A Socially Aware Routing Protocol in Mobile Opportunistic Networks

Ping Liu, Peiyan Yuan, and Chenyang Wang
School of Computer and Information Engineering, Henan Normal University, Xinxiang, China
Engineering Laboratory of Intellectual Business and Internet of Things Technologies, Henan Province
Email: {liuping, peiyan}@htu.cn; mkingsun@foxmail.com

Abstract —Routing is one of the difficult issues in mobile opportunistic networks, due to the lack of global knowledge and sporadic links. Related works use a greedy strategy to forward packets, i.e., they select relays with higher/bigger quality metrics. In this paper, we study the Opportunistic Routing Protocols (ORPs) from the perspective of social network analysis. Specifically, we discuss the impact of social relationship on the performance of ORPs. We first classify nodes into strangers and friends. We then explore the optimized number of strangers we can employ. Third, we propose STRON, a socially aware data forwarding scheme by taking both STRangers and their Optimized Number into account. We finally compare STRON with the state-of-the-art works through synthetical and trace-driven simulations, the numerical results demonstrate that STRON achieves a better performance, especially in terms of combined overhead/packet delivery ratio and the average number of hops per message.

Index Terms—Routing protocol, social forwarding, friends, strangers, mobile opportunistic networks

I. INTRODUCTION

Mobile Opportunistic Networks (MONs) provide a flexible way to forward packets for Internet of Things (IoT) in mobile scenarios. One of the features of MONs is that an end-to-end path between node pairs is rarely (if ever) existed at any moment, which makes routing very difficult in MONs [1], [2]. In this paper, we focus on the influence of social relationship on data forwarding efficiency within a more challenging environment, where the mobility of nodes cannot be known in advance and each node mainly depends on itself to locally estimate the forwarding metric.

The epidemic scheme [3] is a potential solution to deliver messages under the above scenario, because it tries to send each message over all possible paths of the network. The message therefore will be successfully received as long as one of the copies reaches the destination. However, the immoderate spraying will incur a high cost, resulting in the splurge on energy and buffer space, the rapid consumption of available bandwidth, and in turn, the possibility of degenerating the system performance in terms of the packet delivery ratio and average number of hops per message, etc.

These deficiencies motivate researchers to design novel data forwarding schemes, most of them make a tradeoff between the packet delivery ratio and cost by exploiting different contexts (e.g., social information [4], [5], contact information [6]). For these schemes, the data forwarding performance depends heavily on the contexts they used to evaluate the potential relays to the destinations. Furthermore, existing schemes take a greedy mechanism to deliver messages, i.e., only nodes with higher quality metrics than current carriers can be selected as relays to the destinations. As a result, the strangers have no chance to help others to communicate in the network. Whereas, the forwarding efficiency can be improved if permitting a few strangers to participate in the data forwarding process, this is mainly because the strangers have different spatiotemporal distributions from the current carriers [7]. Therefore, we need to address the problem of how to integrate strangers into data forwarding process. It is a critical while challenging question especially in a pure darkness environment.

Recently, there exit some works that explicitly take strangers into account. However, we notice that most of them need to know the mobility of each node and the global topology information. For instance, the authors of [8] exploited the community-detection mechanism [9] to classify nodes and took low number of contacts and short duration as a baseline to identify strangers, on the one hand, which requires the detailed traces of nodes (obviously, it may not be practical when considering the problems such as privacy protection and selfishness of nodes [10]), on the other hand, the centralized community-detection mechanism results in high cost due to the information exchange and calculations, and the complexity of adjusting several threshold parameters [11], [12]. Moreover, it is also challenging to ascertain the optimized values of several key parameters such as contact frequency, short duration and low number of contacts.

Taking all above issues into account, we investigate strangers from the following two aspects. (i) Identifying

Manuscript received October 17, 2016; revised January 22, 2017.
This work was supported by the National Natural Science Foundation of China under Grant Nos. U1404602, 51567018, the Young Scholar Program of Henan Province under Grant No.2015GGJS-086, the Science and Technology Foundation of Henan Educational Committee under Grant No.14A520005, the Dr. Startup Project of Henan Normal University under Grant No.qd14136, and the Young Scholar Program of Henan Normal University with No. 15018.
Corresponding author email: peiyan@htu.cn.
doi: 10.12720/jcm.12.1.62-71

stranger: The strangers are likely to be more useful for taking messages to different areas of the deployed region, which potentially increases the probability to encounter destinations. Thus, we focus on how to differentiate strangers from a large amount of nodes. (ii) The number of strangers we can exploit, which has a big influence on the system performance. Making it too few results in little improvement of data forwarding efficiency, making it too many deteriorates the performance of data forwarding schemes. As such, we have to ascertain the optimized number of strangers before integrating them into data forwarding process.

