LASS: Local-Activity and Social-Similarity Based Data Forwarding in Mobile Social Networks

Zhong Li, Cheng Wang, Siqian Yang, Changjun Jiang, and Xiangyang Li, *Senior Member, IEEE*

Abstract—This paper aims to design an efficient data forwarding scheme based on *local activity and social similarity* (LASS) for mobile social networks (MSNs). Various definitions of social similarity have been proposed as the criterion for relay selection, which results in various forwarding schemes. The appropriateness and practicality of various definitions determine the performances of these forwarding schemes. A popular definition has recently been proven to be more efficient than other existing ones, i.e., the more common interests between two nodes, the larger social similarity between them. In this work, we show that schemes based on such definition ignore the fact that members within the same community, i.e., with the same interest, usually have different levels of local activity, which will result in a low efficiency of data delivery. To address this, in this paper, we design a new data forwarding scheme for MSNs based on community detection in dynamic weighted networks, called Local-Activity and Social-Similarity, taking into account the difference of members' internal activity within each community, i.e., *local activity*. To the best of our knowledge, the proposed scheme is the first one that utilizes different levels of local activity within communities. Through extensive simulations, we demonstrate that LASS achieves better performance than state-of-the-art protocols.

Index Terms—Local activity, social similarity, data forwarding, mobile social networks

1 Introduction

Mobile social networks (MSNs) combine techniques in social science and wireless communications for mobile networking. In a broader sense, a mobile social network is a mobile communications system which involves the social relationship of the users [1]. In such a network, users can publish and share information based on the social connections/relations among them [2]. Due to the ubiquitous availability of mobile devices, mobile social networks can fully take advantage of human interaction and physical proximity to achieve efficient and effective data delivery services.

In mobile social networks, there exist many *encounter-based* data forwarding algorithms. Recently, people have found that social information has big impact on data forwarding. Thus, some *social-aware and encounter-based* forwarding schemes receive enormous attention, see [3], [4], [5], [6], [7], [8], [9], [10]. Hui et al. [4] contributed a milestone work BUBBLE RAP to the data forwarding scheme through creatively exploiting node local centrality and social community structures. Gao et al. [5] investigated the multicast with known community structures for delay tolerant networks. Nguyen et al. [7] proposed an efficient scheme that delivers data to nodes having more common communities to gain a high data delivery ratio.

In the literature, one common implicit and critical assumption has been widely utilized: two nodes can contact with a higher probability if they have more social similarity. The measurement of social similarity differs in different approaches and one of the most popular methods is using common interests or common communities.

In this paper, we show that the measurement of social similarity based on the number of common communities ignores the fact that the members within the same community usually have different levels of local activity. A low local activity will result in a potentially low efficiency in terms of delivery ratio and latency due to the misalignment on the estimation of nodes' contact probability. Here, the concept of *local activity* is associated with a certain community. The local activity of node u in its belonging community i is the ratio of node u's encounter probability with other nodes in community i to any two nodes' encounter probability in community i. Note that the common concept of *activity* is about the whole network, not with a certain community of a node. The local activity of a node reflects a statistics of encounter probability in a node's certain community. More detailed explanation will be presented in Section 3.1.

As depicted in Fig. 1a, assuming Laura, Thomas and Stephen are the members of rugby club in the university, i.e., they have a common interest; now, a message must be sent to Laura through Thomas or Stephen. Thomas has many interactions with other members in the rugby club, i.e., he is with high local activity, while, Stephen has another common interest with Laura, e.g., both are in the university chorus. Therefore, it is unclear how to determine which person should be chosen as a relay due to lack of measurement

- Z. Li, C. Wang, S. Yang, and C. Jiang are with the Department of Computer Science, Tongji University, Shanghai, China. E-mail: {007lizhong, 3chengwang, youngfourkings}@gmail.com, cjjiang@tongji.edu.cn.
- X. Li is also with the Department of Computer Science, Illinois Institute of Technology, Chicago, IL, 60616 and the Department of Computer Science, Tongji University, Shanghai, China. E-mail: xli@cs.iit.edu.

Manuscript received 18 Sept. 2013; revised 26 Dec. 2013; accepted 30 Dec. 2013. Date of publication 24 Feb. 2014; date of current version 5 Dec. 2014.
Recommended for acceptance by K. Wu.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2014.2308200

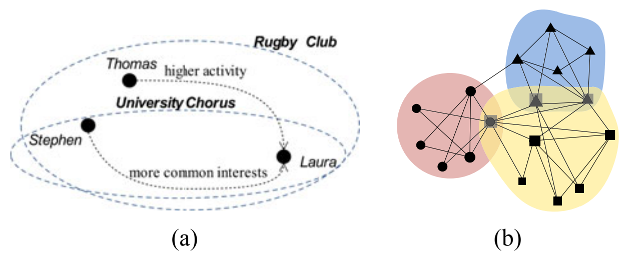


Fig. 1. (a) depicts the relay selection between Thomas and Stephen in message delivery. In (b), different sizes of icons represent each node's different levels of local activity in its communities. In the overlapping area, one node has different local activity in each belonging communities respectively, depicted by overlapping icons of square and triangle or square and circle.

criteria. If we choose Stephen as the relay, the delivery ratio may be low because of his potential lower local activity than that of Thomas.

In this paper, we need to choose a node having more common interests with the destination and having a higher local activity in the community as a relay. Therefore, we design a new data forwarding scheme for MSNs based on local activity and social similarity in dynamic weighted networks, called Local-Activity and Social-Similarly (LASS). Comparing with the existing schemes, LASS has two novel characteristics stated as follows: (1) LASS takes into account the diversity of members' internal activity within each community based on weighted network models, i.e., *local activity*, rather than making no difference to all members. (2) The global activity of a node can be defined by an *forwarding utility*. Its entries are the local activity of this node for all current communities. Then, we give a novel definition of the *social similarity* of two nodes by the *inner product* of their forwarding utilities, by which the forwarding scheme can be operated according to a simple rule of choosing the relay node with higher social similarity to the destination. A higher similarity means that the chosen node has more common interests with the destination and a higher local activity in the community. It guarantees a good data forwarding performance.

