can be easily derived with formula (8). Then the expected delay for x users to receive the data can be derived as follows.

$$D_x = \int_0^\infty t f_{s_0,s_x}(t)dt \qquad (9)$$

TABLE I
SIMULATION PARAMETERS

Simulation parameters	**Default value**
ω_k	$\{0.2, 0.12, 0.12, 0.12, 0.12, 0.12, 0.2\}$[15]
Length of each time slot	10 minutes
Size L of encounter log	7 days
Radio range of devices	100 meters
TTL	$\{1, 2, 4\}$
Number c of copy	$\{2, 4, 8\}$
reference size γ	4
$P_{threshold}$	0.5

C. Case study

Consider a simple example which contains 1 GP and 5 users (1 data source and 4 interested users). The only data carrier is the source itself. $\lambda = 0.1$ and $\Lambda = 0.8$. When the aid of the GP is not considered, namely the GP does not store and forward data, the Markov chain contains 5 states. Otherwise, the Markov chain contains 10 states (Fig. 4). The probability density functions for the state transitions are also determined. For example, the transition from state s_{00} to state s_{10} indicates that the source get to the GP before encountering any one of the four interested users, thus

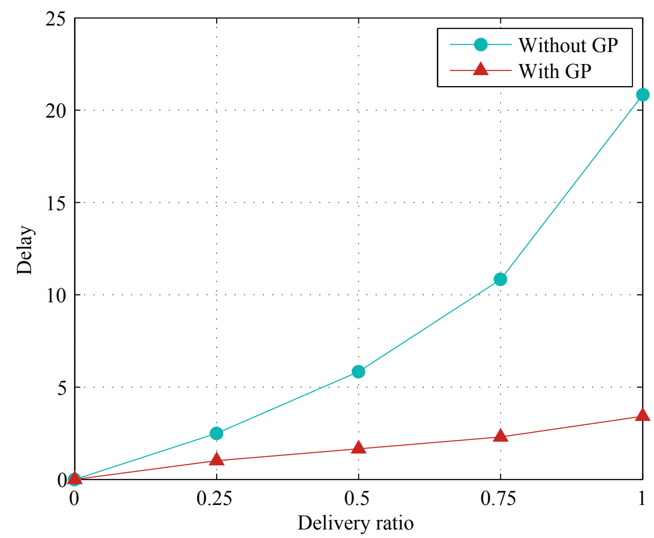


Fig. 5. Delay against delivery ratio

$\phi_{s_{00},s_{10}}(t) = \Lambda e^{-(4\lambda+\Lambda)t} = 0.8e^{-1.2t}$. Based on the two steps discussed in V-*A*, the expected delay for a particular number $(1-4)$ of users to receive the data is computed (Fig. 5).

The expected delay for the first interested user to receive the data is respectively 2.5 and 1.02, which indicates that with the help of the GP, the expected delay is decreased by 60%. Moreover, the expected delay for all the users to receive the data is respectively 20.8 and 3.4, implying that a decrease of 83.7% is achieved.

VI. PERFORMANCE EVALUATION

Simulation experiments are conducted on OPNET Modeler to study the performance of GPS algorithm in terms of delivery ratio and delay in comparison with *SocialCast* [9]. SocialCast is a data dissemination method that selects carriers based on two metrics, the current neighbors and the varying pattern of the neighbor list.

A. Parameter settings

The experiments focus on a campus scenario with a size of $4km \times 4km$ which contains 64 users and 4 GPs (dormitory, mess hall, teaching building and library). The 64 users form 4