In this paper, we propose STRON (*STR*angers and their *O*ptimized *N*umber), an adaptive solution to address these issues. First, we employ local observations of nodes to estimate the similarity between them (i.e., each node only records contact duration between itself and others). We average these contact durations and use the mean as a baseline to identify strangers. Furthermore, each time a new contact was observed, the sum of the contact durations should be updated. Thus, the changing rate of similarity between nodes can be reflected dynamically. Second, we count the number of strangers which act as relays for other nodes on all shortest paths and use the mean ratio (i.e., the number of these strangers over that of all relays) to explore the optimized number of strangers we can employ. Finally, we design the data forwarding scheme by taking both strangers and their number into account. Our main contributions can be summarized as follows:

- We discuss the influence of the number of strangers on data forwarding efficiency and propose a method to explore the optimized number of strangers we can exploit.
- We develop centralized and distributed variants for the computation of the number of strangers that have received the message.
- We conduct extensive simulations to compare our scheme with the greedy mechanism based on a synthetical mobility model and real traces, the simulation results show that our algorithm largely outperforms the greedy mechanism, especially in terms of combined overhead/packet delivery ratio and the average number of hops per message.

The remainder of this paper is organized as follows. Section II reviews the related work. In Section III, we discuss how to integrate strangers into STRON. In Section IV, we make a performance evaluation. Finally, we conclude our paper and discuss some future research areas in Section V.

II. Related Work

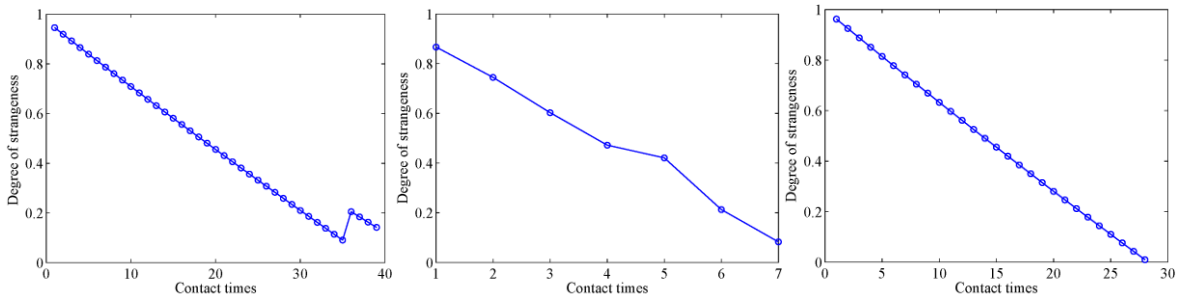
In the past few years, researchers have proposed several strategies to forward packets in MONs. According to the contexts they exploited, existing solutions can be classified into the following two categories.

A. Data Forwarding without Stranger

Data forwarding with extra nodes: The authors of [13]-[14] used the controlled mobility of extra nodes to facilitate message transmission in disconnected Mobile Ad hoc Networks (MANs). W. Zhao, M. Ammar and E. Zegura designed two kinds of no-random movements to forward data [15]. The first is node initiated mobility, where ferries move around the deployed region according to conventional routes. The second is ferry initiated mobility, where a ferry will adjust its trajectory to search the node when it receives a request from that node. They also evaluated the tradeoff between the incurred cost of extra ferries and the improved performance [16]. Besides, the authors of [17] further relaxed the assumptions of [15] and [16], they depended only on partial observations and statistical information of nodes mobility to enable ferries navigate themselves intelligently. Recently, the authors of [18], [19] employed static nodes placed in the system "hot region" to relay messages. If the message entered into the "hot region", the static node sprayed one replica of the message to each encountered nodes, otherwise, the message was sprayed in a binary way [20], [21].

Data forwarding with periodic mobility of nodes: In other scenarios (e.g., bus transportation system [22] and interplanetary internet [23]), the mobility patterns of nodes are repetitious, which motivates researchers to design periodicity oriented data forwarding schemes. Most of them took a modified Dijkstra algorithm to compute shortest paths between source-destination pairs, and then they constructed a routing table based on intermediate nodes along those paths. That is, they assumed that nodes had a global view on network topology. For instance, S. Jain et al. computed the shortest path by utilizing the periodicity of nodes movements [24]. Besides, the authors of [25] proposed a source routing scheme in delay-tolerant networks, they exploited the Expected Minimum Delay (EMD) as a forwarding metric and applied the Markov decision process to derive the EMD of messages.

Data forwarding with partial observations: Sometimes, it is difficult or impractical to acquire global information of network topology, this is mainly because of the problems including time-varying links, privacy protection or selfishness of nodes, etc. In these cases, different local contexts can be exploited to improve system performance. For example, the authors of [5] presented LASS, which exploited local activity and social similarity to select relays. The authors of [26] proposed PER, a prediction and relay algorithm for MONs, which considered the time of a contact. P. Yuan et al. presented OPPO, in which the transient contact ratio of nodes was used to spray data copies proportionally. CAR (context aware routing) exploited the changing rate of neighbors of a node and its current energy level to estimate the delivery probability [27]. In addition, J. Leguay et al. presented MobySpace, a high-dimensional Euclidean space constructed by the past motion patterns of nodes [28]. A recent survey on this topic can be found in [29].



(a) CMM (NodeID=11) (b) KAIST (NodeID=21) (c) NCSU (NodeID=25)

Fig. 1. Degree of strangeness under different contact times.

B. Data Forwarding with Stranger

The aforementioned schemes do not take strangers into account, i.e., most of them take a greedy mechanism to deliver messages, thus, only nodes with a higher quality metric than current carriers can be selected as relays. The data forwarding efficiency, however, can be improved if strangers can participate in the delivery process. This is mainly because the strangers are likely to be more useful for bring messages to different parts of the deployed region, which potentially increases the probability to encounter destinations.