As illustrated in Fig. 1b, the different interest groups emerge through the community detection algorithm and the different levels of local activity are held by users. To the best of our knowledge, this proposed LASS scheme is the first one that utilizes different levels of local activity within communities; furthermore, the inner product based definition may lead to new ideas in measuring the social similarity among nodes. The major contribution of this paper is two-fold:

- We propose a data forwarding scheme LASS by investigating local activity and social similarity for mobile social networks. The *local activity* is defined to describe different levels of node internal activity within each community. It reflects a statistics of encounter probability in a node's certain community. Then, according to the community detection results, *forwarding utility* is formed by using the local activity. Meanwhile, the *social similarity* is also computed by the vector inner product method for settling our problem. Comparing with five data forwarding algorithms (Epidemic [11], PROPHET [12], Simbet [3], BUBBLE RAP [4] and Nguyen's Routing [7]), LASS performs best among them. For example, the popular algorithms-BUBBLE RAP and Nguyen's Routing, in terms of delivery ratio, LASS outweighs BUBBLE RAP 46.18 percent and Nguyen's Routing 34.64 percent. In terms of overhead ratio, LASS outweighs the suboptimal algorithm-Nguyen's Routing 41.7 percent in the five algorithms. The detailed experiment results are presented in Section 5.3.
- We give a self-adaptive weighted dynamic community detection algorithm (SAWD) so as to gain community structures and the node local activity for LASS. The *communication critical value* and the *weighted density embryo* (WDE) are defined to help detect communities. Both definitions assist us in avoiding some low weighted edges to shape meaningless communities. We classifies the adding or removing nodes or edges into "out-pool" and "in-pool" cases to deal with the dynamic network changes using local information. In Section 5.1.1, through Normalized Mutual Information (NMI) experiments, we obtain the optimal value of parameter γ to make SAWD having a good detection effect, i.e., the detection results approach to the ground truth.
- The rest of the paper is organized as follows. Section 2 provides the basic network model. Section 3 gives the designing of LASS data forwarding scheme. In order to obtain the node local activity, Section 4 gives the community detection algorithm SAWD. In Section 5, we establish our experiment environment and gain some interesting results about social relationships and data forwarding performance. In Section 6, we review some related studies. Finally, we conclude the paper in Section 7.

2 Network Model

2.1 Dynamic Weighted Graph

We model the mobile social network as a *dynamic weighted graph* which can be defined as a time sequence of network graph, denoted by $\mathcal{G} = \{G_0, G_1, \ldots, G_t, \ldots\}$, where $G_t = (V_t, E_t, W_t, F_t)$ represents a time dependent *network snapshot* recorded at time t, V_t denotes the set of nodes, $E_t = \{(u,v)|u, v \in V_t\}$ denotes the edge set, $W_t = \{w^t_{uv} \in [0,1)\,|\,u, v \in V_t \text{ and } (u,v) \in E_t\}$ denotes the set of weights on edges at time t, and $F_t : E_t \rightarrow W_t$ is a mapping that assigns weights to edges. Both node set and edge set change over time. For a node u, let d^t_u and $N_t(u)$ denote the degree and the set of all its neighbors at time t respectively. Specially, in our study, the value of w^t_{uv} denotes a ratio of the overall numbers of contacts between node u and v to the overall numbers of contacts for all nodes before time t. It reflects an encounter probability in mobile networks. Extended explanations about w^t_{uv} will be provided in Section 5.2 and Appendix F, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPDS.2014.2308200.

2.2 Community Structure

A community is a structure that has a group of tight-knit nodes with more internal links than external links, [13], [14], [15]. If people have common interests or encounter

TABLE 1
Main Notations

Notation	Meaning
$\mathcal{C}_t$	the community structure at time t
$N_t(u)$	the set of neighbor labels of node u at time t
$Com_t(u)$	the set of community labels of node u at time t
$\mathcal{C}_t(u)$	the set of all communities containing node u at time t
x_t	the average communication level at time t
$O_t(u,v)$	the weighted density embryo generated by (u,v) at time t
$\Phi(O_t(u,v))$	the weighted density function
$\Gamma(C_i^t, C_j^t)$	coupling coefficient
γ	the threshold of combining criterion
$a_{u,i}^t$	the local activity value for node u in community i at time t
$A_t(u)$	the forwarding utility of node u at time t
$SS_t(u,w)$	the social similarity between node u and w at time t

with each other frequently, they may form a community. However, the definition of community is quite subjective (most of the definitions are concerned with the special community detection algorithms or social applications), there is not a uniform definition about community. In our paper, we focus on the geographic proximity mobile social networks. People gather geographically and show the property of clustering. Thus, generally speaking, a community is defined as a group of nodes that are tightly connected in the form of contact frequency.[1] Here, let $\mathcal{C}_t = \{C_1^t, C_2^t, \ldots, C_k^t\}$ denote the *network community structure at time* t, i.e., a collection of subsets of V_t, where the element $C_i^t \in \mathcal{C}_t$ and its induced subgraph form a community of G_t; k is an integer and represents the number of communities at each network snapshot. Particularly, allow $C_i^t \bigcap C_j^t \neq \emptyset$, i.e., the network communities can overlap with each other. For a node u, let $Com_t(u)$ denote the set of labels of all communities containing u at time t, i.e., $\{C_i^t | i \in Com_t(u)\}$; let $\mathcal{C}_t(u)$ denote the set of all community structure containing node u at time t. Some notations are listed in Table 1.

3 LASS Data Forwarding Scheme

In this section, we propose our (LASS) data forwarding scheme.

3.1 Local Activity

Definition 1 (Local Activity). *Let $a_{u,i}^t$ denote the local activity of node u in a community with label $i \in Com_t(u)$ at time t. Then,*

$$a_{u,i}^t = \begin{cases} \dfrac{\sum_{(u,v)\in C_i^t} w_{uv}^t}{\sum_{(v',v'')\in C_i^t} w_{v'v''}^t} & i \in Com_t(u), \\ 0 & otherwise \end{cases}$$

where the numerator represents the sum of weight between node u and other nodes in community C_i^t and the denominator represents the sum of weight between any two nodes in community C_i^t. The discussion about a further efficient way for calculating local activity is provided in Appendix F, available online.

In Definition 1, the information of the community structure is required. The method about how to get it will be presented in Section 4.

Definition 2 (Forwarding Utility). *We define a forwarding utility $A_t(u) = (a_{u,1}^t, a_{u,2}^t, \ldots, a_{u,i}^t, \ldots, a_{u,k}^t)$ for each mobile node u at time t, where $a_{u,i}^t$ denotes the local activity value of node u in community i. The value of k represents the number of communities after applying the following SAWD community detection algorithm.*

A forwarding utility contains three-dimensional information: time, the number of communities, the local activity.

We give a metaphor to explain the meaning of the node local activity. Assuming there exists a rugby club (community) in a university. Two students Stephen and Thomas are belonged to this club. If Thomas has many interactions with other members in the club, while Stephen has few interactions with members, we can say, Thomas has a high local activity and Stephen has a low local activity. If there exists more than one community which Stephen and Thomas are belonged to, Stephen and Thomas will have different local activity in each community. In data forwarding, local activity is important because if the message is given to a node having low local activity, it will bring about a low efficiency in terms of delivery ratio.

We find different nodes in the same community have different local activity values and the same node in different communities has different local activity values. Next, we take advantage of the node local activity to develop the social similarity between two nodes.

3.2 Social Similarity

There are many kinds of social similarity measurements, such as cosine angular distance [16], Hamming feature distance [10], euclidean distance, the number of common communities (interesting groups) [7]. However, the above distance-based methods cannot give a meaningful explanation in real social networks. Of euclidean distance, the smaller the distance is, the more similar two vectors are. However, it is inappropriate in the physical sense in our paper. For example, the destination with forwarding utility (0.3, 0, 0.4). Now, there are two candidate relays. One is with forwarding utility (0.5, 0, 0.6), the other is with (0.3, 0, 0.4). If we use euclidean distance, the latter relay will be chosen. However, in practice, the former relay is better than the latter. The common interests-based method has a problem that if we choose a node having more common communities with the destination as a relay node, the chosen node may be a node with low local activity in its community. In this paper, we introduce the inner product method to define social similarity. It emphasizes the distribution of the components of a vector and seems like putting weight on each component of the destination vector.