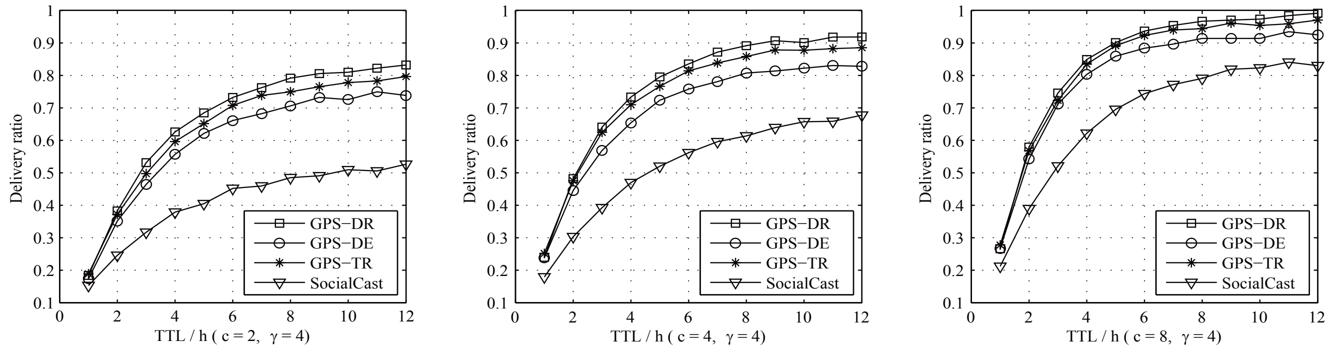


Fig. 6. Delivery ratio against TTL

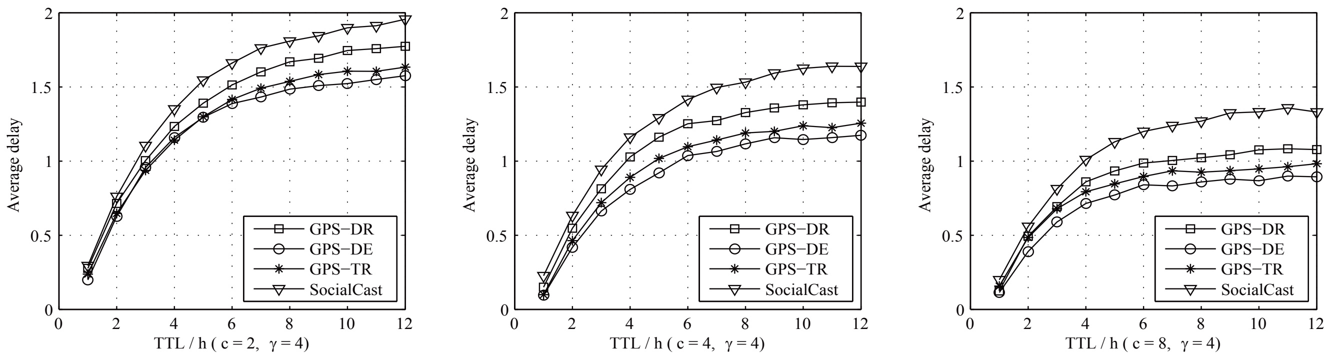


Fig. 7. Average delay against TTL

gossip communities that are respectively interested in 4 interest topics (sports event, campus anecdote, teaching announcement and dormitory notice) and each community contains 20 users with some users interested in more than one topic. For example, users with ID $1-20$ belong to community 1, $13-32$ belong to community 2, $33-52$ belong to community 3 and $45-64$ belong to community 4. The mobility of each user is set with Agenda Driven Model [20], a mobility model in which each user moves based on its individual agenda. The model creates an agenda for each node at the initialization based on its specific occupation and the selectable activity set. In this experiment, 8 occupations (student, teacher, librarian, etc.) and 12 activities (taking class, giving class, having meal, etc.) are set. Other specific parameters are given in Table I.

B. Results and analysis

Now we study the performance of the algorithms in terms of delivery ratio and delay, with respect to the TTL of data, the number c of copy of each data and reference size γ. From the point of view of average, delivery ratio is defined as the rate between the actual number of copy received by the interested users and the ideal one. While delay is the average delay of all users in receiving each data.

1) TTL: Firstly, we set the number c of copy and reference size γ to be the default value and study the performance of the algorithms with the increase of TTL. TTL is an essential parameter for delivery ratio because it decides the length of time the data stays in the system. With no doubt, a long stay will lead to more chances for a successful delivery. Indeed, we can see from Fig. 6 that the delivery ratio of the algorithms increase as TTL becomes larger. But, the increase of delivery ratio is not due to the increase of delivery chance within a given time, but more deliveries that experience a larger delay. This is exactly why the average delay also increases with TTL (Fig. 7).

Comparing the three GPS algorithms, it is easy to find that GPS-DR perform the best in terms of delivery ratio, while GPS-DE the best in average delay. GPS-TR makes a tradeoff between them with a balanced performance in delivery ratio and delay. This is because GPS-TR tends to assign the user with both large delivery probability and small delay to be the data carrier, while GPS-DR and GPS-DE just focus on one point. In addition, the three GPS algorithms outperform SocialCast in both delivery ratio and delay. This is not surprising for GPS makes a full use of the past encounter pattern of the users to assign the best carriers among the one-hop neighbors, while SocialCast only leverages the knowledge about the current neighbors of users and its varying pattern. Moreover, by exploring the aid of GPs, the GPS algorithms earn a greater advantage.