There already exist few works that tries to explore the influence of strangers on system performance. For example, the authors of [8] took low number of contacts and short duration as a baseline to identify strangers and integrate the strangers into their metric by assuming that the mobility of each node/person can be acquired in advance. We argue that it may not be practical since people always are reluctant to expose their daily routines. On the other hand, the centralized community-detection mechanism used in [8] results in high cost due to the information exchange and calculations, and the complexity of adjusting several threshold parameters [11], [12]. Yuan et al. evaluated the influence of strangers on data forwarding performance [7]. They observed that the importance of strangers shows a decreasing trend along the forwarding path, indicating that we should control the number of strangers.

Note that the obvious difference between our work and the aforementioned work comes from the fact that we address a more challenging scenario [21], where each node only depends on itself to locally estimate the forwarding metric to the destination. Furthermore, we explore the influence of the number of strangers on data forwarding performance, rather than pure strangers. Finally, we develop a distributed variant for the computation of the number of strangers that have received the message.

III. Implementing Stranger into STRON

In this section, we discuss how to integrate strangers into STRON. We first introduce the greedy mechanism which has been applied into many data forwarding metrics in Section III.A. In the following two sections, we explore how to identify strangers and ascertain the optimized number of strangers, respectively. In Section III.D, we present our metric. Finally, we have a discussion in Section III.E.

A. Greedy Mechanism

In the past few years, researchers have proposed a large number of data forwarding metrics in MONs. Although they exploited different kinds of contexts (e.g., similarity [5], intra-contact time [2], energy level and virtual community [30] etc.), most of them took a greedy mechanism. That is, when two nodes have a contact, a node with a lower quality metric to the destination will forward messages to the node with a higher quality. For ease of presentation, in this paper, we take intra-contact time as an example to illustrate the main difference between the greedy mechanism and the STRON, other metrics are also welcome.

Let random variable X_i denote the intra-contact time (ICT) between node i and other nodes, let $x_i(d)$ denote the intra-contact time between node i and any node d. Let N denote the set of nodes in the network. The notations used in this paper are listed in Table I.

Table I: Notation Summary

NOTATION	Explanation
N	The set of nodes
$\|N\|$	Number of nodes in the network
i, j	Two randomly chosen nodes
$x_i(j)$	ICT between node i and j
m_d	The destination of message m
$D_s(i,j)$	Degree of strangeness between i and j
P_{ij}	Shortest path from i to j
S_{ij}	The number of strangers in P_{ij}
R_{ij}	Number of relays in P_{ij}
R_s	The mean of S_{ij}/R_{ij}
I_s	Number of strangers which carry m
T_s	The threshold of I_s
f_p	Forwarding probability

We outline the greedy mechanism as follows. Take the node i as an example. When it meets node j, for any message m that i carries, if its destination m_d is node j, node i delivers the message to node j and removes it from i's buffer. Otherwise, if node j does not hold this message, they swap their own quality metric. If $x_i(m_d)$ is smaller

than $x_j(m_d)$, node i forwards m to node j, where i, j and $m_d \in$ N.

B. Identifying Strangers

The authors of [31] firstly explored the issue of how to identify the friendship in MONs. They exploited the number of contacts and the contact duration to cluster nodes, which was also adopted in [8]. A. Miklas et al. classified nodes with the contact frequency [32]. Pairs of nodes which encounter frequently are called as friends, whereas those encountering sporadically are labeled as strangers. The difference between the two methods is that the familiar strangers are excluded from strangers in the former, while they are included in the latter. Recall that we mainly focus on the influence of strangers on system performance, and in fact the familiar stranger still belongs to the stranger, we therefore label nodes as friends or strangers and use the mean of intra-contact time as the threshold to recognize nodes. In fact, more information we used, more precise the classifying is. We here use the intra-contact time as an example to classify nodes, other information used in the related works can be easily integrated into our work. More specifically, let $E(X_i)$ denote the mean of X_i, we have

$$E(X_i) = \frac{\sum_{i,k \in N, i \neq k} x_i(k)}{\|N\|} \tag{1}$$

Hence, whenever node i encounters node j, if $x_i(j)$ is smaller than $E(X_i)$, we call that node j is a stranger to node i. Let $D_s(i, j)$ denote the degree of strangeness between node i and node j, let function f denote the mapping from $x_i(j)$ to $D_s(i, j)$, we have

$$f : x_i(j) \rightarrow D_s(i, j) \tag{2}$$

Obviously, f is a decreasing function, that is, the bigger the value of $x_i(j)$ is, the smaller the value of $D_s(i, j)$ should be. It is difficult and impractical to obtain an optimized f, due to the intermittent connectivity in MONs. Whereas, it is possible to gain some qualitative insights on roles of different functions, we experiment the following three types: convex, linear and concave in general.

$$D_s(i, j) = \sqrt{1 - (x_i(j) / E(X_i))^2} \tag{3}$$

$$D_s(i, j) = 1 - \frac{x_i(j)}{E(X_i)} \tag{4}$$

$$D_s(i, j) = 1 - \sqrt{1 - (1 - x_i(j) / E(X_i))^2} \tag{5}$$

To make the above equations hardness, we set $x_i(j) = E(X_i)$ if $x_i(j) > E(X_i)$. Fig.1 portrays the behavior of $D_s(i, j)$ at different contact times when using Equation (4), where the ID of node i is set to 0 and three other nodes are randomly chosen as partners, besides, CMM [6] denotes the community mobility model, KAIST and NCSU [33]-[34] denote two real traces (please refer to the section Ⅳ). It is clear to see that the experimental results show a close match to the Eq.(4) (i.e., a linearly decreased trend). We analyze the influence of the different decreased functions on STRON performance in the section Ⅳ.B.