Definition 3 (Social Similarity). *Given two forwarding utilities $A_t(u) = (a_{u,1}^t, a_{u,2}^t, \ldots, a_{u,i}^t, \ldots, a_{u,k}^t)$ of node u and $A_t(w) = (a_{w,1}^t, a_{w,2}^t, \ldots, a_{w,i}^t, \ldots, a_{w,k}^t)$ of node w, we define the social similarity between u and w at time t as $SS_t(u,w)$, having $SS_t(u,w) = A_t(u) \cdot A_t(w)$, where the symbol $\cdot$ denotes the inner product of vectors.*

1. This definition of community is based on the assumption that people gathering in the same location means that they tend to have the same aims or the same interests. It is a limitation of the designed system, i.e., our LASS data forwarding scheme suits for the scenario when the interest and contact matches.

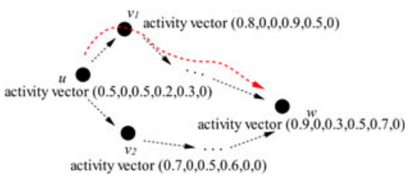


Fig. 2. Node u transmits a message to destination node w. At time t, u meets two nodes v_1 and v_2. Through LASS algorithm, finally, we choose node v_1 as the next hop. The data forwarding path is marked by the red dotted line with arrow.

Using the inner product-based method, we do not need to distinguish the different social features and it can guarantee the number of common interests and the high local activity. Besides, the method can deal with both uniform and nonuniform local activity distribution in vectors. A special case is the binary value of local activity. It is reduced to the case that the data forwarding only relies on the number of common interests.

All in all, the major difference is that the inner product in LASS actually is a summation of a node's probabilities to meet the destination node in each community, while previous methods mainly rely on the community structure to forward the packet: first send the package to the community of the destination, then use a certain metric to guide intra-community forwarding. The detailed comparisons and our improvement are provided in Appendix E, available online.

3.3 LASS Algorithm

Based on Definitions 1, 2 and 3, the description of LASS is presented in Algorithm 1. In order to make it clear, an example is also given to show the process of the data forwarding.

Algorithm 1 LASS: a session from node u to w at time t

```
for each encountered node v_i do
    calculate SS_t(u,w) and SS_t(v_i,w)
      if SS_t(v_i,w) > SS_t(u,w)
        add SS_t(v_i,w) to the set Temp^t
if Temp^t ≠ ∅
    sort the values in set Temp^t in descending order
    choose the largest SS_t(v_i,w) from Temp^t
      node u transmits the message to node v_i
else
      node u maintains the message
```

There is an example as shown in Fig. 2. At time t, node u transmits a message to destination node w in the mobile social network. Node u meets nodes v_1 and v_2. The vectors of node u, v_1, v_2 and w are

$$A_t(u) = (0.5, 0, 0.5, 0.2, 0.3, 0)$$
$$A_t(v_1) = (0.8, 0, 0, 0.9, 0.5, 0)$$
$$A_t(v_2) = (0.7, 0, 0.5, 0.6, 0, 0)$$
$$A_t(w) = (0.9, 0, 0.3, 0.5, 0.7, 0)$$

According to Algorithm 1, we calculate the social similarity and gain $SS_t(u,w) = 0.91$, $SS_t(v_1,w) = 1.52$, $SS_t(v_2,w) = 1.08$. From above results, node v_1 and v_2 can both use as the next hop. But $SS_t(v_1,w) > SS_t(v_2,w)$, so we finally choose node v_1 and transmit the message from u to v_1. After that, node v_1 keeps on doing the similar operations like above ways. The performance evaluation and fairness issues of LASS will be presented in Section 5.3 and Appendix F, available online.

4 Self-Adaptive Weighted Dynamic Community Detection

In LASS algorithm, we give the definition of local activity, forwarding utility and social similarity. In these definitions, one of the important elements is the information of the community structure. For example, in Definition 1, the belonging communities of node u; in Definiton 2, the number of communities. Thus, an efficient community detection algorithm is required to fix the community structure in our dynamic weighted graph. There have existed many static and unweighted algorithms [17], [18], [19], [20] and some of them can be transplanted to the dynamic weighted graph, but they have many problems. For example, K-clique-based algorithms [18], [19] require prior community information about K as inputs. It is not real for social networks, since we can not know the number of communities in advance, i.e., prior value of K. Some modularity-based algorithms [17], [20] have the problems of resolution limit and extreme degeneracy [21]. In addition, when meeting the dynamic environment, some algorithms need to repeat identification. The cost of time and computation is very high.

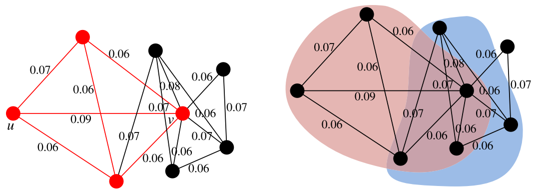
We refer to algorithm AFOCS in literature [7] and design our self-adaptive weighted dynamic community detection algorithm.[2] AFOCS is a newly overlapped and dynamic detection algorithm and avoids above-mentioned problems. However, it is only for unweighted graph and has some repeated operations on nodes and edges (Detailed analysis is presented in Section 4.2).

First of all, SAWD gives some weighted concepts and criterions (e.g., weighted density embryo, weighted criterion of communities) to handle the weighted graph. Then, SAWD classifies the adding or removing nodes or edges into "out-pool" and "in-pool" cases to deal with the dynamic network changes using local information. SAWD has two steps: 1) At first network snapshot, we use a centralized algorithm (Algorithm 2) to handle the mobile social network and identify the initial weighted community structure using a weighted criterion in Section 4.1.2) As time goes by, the structure of the network changes. We deal with the evolving structures using the local information of community structure, i.e., a dynamic tracking method in Section 4.2.

4.1 Initializing Community Structure

First, according to a criterion *weighted density* (Equation (1)) nodes are classified into different groups, i.e., raw communities. Then, on the basis of a *combining criterion of communities* (Definition 8), the highly overlapped raw communities will merge.

2. SAWD is designed based on AFOCS. The correctness verification has been given in literature [7]. Besides, some comparisons among AFOCS, K-clique-based algorithms and modularity-based algorithms have also been done in literature [7]. When K-clique-based and modularity-based algorithms are transplanted to the dynamic weighted graph, the similar comparison results will appear between them and SAWD. In this paper, we focus on the designing of data forwarding algorithm.



(a) (b)

Fig. 3. In (a), the red subgraph shows a weighted density embryo $O_t(u, v)$. Correspondingly, $E_t(u, v)$ is the set of red lines. In (b), the red community is generated by the edge weighted 0.09 and the blue community arises from the edge weighted 0.08. Two communities overlap with each other.

Definition 4 (Communication Critical Value). *Define the median value of the set of weights W_t as the* communication critical value *at time t, denoted by x_t.*

Communication critical value reflects a medial encounter probability or traffic between any two nodes in social networks. Thus, a sequence $\{x_t\}$ is formed over time. x_t is important because it can avoid some low weighted edges to form meaningless communities in social networks, i.e., those rarely meeting nodes can not form communities. The discussion about the choice of communication critical value is provided in Appendix F, available online.