2) Number c of copy: We conduct the second experiment to study the performance of the algorithms with regard to the number c of copy by setting TTL and γ to be the default value. The number of copy created by the source is also an

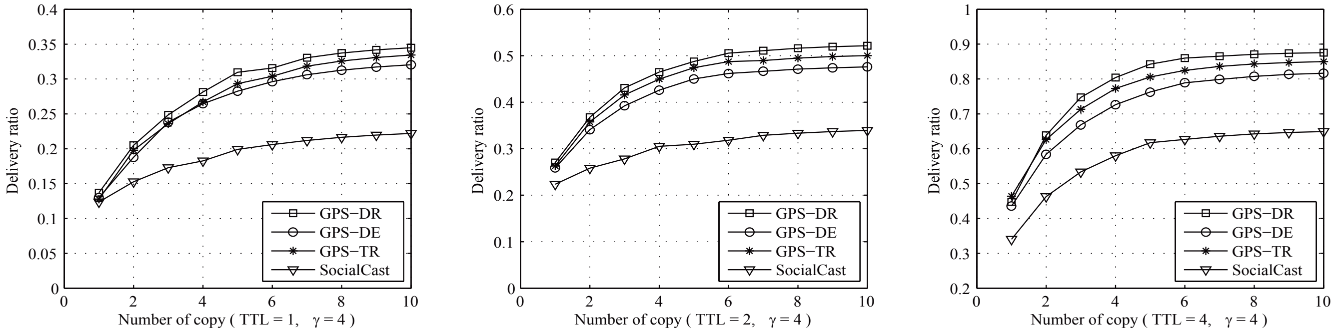


Fig. 8. Delivery ratio against number of copy

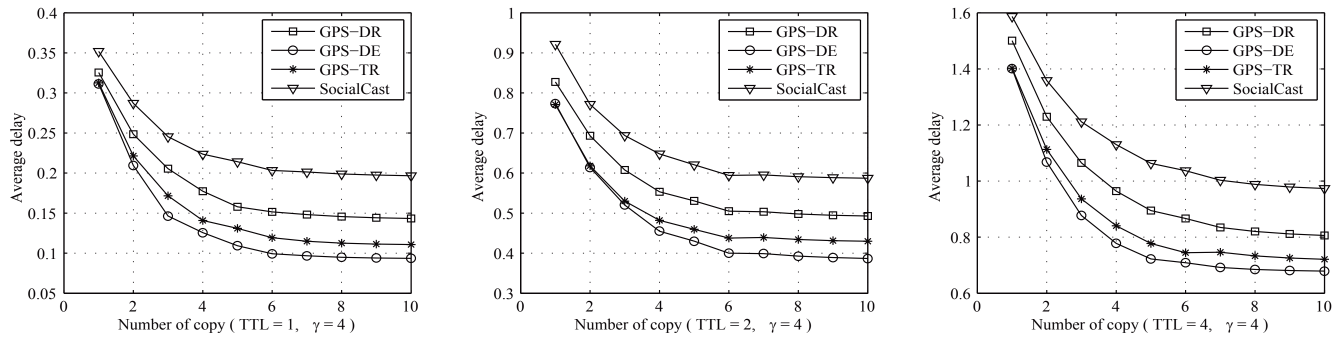


Fig. 9. Average delay against number of copy

importance parameter, for it directly decides how many data carriers exist in the network. A large number of carriers not only enlarges the scope the data can reach and earns more delivery chances, but also accelerates the spreading of data for more delivery is available within a given time comparing with a smaller c. Indeed, as shown in Fig. 8, delivery ratio increases as c becomes larger and finally levels off when c increases to a specific threshold value. Similarly, average delay decreases and levels out as c enlarges to the threshold value (Fig. 9). The threshold is decided by other parameters, such as TTL, and we are interested in studying it in the future work.

Similarly, from the comparison of the algorithms, it is easy to find that GPS-DR and GPS-DE respectively perform the best in terms of delivery ratio and delay, while GPS-TR gets a balanced performance between them due to the design of utility function. SocialCast performs worse than GPS algorithms for even with the same number of copy, SocialCast is not able to find the users that have the largest probability and smallest delay to encounter the interested users. Moreover, it fails to explore the help of the GP which can bring a great many delivery chances.