C. The Optimized Number of Strangers We Can Employ

To explore the optimized number of strangers we can employ in STRON, we first count the number of strangers which act as relays for other nodes on all shortest paths and then use the mean ratio R_s (i.e., the number of these strangers over that of all relays). Let P_{ij} denote the shortest path from node i to node j, let S_{ij} denote the number of strangers which participate in P_{ij} and R_{ij} denote that of total relays, we have

$$R_s = \sum_{\forall i \in N} \sum_{\forall j \in N, j \neq i} \frac{S_{ij}}{\mathrm{R}_{ij}} \Big/ (N \times (N-1)) \tag{6}$$

According to the Eq.(6), we get the values of R_s are 0.7044, 0.0435 and 0.283 in CMM, KAIST and NCSU, respectively. Interestingly, we discover that the value of R_s in synthetical mobility model is much bigger than those in real traces. It is reasonable since that nodes in the latter show bursty dispersion, instead of randomly selecting one community and going ahead as in the former. Furthermore, the authors of [20] indicated that the "Spray and Wait" scheme can retain good performance under CMM by only assigning a limited number of copies (about 5%~10%) for each message. Considering this fact, the threshold of the number of strangers (T_s) used in this paper is $\|N\| \times 10\% \times 70.44\% = \|N\| \times 7.044\%$ under CMM, $\|N\| \times 4.35\%$ under KAIST and $\|N\| \times 28.3\%$ under NCSU, respectively.

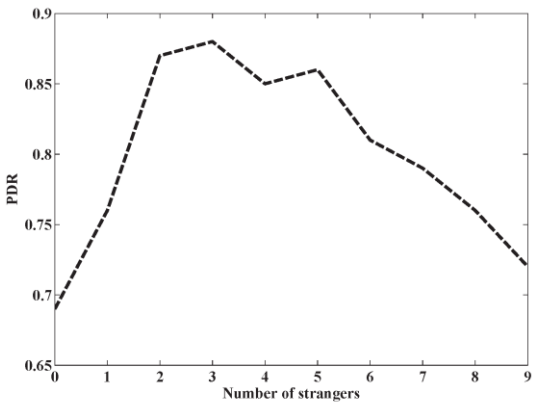


Fig. 2. Influence of the number of strangers on STRON (N=60).

Fig. 2 shows the influence of the number of strangers on STRON in CMM, where we use the concave function to compute the degree of strangeness between nodes. It is clear to see that the number of strangers plays a big role in data forwarding efficiency. The STRON scheme can deliver more packets when the number of strangers ranges from 2 to 5, which has a close match with the aforementioned analysis (Recall that the integer value of $\|N\| \times 7.044\%$ equals to 4, which obviously belongs to the set).

D. STRON Scheme

In this subsection, we discuss how to integrate the strangers and their number into STRON. We develop centralized and distributed variants for the computation of the number strangers that have received the message, since this heuristic number plays a big role in our scheme. Therefore, each node has to be conscious of this number before making a forwarding decision.

Centralized method: We also take two nodes i and j as samples. When node i meets up node j, for any message m in i's buffer, if its destination m_d is node j, node i delivers it to node j and removes the message from its buffer. Otherwise, if node j does not hold this message, node i will make a forwarding decision based on the following two situations:

- If $x_i(m_d)$ is bigger than $x_j(m_d)$ and node j is a stranger to node i and the number of strangers that have received the message is smaller than T_s, the node i forwards m to node j with a probability $f_p = D_s(i, j)\,(x_j(m_d)/(x_i(m_d) + x_j(m_d)))$.
- If $x_i(m_d)$ is smaller than $x_j(m_d)$ and node j is not a stranger to node i, the forwarding probability is $(E(X_i)/x_i(j))\,(x_j(m_d)/(x_i(m_d) + x_j(m_d)))$, otherwise, if the number of strangers that have received the message is smaller than T_s, the forwarding probability is set to 1.

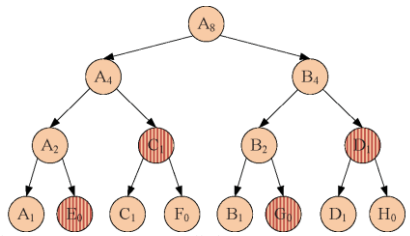
In the aforementioned two situations, whenever a stranger has received the message, the counter I_s increases by one, where I_s denotes the number of infected strangers. We list the above communication process in Algorithm 1.