Given a communication critical value x_t, we can obtain a spanning subgraph of G_t by deleting edges whose weights are smaller than x_t; we call such spanning subgraph *filtered graph*, and denote it by $G_t(x_t)$.

Before defining the *community*, we give a notion called *weighted density embryo*.

Algorithm 2 Constructing Initial Community Structure

Input: $G_0 = (V_0, E_0, W_0, F_0)$
Output: the set of initial communities $\mathcal{C}_{init}$
$x_0 \leftarrow$ apply Definition 4 on G_0
$E' \leftarrow E_0$
for each $w^t_{uv} \in W_0$
  if $w^t_{uv} < x_0$
    $E' \leftarrow E' \setminus (u, v)$
sort the edge weight in a descending order
from the largest weighted edge $(u, v) \in E'$
  if $Com_t(u) \bigcap Com_t(v) = \emptyset$
    find $O_t(u, v)$ according to Definition 5
    if $\Phi(O_t(u, v)) \geq \delta(O_t(u, v))$ and $|V_t(u, v)| \geq 4$
      $\mathcal{C}_{raw} = \mathcal{C}_{raw} \bigcup \{V_t(u, v)\}$
$\mathcal{C}_{init} \leftarrow \mathcal{C}_{raw}$
for $C^t_i, C^t_j \in \mathcal{C}_{raw}$ and !Done
  if $\Gamma(C^t_i, C^t_j) \geq \gamma$
    $C' \leftarrow$ combine C^t_i and C^t_j
    $\mathcal{C}_{init} = (\mathcal{C}_{init} \setminus \{C^t_i, C^t_j\}) \bigcup \{C'\}$
    Done $\leftarrow$ False

Definition 5 (Weighted Density Embryo). *Given an edge (u, v) at time t, an induced subgraph of $G_t(x_t)$ whose all nodes belong to $N_t(u) \cap N_t(v)$ is called x_t-level* weighted density embryo *generated by (u, v) at time t, denoted by $O_t(u, v; x_t)$.*

For brevity, denote WDE $O_t(u, v; x_t)$, the node and edge sets of $O_t(u, v; x_t)$ by $O_t(u, v)$, $V_t(u, v)$ and $E_t(u, v)$, respectively, without confusion.

An example of weighted density embryo $O_t(u, v)$ is depicted in Fig. 3a.

Then, we define the *weighted density* of WDE $O_t(u, v)$ by

$$\Phi(O_t(u, v)) = \frac{|E_t(u, v)|}{\binom{|V_t(u,v)|}{2}}. \tag{1}$$

Now, we can give a weighted criterion for determining whether a WDE is a *community*.

Definition 6 (Weighted Criterion of Communities). *A WDE $O_t(u, v)$ is a community iff the weighted density satisfies that $\Phi(O_t(u, v)) \geq \delta(O_t(u, v))$, where*

$$\delta(O_t(u, v)) = \frac{\binom{|V_t(u,v)|}{2}^{1-\frac{1}{\binom{|V_t(u,v)|}{2}}}}{\binom{|V_t(u,v)|}{2}}.$$

The threshold $\delta(O_t(u, v))$ is an increasing function [7]. It is a relaxation version of the traditional density threshold (complete graph). According to Definition 6, some nodes and edges can be grouped into different raw communities. However, there may exist some substructures which are highly overlapped. So, a combining criterion is necessary to help them merge into large ones. Before proposing the criterion, we give a notion called *coupling coefficient*.

Definition 7 (Coupling Coefficient). *For two weighted communities, say C^t_i and C^t_j, the* coupling coefficient, *denoted by $\Gamma(C^t_i, C^t_j)$, is defined as:*

$$\Gamma(C^t_i, C^t_j) = \frac{\sum_{(u,v)\in C^t_i \bigcap C^t_j} w^t_{uv}}{\min\{\sum_{(u',v')\in C^t_i} w^t_{u'v'}, \sum_{(u'',v'')\in C^t_j} w^t_{u''v''}\}} + \frac{\sum_{u\in C^t_i \bigcap C^t_j} \sum_{v\in C^t_i \bigcap C^t_j} w^t_{uv}}{\min\{\sum_{u'\in C^t_i} \sum^t_{v'\in C_i} w^t_{u'v'}, \sum_{u''\in C^t_j} \sum_{v''\in C^t_j} w^t_{u''v''}\}}.$$

The coupling coefficient is comprised of two parts, one is the intra edge weights ratio, the other is the intra node weights ratio. Based on it, we have

Definition 8 (Combining Criterion of Communities). *Two communities C^t_i and C^t_j should be combined, if their coupling coefficient $\Gamma(C^t_i, C^t_j) \geq \gamma$, where γ is a given threshold.*

Note that the parameter γ will be determined in the experiment in Section 5.1.1, i.e., we will choose an optimal value of γ that makes the community detection have a good effect. Fig. 3b shows two weighted overlapping communities. The discussion about bounds for combining criterion of communities is provided in Appendix F, available online.

The procedures of constructing initial community structure are described in Algorithm 2.

4.2 Dynamic Tracking Method

After constructing the initial communities, with the passage of time, the edge weights will vary due to strength changes of social relationships, such as new people making friends with each other, users joining in or withdrawing from the entire social network or local communities. So, we need to cope with the dynamic changes. It shows in two aspects, one is the physical mobility, the other is the strength changes of relationships. Here, we regard a network as a "pool". Reflected in the weighted graph, the changes can be classified into two types: 1) the number of nodes changes and the weight of edges also changes, called "out-pool" changes; 2) the number of nodes does not change but the weight of edges changes, called "in-pool" changes.

As soon as finding the social changes, the dynamic tracking method can deal with all the changes of nodes and edges simultaneously. The "out-pool" case includes adding foreign nodes to the current social network and removing nodes from the network. The "in-pool" case includes adding edges and removing edges operations. The detailed procedures are presented in Algorithm 4 $\sim$ 6 in Appendix A and B, available online. Some explanations about the tracking method are described as follows:

- Through checking the sets V_t and E_t, we can find the insertion and deletion actions of nodes and edges. Especially with varying edge weights, the adding and removing edges are found by Algorithm 3.
- For simplicity, we assume that every node has a community label set $Com_t(u)$, including solitary nodes. In final experiment results, if we find the number of nodes in a community is only one, we will discard it.
- We distinguish two types of nodes, one is *the foreign node* with its $Com_t(u) = \emptyset$ and its $N_t(u) = \emptyset$, i.e., it is not in the current network pool. The other is *the solitary node* with its $Com_t(u) \neq \emptyset$ and its $N_t(u) = \emptyset$, i.e., it is in the current network pool. Let CS^t denote the set of solitary nodes at time t.