3) Reference size γ: We are also interested in the influence of reference size γ on the performance of the algorithms. γ is the number of time slot considered in utility functions, namely the method compares the probability of users to encounter the interested users in the future γ time slots to decide who is better. The value of γ directly decides the standard to select carrier. It can be found that When $\gamma = 1$, GPS-DR and GPS-TR perform the same (Fig. 10 and Fig. 11), as $Utility_{DR}$ is equal to $Utility_{TR}$ at this time. The performance is bad, for the carrier assignment is based on the encounter information in only one time slot. This is too rash to make a reasonable decision for it ignores some useful encounter information in the following time slots. In fact, the user who has a probability of 0.4 and 0.9 to encounter the interested users respectively in time slot 1 and 2 is a better carrier than the one who has a probability of 0.5 and 0.1. However, the algorithm chooses the latter as $\gamma = 1$. But it does not mean that the value of γ should be as large as possible. Too large a value of γ makes the one with a large delivery chance in the far future stand out in utility comparing which not only increases the delay of data forwarding, but also brings more uncertainty to the delivery, and decreases delivery ratio. That is why GPS-DR and GPS-TR perform worse in both delivery ratio and delay with γ increasing to 10. The optimal value of γ is 4 in this experiment where both GPS-DR and GPS-TR achieve a satisfying performance. The performance of GPS-DE always stays the same for $Utility_{DE}$ has nothing to do with γ.

VII. CONCLUSION

This paper focuses on a special group of users called gossip community, that consists of users that have common interest but are usually weak in relationship. Different from traditional communities in MSN, the gossip community has a weak connection among the community members. We study the way to realize efficient data sharing among them. Based on the past encounter pattern, the encounter probability among users is predicted and three utility functions are designed to assign the

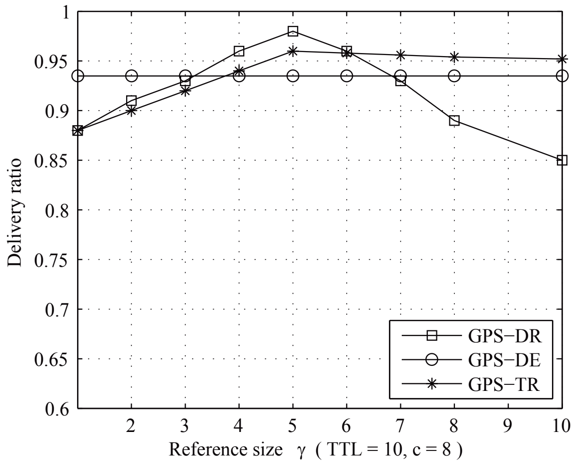


Fig. 10. Delivery ratio against reference size

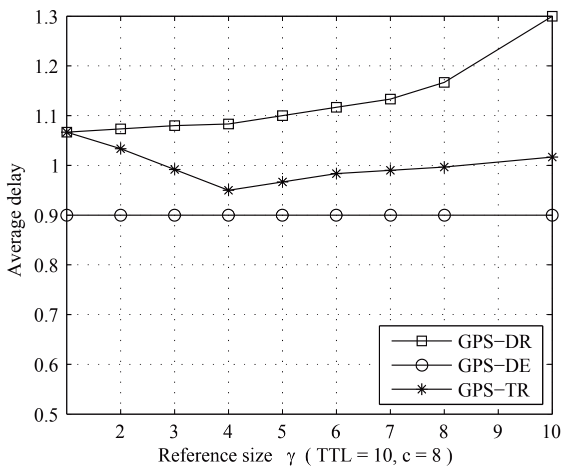


Fig. 11. Average delay against reference size

best users as data carriers, aiming at three performance objects respectively, i.e., maximizing delivery ratio, minimizing delay and balancing the tradeoff between them. After that, we propose GPS, a data dissemination algorithm for data sharing among the gossip community. In GPS, data source first sprays several copies of data to the data carriers and then the carriers take the data and spread them to other interested users. When the carriers arrive a gathering point, a copy of each data is stored there to ensure that the data can be further shared even the data carriers leave. Based on different utility functions, GPS can be modified into three versions (GPS-DR, GPS-DE and GPS-TR). Simulation experiments show that the proposed GPS algorithms outperform the existing method SocialCast in terms of delivery ratio and delay. The comparison among the GPS algorithms indicates that GPS-DR performs the best in delivery ratio and GPS-DE achieves the shortest delay, while GPS-TR makes a tradeoff between them with a balanced performance.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under Grant No.61374189, the Program for New Century Excellent Talents in University (NCET-10-0294), China, the Fundamental Research Funds for the Central Universities No. ZYGX2013J009, China, EU FP7 Project CLIMBER (PIRSES-GA-2012-318939), UK EPSRC Project DANCER (EP/K002643/1) and EU FP7 Project MONICA (GA-2011-2952220).