Algorithm 1 Centralized STRON, pseudo-code of node i

```
upon meeting up node j do
for any message m in i's buffer do
  if m_d == j then
    deliverMsg(m)
    remove(m)
  else if m ∉ j then
    i ← x_j(m_d)
    if x_i(m_d) > x_j(m_d) ∧ x_i(j) < E(X_i) ∧ I_s < T_s then
      m → j with a forwarding probability f_p
      if forwarded then
        I_s ← I_s + 1
      end if
    end if
    if x_i(m_d) < x_j(m_d) then
      if x_i(j) > E(X_i) then
        m → j with a forwarding probability (E(X_i)/x_i(j)) (x_j(m_d)/(x_i(m_d) + x_j(m_d)))
      else if x_i(j) < E(X_i) ∧ I_s < T_s then
        m → j with a forwarding probability 1
        I_s ← I_s + 1
      end if
    end if
  end if
end for
```

Note that the method we presented thus far is suitable for centralized implementations where each relay knows the update of I_s. Clearly, it may not be feasible in the intermittently connected environment [35]-[37]. We next present the distributed version of our method.

Distributed method: For any message m in node i's buffer, let $i_{Ts}(m)$ denote the number of strangers that node i can spray (if node i is a source node, the initial value of $i_{Ts}(m)$ equals to T_s). When meeting up node j, node i uses the binary spray mechanism to update $i_{Ts}(m)$ and assigns half of $i_{Ts}(m)$ to node j (the binary spray mechanism has the optimal spray speed [20]). More specifically, it assigns $\lfloor i_{Ts}(m)/2 \rfloor$ to node j and keeps $\lceil i_{Ts}(m)/2 \rceil$ for itself if node j is an acquaintance to node i. Otherwise, if node j is a stranger, it assigns $\lfloor (i_{Ts}(m) - 1)/2 \rfloor$ to node j and keeps $\lceil (i_{Ts}(m) - 1)/2 \rceil$ for itself. Fig. 3 illustrates this spray process. Suppose the number of strangers that node A can spray equals to 8, when node A encounters node B, it keeps $\lceil 8/2 \rceil$ for itself and assigns $\lfloor 8/2 \rfloor$ to node B. When it encounters node C, it assigns $\lfloor (4 - 1)/2 \rfloor$ to node C and keeps $\lceil (4 - 1)/2 \rceil$ for itself.

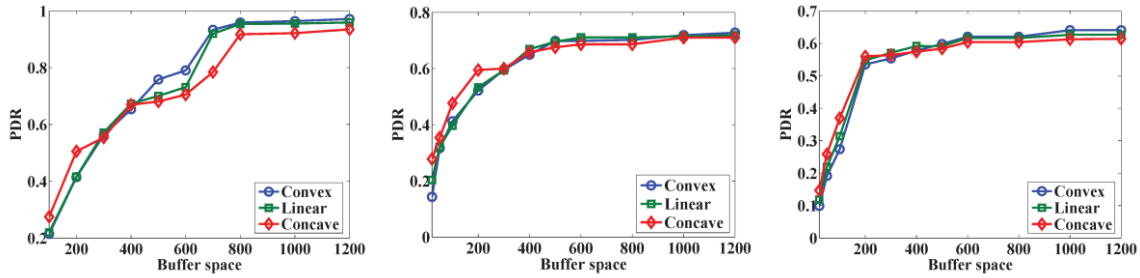


Dark orange → strangers, light orange → acquaintances.

Fig. 3. Binary spray tree.

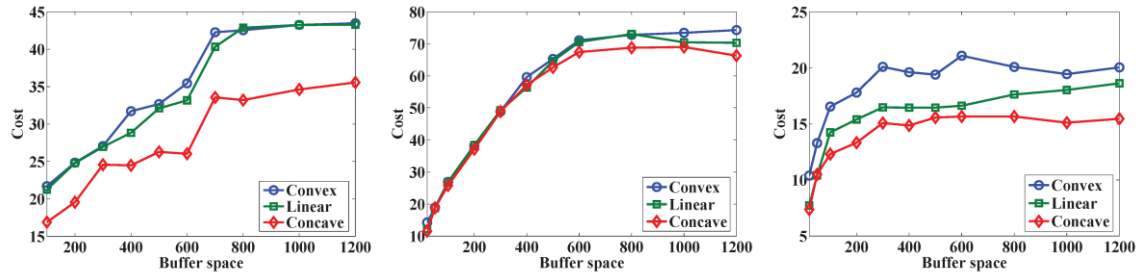
Algorithm 2 Distributed STRON, pseudo-code of node i

```
upon meeting up node j do
for any message m in i's buffer do
  if m_d == j then
    deliverMsg(m)
    remove(m)
  else if m ∉ j then
    i ← x_j(m_d)
    if x_i(m_d) > x_j(m_d) ∧ x_i(j) < E(X_i) ∧ i_{T_s}(m) > 0 then
      m → j with a forwarding probability f_p
      if forwarded then
        i_{T_s}(m) ← ⌈(i_{T_s}(m) − 1)/2⌉
        j_{T_s}(m) ← ⌊(i_{T_s}(m) − 1)/2⌋
      end if
    end if
    if x_i(m_d) < x_j(m_d) then
      if x_i(j) > E(X_i) then
        m → j with a forwarding probability (E(X_i)/x_i(j)) (x_j(m_d)/(x_i(m_d) + x_j(m_d)))
        if forwarded ∧ i_{T_s}(m) > 0 then
          i_{T_s}(m) ← ⌈i_{T_s}(m)/2⌉
          j_{T_s}(m) ← ⌊i_{T_s}(m)/2⌋
        end if
      else if x_i(j) < E(X_i) ∧ i_{T_s}(m) > 0 then
        m → j with a forwarding probability 1
        i_{T_s}(m) ← ⌈(i_{T_s}(m) − 1)/2⌉
        j_{T_s}(m) ← ⌊(i_{T_s}(m) − 1)/2⌋
      end if
    end if
  end if
end for
```