Algorithm 3 Finding Changed Edges

Input: the community structures G_{t-1} and G_t
Output: the set of ΔE_t
$x_{t-1} \leftarrow$ apply Definition 4 on G_{t-1}
$E'_{t-1} \leftarrow E_{t-1}$
for each $w^t_{uv} \in W_{t-1}$
  if $w^t_{yv} < x_{t-1}$
    $E'_{t-1} \leftarrow E'_{t-1} \setminus (u, v)$
$x_t \leftarrow$ apply Definition 4 on G_t
$E'_t \leftarrow E_t$
for each $w^t_{uv} \in W_t$
  if $w^t_{yv} < x_t$
    $E'_t \leftarrow E'_t \setminus (u, v)$
compare E'_{t-1} and E'_t
  get the set of changed edges ΔE_t

Our object is to find a good community assignment which maximizes the overall internal weighed density function. According to above definitions and algorithms of SAWD, the high weighted substructures are clustered into different communities. As time goes by, the overall internal weighed density can always maintain the maximum value.

For the implementation of our community detection method SAWD, the detailed discussion is provided in Appendix C, available online.

5 Performance Evaluations

5.1 The Goodness of Fit for SAWD Detection Method

The goodness of fit for the community detection is measured by NMI score $N(X \mid Y)$ (Normalized Mutual Information) between *the proposed detection method results* X and *the ground truth/the benchmark results* Y. NMI score $N(X \mid Y)$ is an entropy method in information theory and detailed statement is provided in the following Section 5.1.3. The higher the NMI score is, the more similar the two community partitions are. Usually, if the experimental data set has the ground truth result itself, ground truth will be chosen as the comparison object; if the experimental data set has not the ground truth itself, the benchmark result will be used as the comparison object. According to our data set-MIT Reality Data Mining, we choose benchmark-LFR [22] (it is often used in many studies) to validate the goodness of fit for our SAWD detection method.

5.1.1 Parameter Choosing for Combining Threshold

Our detection algorithm (SAWD) does not need any prior user-input information about communities, e.g., the number of communities. The only parameter required to be fixed is the combining threshold value γ. γ is the combining threshold for two overlapped communities. An optimal value of γ is associated with the goodness of fit for our SAWD detection method. By the following benchmark experiments, we determine an appropriate value for γ to guarantee a good detection effect. Once gained, it will be used in the step of constructing initial static community and does not need to change its value in future dynamic operations. Moreover, it is only concerned with the detection method, not with the real mobile social networks.

5.1.2 Network Generation for NMI Experiments

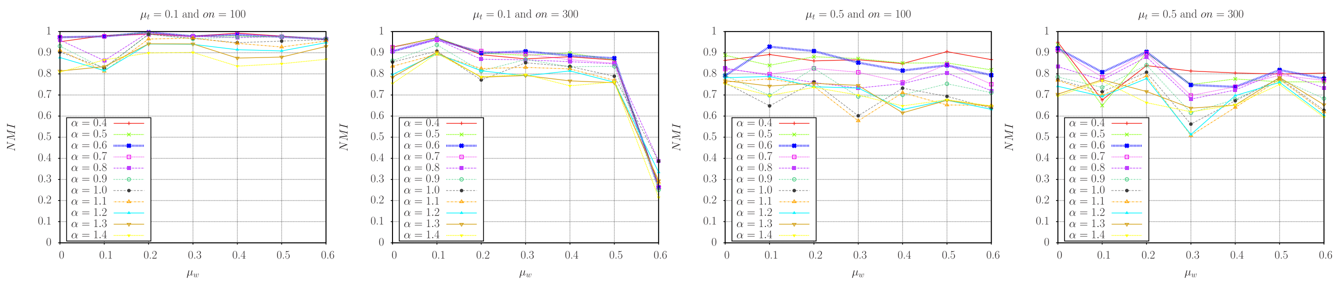
We choose *LFR undirected and weighted benchmark* [22] to generate a synthetic social network. That is to say it can produce undirected weighted graphs with possible overlapping communities and satisfies the power-law degree distribution. We refer [23] to choose some parameters: exponent for the weight distribution $\beta = 1.5$ and the number of memberships for the overlapping nodes $om = 2$. We freeze the number of nodes $N = 1{,}000$, topology mixing parameters $\mu_t = 0.1$ or $\mu_t = 0.5$ and the number of overlapping nodes $on = 100$ or $on = 300$. Then, we vary the weighted mixing parameter μ_w from $0 - 0.6$ to find the best value of γ.

5.1.3 Metrics

We use NMI overlapping version [24] as metrics, i.e., calculating the NMI score $N(X \mid Y)$. It is one of the most important entropy measures in information theory. $N(X \mid Y)$ can be interpreted as the average relative lack of information to infer random variable X given Y, $N(X \mid Y) \in [0, 1]$. The higher the NMI score is, the more similar the two community partitions are. If $N(X \mid Y)$ equals 1, it means the two kinds of community partitions are exactly coincident. Therefore, we set our detection algorithm (SAWD) as X and the LFR benchmark as Y.

5.1.4 Experiment Results and Analysis

From large numbers of tests, we obtain the combining threshold γ ranging from $0 - 1.8$. We select the representative values $0.4 - 1.4$ to get an appropriate value for γ. Because in this scope, the NMI score shows better than in other scopes. The experiment results are shown in Fig. 4.



(a) (b) (c) (d)

Fig. 4. Parameter choosing for combining threshold γ.

We can see, all curves declines as μ_w increases. μ_w indicates the fraction of the strength of a node which lies on links connecting the node to the nodes outside its community with respect to the total strength of this node [23]. The larger the μ_w is, the weaker the node strength is in its community. The weak strength results in the difficulty in community detection. Another view is from the horizontal and vertical perspectives. The small value of μ_t means a clear mixing topology and a dense community inner structure. The small on represents that the community structures approach to a disjoint status. The above two factors help us to find the community structure easily. Thus, Fig. 4a shows the best performance in NMI tests and Fig. 4d is the worst case in all tests. As depicted, through four different kinds of scenarios in benchmark experiments, in both the best and the worst cases, we gain the appropriate value for γ is 0.6. When $\gamma = 0.6$, the NMI score is highest among ten candidate γ values. NMI achieves 0.86 approximately on average. It means that the quality of the detected communities is good to close to the benchmark LFR. The communities identified by the method SAWD indeed have the structure that our proposed data forwarding scheme tries to exploit.

5.2 Constructing Social Graphs

The evaluation of LASS is based on MIT Reality Mining data set [25]. In trace files, 97 Nokia 6600 mobile phones were carried by users over the course of nine months in MIT campus and its surroundings. In the long-term observation, phone users use Bluetooth sightings with 5 minutes interval to record the direct contacts between nodes. The trace files offer insights into the real world interactions among mobile users from different aspects and constitute a valuable database for various studies. However, they do not provide the information of social relationship directly. We need to construct a weighted social graph using bluetooth device scanning records in those files.

Based on the fact that the encounter probability can reflect the strength of social relationship in MSNs [25], we construct our social graphs as following steps.

- We capture a period between date $2004-09-10$ and $2005-03-23$ from the original data since there are no significant groups of device contacts before and after this period. We use real trace from date $2004-09-10$ to date $2004-10-01$ to generate the initial community structure.
- We add the number of direct contacts between node pairs iteratively in chosen period t_0 to t_q. $\sum_{i=0}^{p} l_{uv}^{t_i}$ denotes the overall numbers of contacts between node u and v in time period t_0 to t_p, $\sum_{i=0}^{p} l_{*}^{t_i}$ denotes the overall numbers of contacts for all nodes. Thus, we have a matrix with $w_{uv}^{t_p} = \frac{\sum_{i=0}^{p} l_{uv}^{t_i}}{\sum_{i=0}^{p} l_{*}^{t_i}}$, where $0 \leq p \leq q$.
- For simplicity, we change the matrix to a symmetric one in order to cope with the invalidation of Bluetooth devices. Finally, we obtain a weighted matrix representing the social graph with social relationship links.