REFERENCES

[1] H. Li, K. Bok, J. Yoo, “An Efficient Mobile Social Network for Enhancing Contents Sharing over Mobile Ad-Hoc Networks,” in proc. 13th International Conference on Parallel and Distributed Computing, Applications and Technologies, 2012.

[2] R. Cabaniss, S. Madria, G. Rush, “Dynamic Social Grouping Based Routing in a Mobile Ad-Hoc Network,” 11th International Conference on Mobile Data Management (MDM),2010.

[3] B. Qureshi, G. Y. Min, D. Kouvatsos, M Ilyas, “An Adaptive Content Sharing Protocol for P2P Mobile Social Networks,” 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops.

[4] C. E. Perkins and E. M. Royer, “Ad-hoc on-demand distance vector routing,” in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90C100, February 1999.

[5] A. Vahdat and D. Becker, “Epidemic Routing for Partially Connected Ad Hoc Networks,” Tech. Rep. CS-200006, Duke University, 2000.

[6] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: an efficient routing scheme for intermittently connected mobile networks,” in Proc. the ACM SIGCOMM Workshop , pp. 252–259, August 2005.

[7] M. Musolesi and C. Mascolo, “CAR: Context-aware adaptive routing for delay-tolerant mobile networks,” IEEE Trans. Mobile Computing, Vol. 8, pp. 246–260, 2009.

[8] Jie Wu, Mingjun Xiao, “Homing Spread: Community Home-based Multi-copy Routing in Mobile Social Networks,” in Proc. IEEE InFOCOM, 2013.

[9] Paolo Costa, Cecilia Mascolo and Mirco Musolesi, “Socially-Aware Routing for Publish-Subscribe in Delay-Tolerant Mobile Ad Hoc Networks,” IEEE Journal on Selected Areas in Communications, Vol. 26, No.5, 2008.

[10] S. Ioannidis, A. Chaintreau, L. Massoulie, “Optimal and Scalable Distribution of Content Updates over a Mobile Social Network,” in Proc. IEEE INFOCOM 2009.

[11] D. Niyato, P. Wang, E. Hossain, “Optimal Content Transmission Policy in Publish-Subscribe Mobile Social Networks,” in Proc. IEEE Globecom 2010.

[12] Yanqin Zhu, Haojun Zhang, and Qijin Ji, “How Much Delay Has to Be Tolerated in a Mobile Social Network?,” International Journal of Distributed Sensor Networks, vol. 2013,2013.

[13] D. Kotz, T. Henderson, I. Abyzov, and J. Yeo, “CRAWDAD data set dartmouth campus (v. 2009-09-09),” Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/campus, Sep. 2009.

[14] Mingjun Xiao, Jie Wu, and Liusheng Huang, “Community-Aware Opportunistic Routing in Mobile Social Networks,” IEEE TRANSACTIONS ON COMPUTERS, 2013.

[15] Bo Fan, Supeng Leng, “Contact Probability based Routing Protocol for Mobile Social Network,” in Proc. ICC workshop on beyond social network: Collective Awareness, 2013.

[16] M. Ibrahim, P. Nain, and I. Carreras, “Analysis of relay protocols for throwbox-equipped dtns,” in WiOPT, 2009.

[17] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Efficient routing in intermittently connected mobile networks: The multiple-copy case,” IEEE/ACM Transactions on Networking, Vol. 16, No. 1, pp. 77C90, 2008.

[18] Yan Zhang, and Masayuki Fujise, “Location Management Congestion Problem in Wireless Networks,” IEEE Transactions on Vehicular Technology, Vol. 56, No. 2, 2007.

[19] Guangchun Luo, Junbao Zhang, et al., “Exploiting intercontact time for routing in delay tolerant networks,” Transactions on Emerging Telecommunications Technologies, Vol.24, pp. 589-599, 2013.

[20] Q. Zheng et al., “Agenda Driven Mobility Modeling,” Int'l. J. Ad Hoc and Ubiquitous Computing, vol. 5, 2010, pp. 22–36.