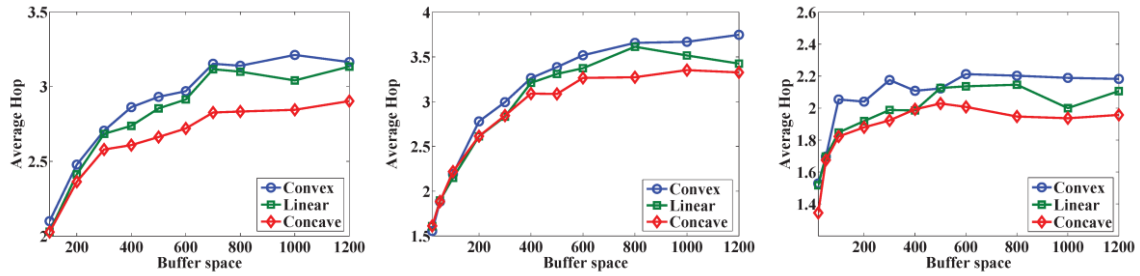
(a) CMM (b) KAIST (c) NCSU

Fig. 4. Packet delivery ratio under three scenarios.



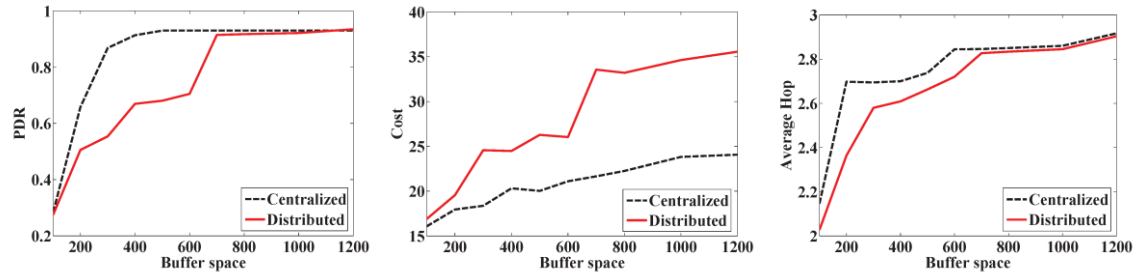
(a) CMM (b) KAIST (c) NCSU

Fig. 5. Cost under three scenarios.



(a) CMM (b) KAIST (c) NCSU

Fig. 6. Average hop per message under three scenarios.



(a) CMM (b) KAIST (c) NCSU

Fig. 7. Centralized vs distributed STRON under the CMM scenario.

We have the Algorithm 2 by taking this adaptive update of T_s into account.

E. Possible Limits and Issues

In this paper, we discuss the influences of the strangers and their number on data forwarding performance in MONs. We do not focus on the correlations among strangers, that is, we only choose k strangers out of ||N|| nodes in the network ($k < T_s$), we do not ascertain which k strangers should be selected. We think it deserves separate study and leave it for future work.

In addition, researchers have proposed lots of metrics to weigh the importance of nodes, we think all of them should be esteemed, whereas, since we mainly focus on the strangers and their optimized number, we here only take intra-contact time as a sample and will evaluate other metrics in future work.

IV. Performance Evaluation and Analysis

A. Mobility Model, Real Trace and System Parameters

In this paper, we use a synthetical mobility model which is called Community Mobility Model (CMM) [6]

and two real trace called KAIST and NCSU to evaluate the performance of greedy mechanism, OP [8] and STRON. The system parameters used in CMM are listed below.

The simulation area is 600m × 600m and is divided into 9 sub-communities. We randomly place 60 mobile nodes at the area. Each node randomly selects one community as its hometown that it is more likely to visit than other communities. The mobility of a node is that it randomly selects a point of a community as its potential destination, moves there, pauses there for a while and selects a new destination. If it is at hometown, it still stays in hometown with a high probability p and visits other communities with a probability 1-p. If it is away from hometown, it will return hometown with a high probability q and other communities with a probability 1-q. The values of p and q are set to 0.8 and 0.9, respectively, two default values also adopted in [6]. The mobility speed is between 10m/s and 30m/s. The pause time is 1s and the communication range is 30m.

In KAIST, 34 volunteers carried the GPS devices (GPS60CSx) from 2006-09-26 to 2007-10-03 and altogether 92 daily traces were gathered. Each individual trace consists of a sequence of three-tuples (Timestamp, X-coordinate, Y-coordinate), which denotes a stay point recorded every 30 seconds.

In NCSU, 20 students from the computer science department were randomly chosen. Every week, 2 or 3 of them carried the GPS receivers for their daily regular activities and 35 traces were gathered.