Some discussion about the data sets selection and the meaning of the edge weight will be seen in Appendix F, available online.

5.3 Data Forwarding Experiment

5.3.1 Algorithm Comparison

In this section, we compare our LASS algorithm against some encounter-based strategies (Epidemic [11], PROPHET [12], Simbet [3], BUBBLE RAP [4] and Nguyen's Routing [7]). Particularly, the last three have social-aware properties further.

5.3.2 Simulation Setup

We choose the ONE simulator as our experimental tool [26]. It not only provides various mobile models including some complex mobility scenarios in daily life, but also can incorporate real world traces. In MIT trace files, one of the most important records is the contact between Bluetooth devices. It includes the start time, end time and communication peers. These discrete contact events can be taken as the inputs of the ONE simulator. In order to model connecting and disconnecting, we reorder the start times and end times. Corresponding to communication peers, we set the start time as up and the end time as down. The form of the extracted trace data is like:

$$\begin{array}{ccccc}
0 & CONN & 93 & 96 & up \\
0 & CONN & 93 & 14 & up \\
128 & CONN & 85 & 17 & up \\
129 & CONN & 94 & 29 & up \\
 & & \cdots & & \\
1169 & CONN & 28 & 5 & down \\
1169 & CONN & 28 & 17 & down.
\end{array}$$

For all simulations conducted in this work, each node generates $1{,}000$ packets during the simulation time. The packet size is distributed from 50 to $100\ KB$ uniformly. Data

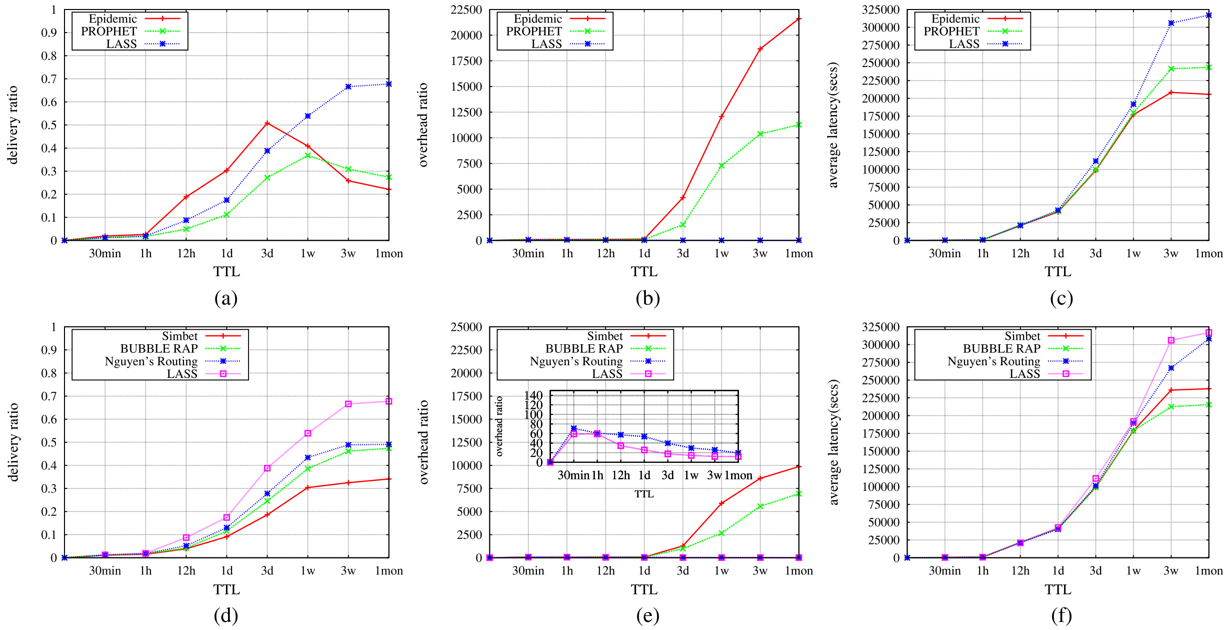


Fig. 5. Simulation results on MIT reality mining data set.

transmission speed is $2\,Mbps$ and the transmission range is in $10\,m$. The buffer size of each node is $5\,MB$. The source and destination pairs are chosen randomly among all nodes. Each emulation is repeated 20 times with different random seeds. Without losing precision, we set the update interval is 1. The interface of the underlying network is assigned to Bluetooth.

5.3.3 Metrics

- Delivery Ratio: the ratio of the number of successfully delivered messages to the total number of created messages.
- Average Latency: the average messages delay for all the successful sessions.
- Overhead Ratio: the proportion of the difference between the number of relayed messages and successfully delivered messages out of the successfully delivered messages.

5.3.4 Experiment Results and Analysis

MIT Reality Mining data set is a long-term observation repository. Thus, some cumulative social phenomena (local activity, community structure etc.) require a period of time to reveal. Here, we make TTL[3] from 30 to 1 mon. The experiment results are illustrated in Figs. 5a, 5b, 5c, 5d, 5e, and 5f.

Figs. 5a, 5b, and 5c show the delivery ratio, overhead ratio and average latency of LASS, Epidemic and PROPHET algorithms respectively.

In terms of delivery ratio, shown in Fig. 5a, due to Epidemic's flooding-based copy strategy, Epidemic performs better than PROPHET during the initial phase. At time three days, it reaches the peak value, close to 50 percent. But after that, the delivery performance decreases because of network congestion resulting from large numbers of copies.[4] Similarly, the turning point also appears in PROPHET at 1 week because it needs redundant relays to adapt to the fluctuation of meeting probabilities [4]. However, due to using the encounter history to predict the next hop, after the turning point, its delivery ratio shows better than Epidemic. By contrast, LASS goes up in steps and shows the best delivery performance among them, which outweighs Epidemic 32.63 percent and PROPHET 81.5 percent. At 1 month, it finally reaches 66.74 percent delivery ratio. Although LASS is also an unlimit copy algorithm like above two, its turning points will not emerge too early. This is because we use social similarity strategy to forward data. It means that the relay node has a high chance to meet the destination.

In terms of overhead ratio, illustrated in Fig. 5b, due to the nature of flooding, the disadvantage of Epidemic is obvious among the three algorithms. It exceeds 64.64 percent than PROPHET. However, thanks to its social similarity forwarding scheme, LASS performs best among them. To a great extent, through the whole TTL experiment period, the scheme controls the number of relaying copies with the overhead ratio only at 26.07 percent on average. In Fig. 5c, the delays of all the three algorithms arise with TTL increasing. However, due to Epidemic's large numbers of copies, it can achieve the lowest delay rapidly among three. PROPHET falls in between Epidemic and LASS. LASS is a little higher than PROPHET with 19.01 percent. This is because the balance effect between the number of copies and the precise relay choosing strategy, i.e., when we aspire to find a good data forwarding

3. We do experiments from date 2004-10-01 to date 2004-11-01. Because MIT is a long-term observation data set, we choose a large TTL-1 month, instead of several days. A larger TTL (larger than 1 month) also can be done with more simulation time. But through our experiment analysis, the overall trend is similar with 1 month.