For the three scenarios, each source sends one message to a randomly chosen destination and altogether 1200 messages are generated. The communication range of nodes is set to 250m in KATST and NCSU, a typical value of WiFi. Besides, since each relay needs to buffer packets for a long period of time in order to cope with the intermittent connections, we compare the three data forwarding strategies in a buffer space constrained system so as to better understand and observe their performance. The simulation results are the average over 20 runs. The evaluation metrics are packet delivery ratio (PDR), average cost and average number of hops per message. The average cost is a correlative factor, which means a message needs to be forwarded how many times before it is received by the destination node.

B. Influence of Different Decreased Functions on STRON Performance

This section focuses on analyzing the influence of different decreased functions on STRON. Fig. 4, Fig. 5 and Fig. 6 show the experimental results. In these figures, we use terms “Convex”, “Linear" and “Concave” to denote the three kinds of decreased functions, respectively. Compared to the convex and linear schemes, the concave scheme achieves a better performance in terms of cost and average hop per message, which still has a competitive performance in packet delivery ratio. We conjecture that this is mainly because of the more refined forwarding probability it used (refer to the section III.B).

C. Centralized vs Distributed STRON

In this section, we show the performance of the centralized and distributed algorithm, respectively. Fig. 7 plots the experimental results. We can see that the centralized scheme achieves upper bounds on the PDR and average hop, and a lower bound on the cost. This difference can be explained by the fact that the centralized scheme does not need to assign the number of strangers. When a carrier encounters a qualified stranger, it sends one copy of the messages to the stranger if the number of infected strangers is smaller than T_s. Whereas, the same situation may not hold true for the distributed version, where the carrier fails to send one copy to the stranger if the residual number of strangers which the carrier can spray equals to one. That is, the distributed version slows down the message diffusion and has a lower PDR and average hop than the centralized one. Simultaneously, the longer delay means that the messages need to be cached for potentially long periods of time, resulting in a heavy cost.

TABLE II: STATISTICS OF CONTACT TIME AMONG STRANGERS/FRIENDS

	CMM	KAIST	NCSU
Strangers(s)	0.62	36.54	128.17
Friends(s)	11.36	326.62	6448.34

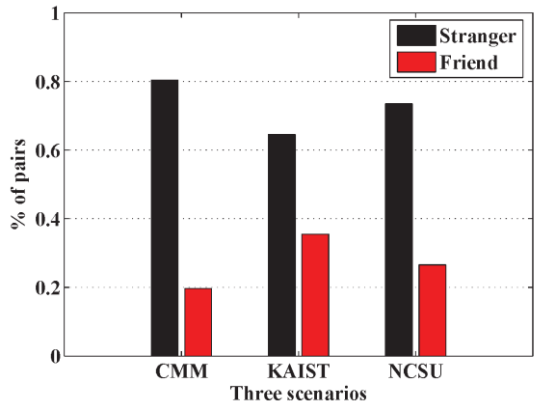
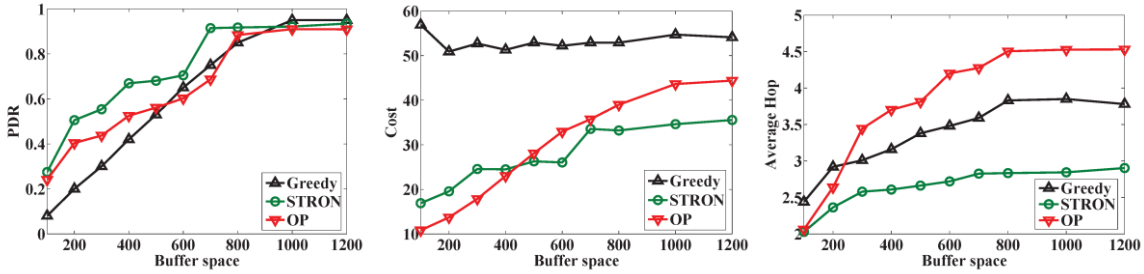


Fig. 8. Percentage of strangers and friends.

D. Accuracy of Classifying Nodes

This section validates the accuracy of the proposed method. As noted in the section III.B, we use the mean of the intra-contact time to classify nodes. Pairs of nodes which have contact duration longer than the mean are classified as friends, and those with shorter contact duration are labeled as strangers. Intuitively, people encounter many strangers and contact a few friends in their daily life. Fig. 8 shows the percentage of the two social relationships. We observe that only 20% of pairs of nodes are friends at CMM, those in two real data sets are still below 40%. This demonstrates that most node pairs are strangers with weak social relation. Thus, if our concern is to collect messages quickly across MONs, we need to focus on strangers since they provide valuable opportunities for different nodes to exchange information. Another intuition is that the contact duration among

strangers is far shorter than those among friends as shown in Table II. We find that the time spending to communicate with friends is nine times longer than those with strangers at KAIST. This demonstrates that friends show a more stable contact behavior. Both the two results validate the accuracy of our method.



(a) PDR (b) Cost (c) Average hop

Fig. 9. Performance metrics of different data forwarding schemes under the CMM.