4. Here, in order to avoid the serious declining of performance, we process Epidemic algorithm with copy-limits. Although the turning point still exists, its performance will not descend too much and will maintain relatively steady as time goes by.

scheme, the cost may be a little higher delay which is due to a relatively fewer copies.

Figs. 5d, 5e, and 5f show the delivery ratio, overhead ratio and average latency of our LASS, Simbet, BUBBLE RAP and Nguyen's Routing algorithms respectively.

In Fig. 5d, LASS performs best among four algorithms. Its delivery ratio is higher than Nguyen's Routing with 34.64 percent, BUBBLE RAP with 46.18 percent and Simbet with 120 percent on average. For Simbet, BUBBLE RAP and Nguyen's Routing, a common problem is that they do not consider contact frequency. In Simbet, BUBBLE RAP and Nguyen's Routing, the social graphs are formed according to that if there exists a contact between two nodes at time t, an edge will be added between the two nodes. That is to say, they do not concern the contact frequency in a certain period of time, only concern the contact at some time point. In Simbet, BUBBLE RAP and Nguyen's Routing, if the message is delivered to a node that has an edge (a contact) with the message holder, but this node has no contact with others in future, this delivering will lead to an invalid transmission.

If Simbet, BUBBLE RAP and Nguyen's Routing consider the contact frequency, the social graph will be formed according to our method, i.e., the edge will be added based on high contact frequency (edges filtering through the median value of the weight set in Section 4.1). Then, the improper relays with low contact frequency will not be chosen. A better delivery ratio will be possessed by Simbet, BUBBLE RAP and Nguyen's Routing than they had in the past. Thus, considering contact frequency to form a weighted social graph is one of essentials in our paper.

However, considering contact frequency is not the only decisive factor for LASS in our paper. Even if contact frequency is incorporated in Simbet, BUBBLE RAP and Nguyen's Routing, other problems restrict them. For Simbet and BUBBLE RAP, the problem is using the global betweenness in entire or partial phase of data forwarding. In Simbet and BUBBLE RAP, the concept of community is implicitly and explicitly considered respectively. That is to say, each node has its belonging community(ies), expect solitary nodes. If we deliver the message to a node having high global betweenness, although it indeed has high contact frequency with other nodes with respect to the entire network, it may be in a community which is irrelevant to or does not overlap with the destination community. As a relay, this node is not proper and increases the time of reaching the destination. For Nguyens Routing, it tends to send messages to nodes having many interests with the destination, however, it may deliver them to nodes which have low local activity in their communities (or interests groups). It is the main reason for the low delivery ratio of Nguyen's Routing. Therefore, in our paper, considering local activity in each weighted community and using social similarity to guide the routing path are the two important factors in improving data forwarding.

In Fig. 5e, the overhead ratio of Simbet and BUBBLE RAP are much higher than LASS and Nguyen's Routing. The reason is LASS and Nguyen's Routing prefer to choose the similar interests nodes as relays, which can control the number of copies in sessions. In the enlarged legend, the overhead ration of LASS and Nguyen's Routing are descending with TTL increasing. This is because both of them use social similarity strategy (LASS uses the inner product of forwarding utilities as similarity, Nguyen's Routing uses the number of common interests as similarity) to delivery message. As time goes by, the social phenomena becomes increasingly clearer, which makes the algorithms more and more suitable for the social network, i.e., just fewer copies can handle the data forwarding. On average, LASS keeps a low overhead ratio of 26.07. It is better than Nguyen's Routing which is 44.72. This is due to that LASS is good at finding high local activity nodes in pace with dynamic network. In Fig. 5f, the delays of four algorithms go up with TTL increasing. LASS and Nguyen's Routing are close to each other and slightly higher than BUBBLE RAP and Simbet. The gap is caused by few copies of LASS and Nguyen's Routing.

From above results and analysis, in terms of delivery ratio, we have LASS outweighs Epidemic 32.63 percent, Nguyens Routing 34.64 percent, BUBBLE RAP 46.18 percent, PROPHET 81.5 percent and Simbet 120 percent on average. In terms of overhead ratio, LASS performs much better than Epidemic, PROPHET, Simbet and BUBBLE RAP. For Nguyens Routing, it is a suboptimal algorithm with 44.72 overhead ratio on average. But LASS is only with 26.07, outweighing Nguyens Routing 41.7 percent. All in all, LASS has proved its competitive ability, which can achieve 66.74 percent delivery ratio and only have 26.07 percent overhead ratio with controlled delay in a reasonable range.

Besides, for the pros-and-cons of LASS, the role of local activity and social similarity of LASS and the performance under different parameter settings, we provide the experiments and analysis in Appendix G, H and I, available online.

6 Related Work

In mobile social networks, there exist some encounter-based data forwarding algorithms. Epidemic [11] and Spray-and-Wait [27] are simple encounter-based algorithms. After that, some studies [12], [28], [29], [30] use the history of node contacts, spatial information or contextual information to predict the future encounter probability. These heuristic methods aim at finding appropriate relay nodes who are like to meet the destination nodes.

Recently, people have found that social information has big impact on data forwarding. This is because some social relationships can reflect people's preference, which is important in node encounter prediction. Therefore, some social-aware and encounter-based algorithms emerge [3], [4], [5], [6], [7], [8], [9], [10], [31]. They can be classified into two kinds. The detailed descriptions and comparisons are provided in Appendix J, available online.

7 Conclusion

This paper designs a social-based data forwarding scheme for mobile social networks, called LASS, which measures the social similarity through nodes' different levels of local activity. Extensive simulations on MIT Reality Mining Database show that it can achieve a very good data forwarding performance in an efficient way. To the best of our knowledge, our work is the first one that utilizes different levels of local activity within communities; furthermore, it enriches the methodology of social similarity's measurement.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their constructive comments. The research of authors is partially supported by the National Basic Research Program of China (973 Program) under grant No. 2010CB328101, the Integrated Project for Major Research Plan of the National Natural Science Foundation of China under grant No. 91218301, the the National Natural Science Foundation of China (NSFC) under grant No. 61202383, 61003222, the Shanghai Foundation for Development of Science and Technology under grant No. 11JC1412800, the Program for New Century Excellent Talents in University (NCET) under grant No. NCET-12-0414, the Natural Science Foundation of Shanghai under grant No. 12ZR1451200, the National Science Foundation for Postdoctoral Scientists of China under grant No. 2012M510118, the Research Fund for the Doctoral Program of Higher Education of China (RFDP) under grant No. 20120072120075, the Special Foundation of China Postdoctoral Science under grant No. 2013T60463.

REFERENCES

[1] N. Kayastha, D. Niyato, P. Wang, and E. Hossain, "Applications, architectures, and protocol design issues for mobile social networks: A survey," in *Proc. IEEE*, vol. 99, no. 12, pp. 2130–2158, Dec. 2011.

[2] L. Jin, Y. Chen, T. Wang, P. Hui, and A. V. Vasilakos, "Understanding user behavior in online social networks: A survey," *IEEE Commun. Mag.*, vol. 51, no. 9, pp. 144–150, Sep. 2013.

[3] E. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proc. 8th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2007, pp. 32–40.

[4] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay-tolerant networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1576–1589, Nov. 2011.

[5] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: A social network perspective," in *Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2009, pp. 299–308.

[6] Q. Li, S. Zhu, and G. Cao, "Routing in socially selfish delay tolerant networks," in *Proc. IEEE Conf. Comput. Commun.*, 2010, pp. 857–865.

[7] N. P. Nguyen, T. N. Dinh, S. Tokala, and M. T. Thai, "Overlapping communities in dynamic networks: their detection and mobile applications," in *Proc. ACM 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 85–96.

[8] H. Li, C. Wu, Z. Li, W. Huang, and F. Lau, "Stochastic optimal multirate multicast in socially selfish wireless networks," in *Proc. IEEE Conf. Comput. Commun.*, 2012, pp. 172–180.

[9] K. Lin, C. Chen, and C. Chou, "Preference-aware content dissemination in opportunistic mobile social networks," in *Proc. IEEE Conf. Comput. Commun.*, 2012, pp. 1960–1968.

[10] J. Wu and Y. Wang, "Social feature-based multi-path routing in delay tolerant networks," in *Proc. IEEE Conf. Comput. Commun.*, 2012, pp. 1368–1376.

[11] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Dept. Comput. Sci., Duke Univ. Durham, NC, USA,Tech. Rep. CS-200006, , 2000.

[12] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.

[13] M. Girvan and M. Newman, "Community structure in social and biological networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, 2002.

[14] M. Porter, J. Onnela, and P. Mucha, "Communities in networks," *Notices Amer. Math. Soc.*, vol. 56, no. 9, pp. 1082–1097, 2009.

[15] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, nos. 3-5, pp. 75–174, 2010.

[16] F. Papadopoulos, M. Kitsak, M.Á. Serrano, M. Boguná, and D. Krioukov, "Popularity versus similarity in growing networks," *Nature*, vol. 489, no. 7417, pp. 537–540, 2012.

[17] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, no. 2, p. 026113, 2004.

[18] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[19] M.-S. Kim and J. Han, "A particle-and-density based evolutionary clustering method for dynamic networks," *Proc. VLDB Endowment*, vol. 2, no. 1, pp. 622–633, 2009.

[20] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech.: Theory and Exp.*, vol. 2008, no. 10, p. P10008, 2008.

[21] A. Khadivi, A. Rad, and M. Hasler, "Network community-detection enhancement by proper weighting," *Phys. Rev. E*, vol. 83, no. 4, p. 046104, 2011.

[22] A. Lancichinetti and S. Fortunato, "Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities," *Phys. Rev. E*, vol. 80, no. 1, p. 016118, 2009.

[23] A. Lancichinetti and S. Fortunato, "Community detection algorithms: A comparative analysis," *Phys. Rev. E*, vol. 80, no. 5, p. 056117, 2009.

[24] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New J. Phys.*, vol. 11, p. 033015, 2009.

[25] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," in *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 36, pp. 15 274–15 278, 2009.

[26] A. Keränen, J. Ott, and T. Kärkkäinen, "The one simulator for DTN protocol evaluation," in *Proc. 2nd Int. Conf. Simul. Tools Techn.*, 2009, pp. 55–64.

[27] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw.*, 2005, pp. 252–259.

[28] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," in *Proc. IEEE Conf. Comput. Commun.*, 2006, pp. 1–11.

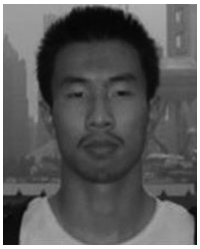
[29] J. LeBrun, C. Chuah, D. Ghosal, and M. Zhang, "Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks," in *Proc. IEEE 61st Veh. Technol. Conf.*, 2005, pp. 2289–2293.

[30] M. Musolesi, S. Hailes, and C. Mascolo, "Adaptive routing for intermittently connected mobile ad hoc networks," in *Proc. IEEE Int. Symp. World Wireless Mobile Multimedia Netw.*, 2005, pp. 183–189.

[31] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, and Y. Sun, "Geocommunity-based broadcasting for data dissemination in mobile social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 734–743, Apr. 2013.

Zhong Li received the BS and MS degrees from the Department of Computer Science and Technology, Shandong Normal University in 2007 and 2010, respectively. She is currently working toward the PhD degree in the Department of Computer Science, Tongji University in Shanghai, China. Her research interests include wireless communication, social network analysis, and distributed computing.

Cheng Wang received the PhD degree from the Department of Computer Science at Tongji University in 2011. Currently, he is a research professor at Tongji University. His research interests include wireless networking, mobile social networks, and mobile cloud computing.



Siqian Yang received the BS degree from the Department of Computer Science and Technology from Tongji University in 2012. He is currently working toward the PhD degree in the Department of Computer Science at Tongji University in Shanghai, China. His research interests include social network analysis, delay tolerant networking, and distributed computing.

Changjun Jiang received the PhD degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1995, and conducted post-doctoral research at the Institute of Computing Technology, Chinese Academy of Sciences, in 1997. Currently, he is a professor with the Department of Computer Science and Engineering, Tongji University, Shanghai. He is also a council member of China Automation Federation and Artificial Intelligence Federation, the vice director of Professional Committee of Petri Net of China Computer Federation, and the vice director of Professional Committee of Management Systems of China Automation Federation. He was a visiting professor of Institute of Computing Technology, Chinese Academy of Science; a research fellow of the City University of Hong Kong, Kowloon, Hong Kong; and an information area specialist of Shanghai Municipal Government. His current areas of research are concurrent theory, Petri net and formal verification of software, concurrency processing and intelligent transportation systems.

Xiangyang Li received MS and PhD degree at Department of Computer Science from the University of Illinois at Urbana-Champaign, 2000 and 2001, respectively, the bachelor's degree from the Department of Computer Science and the Department of Business Management from Tsinghua University, P.R. China, both in 1995. He is a professor at the Illinois Institute of Technology. He holds EMC-Endowed Visiting Chair Professorship at Tsinghua University. He is currently a distinguished visiting professor at Xi'An JiaoTong University, University of Science and Technology of China, and TongJi University. He received the China NSF Outstanding Overseas Young Researcher (B). His research interests include mobile computing, cyber physical systems, wireless networks, security and privacy, and algorithms. He published a monograph "Wireless Ad Hoc and Sensor Networks: Theory and Applications". He co-edited several books, including, "Encyclopedia of Algorithms". He is an editor of several journals, including *IEEE Transaction on Parallel and Distributed Systems, IEEE Transaction on Mobile Computing*. He has served many international conferences in various capacities, including ACM MobiCom, ACM MobiHoc, ACM STOC, IEEE MASS. His research is partially supported by the US National Science Foundation (NSF), National Natural Science Foundation of China, and RGC of HongKong. He is a senior member of the IEEE and a member of ACM.