E. Performance Evaluation

Fig. 9 (a) illustrates the performance of packet delivery ratio when increasing the size of buffer space. It's obvious to see that the buffer size has a heavy effect on the PDR metric. When buffer size is relatively small, STRON shows the best performance. Compared to the greedy mechanism and the OP, it improves the delivery rate by about 35% and 20% when the size of buffer space is smaller than 400. This is mainly because that STRON consumes fewer resources, thus, the impact of buffer size on its performance is very slight. For example, when the buffer size exceeds 700, the PDR performance dominated by STRON is almost free from the buffer size.

Fig. 9 (b) shows the performance of cost. It's interesting to note that increasing the buffer size results in different influences on the cost metric. Generally speaking, STRON first shows an increasing trend and then reaches stable state, rather than the pure increasing trend of OP or the relatively stable state of the greedy scheme. Compared to the greedy and OP scheme, the STRON scheme at least reduces the communication cost by 40% and 22%, respectively, when all of them reach the stable state. From the two algorithms, we know that STRON considers the strangers and their numbers. On the one hand, the several strangers can bring messages into different subareas of the network, which increases the probability to encounter destinations. On the other hand, it delivers messages with different forwarding probabilities according to the degree of strangeness between nodes and the limited number of strangers, which reduces the number of redundant copies, thus, alleviating the overhead of the network.

Fig. 9(c) demonstrates the average number of hops per message. It seems increasing the buffer size also increases the average hops. The reason behind this is that more messages are delivered. These extra delivered messages are those which could be dropped at smaller buffer spaces, but now are able to stay in the buffer space long enough to be delivered to their destinations, which results in a longer hops for those messages. Compared with the greedy and OP scheme, STRON still achieves the best performance. It reduces the average hops by about 45% and 30%, respectively. For instance, at CMM, when the buffer size is bigger than or equal to 800, the average number of hops per message achieved by STRON is near to 2.8 ("Concave"), whereas the other two lead to longer routing paths almost resulting in an average hop value of 3.8 and 4.5, respectively.

V. Conclusion

In this paper, we study the routing problem of MONs. We explore the influence of strangers and their number on system performance. We present an adaptive solution to identify strangers and propose a statistical method to estimate the optimized number of strangers we can employ. Based on this heuristic number, we develop centralized and distributed variants for the computation of this number and integrate them into data forwarding scheme. That is, when two nodes have a contact, we take different forwarding probability based on the current number of infected strangers and the degree of strangeness between them. We finally compare STRON with the state-of-the-art works through synthetical and trace-driven simulations, the results show that our strategy has a better performance, especially in terms of combined overhead/packet delivery ratio and the average number of hops per message.

The significant topics for future work are to study the influence of friends/strangers on the message diffusion process and the different metrics to evaluate node similarity.

References

[1] P. Yuan and P. Liu, "Data Fusion Prolongs the Lifetime of Mobile Sensing Networks," *Journal of Network and Computer Applications*, vol. 49, pp. 51-59, December 2015.

[2] P. Yuan and H. Ma, "Fuzzy Forwarding for Opportunistic Networks," in *Proc. of IEEE International Conference on Communications*, Budapest, 2013, pp. 94-98.

[3] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," CS2200006. Durham, North Carolina : Duke University, 2000, pp. 1-14.

[4] P. Yuan and H. Ma, "Differentiation forwarding with hotspot," I*nternational Journal of Ad Hoc and Ubiquitous Computing*, vol. 15, no. 4, pp. 227-238, May 2014.

[5] Z. Li, W. Cheng, and Y. Siqian, *et al.*, "LASS: Local-activity and social-similarity based data forwarding in mobile social networks," *IEEE Transactions on Parallel & Distributed Systems*, vol. 26, no. 1, pp. 174-184, January 2015.

[6] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *Lecture Notes in Computer Science*, vol. 3126, pp. 239-254, August 2004.

[7] P. Yuan, H. Ma, and P. Duan, "Impact of strangers on opportunistic routing performance," *Journal of Computer Science and Technology*, vol. 28, no. 3, pp. 574-582, May 2013.

[8] S. Gaito, E. Pagani, and G. P. Rossi, "Strangers help friends to communicate in opportunistic networks," *Computer Networks*, vol. 55, no. 2, pp. 374-385, October 2011.

[9] F. Li, L. Zhao, and C. Zhang, *et al.*, "Routing with multi-level cross-community social groups in mobile opportunistic networks," *Personal & Ubiquitous Computing*, vol. 18, no. 2, pp. 385-396, February 2014

[10] P. Sermpezis and T. Spyropoulos, "Understanding the effects of social selfishness on the performance of heterogeneous opportunistic networks," *Computer Communications*, vol. 48, no. 14, pp. 71-83, July 2014

[11] M. M. D. Khomami, A. Rezvanian, and M. R. Meybodi, "Distributed learning automata-based algorithm for community detection in complex networks," *International Journal of Modern Physics B*, vol. 30, no. 8, March 2016.

[12] T. Alzahrani and K. J. Horadam, "Community detection in bipartite networks: Algorithms and case studies," *Understanding Complex Systems*, vol. 73, pp. 25-50, August 2016.

[13] R. C. Shah, S. Roy, and S. Jain, "Data MULEs: Modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 215-233, November 2003.

[14] Q. Li and D. Rus, "Sending messages to mobile users in disconnected ad-hoc wireless networks," in *Proc. 6th Annual International Conference on Mobile Computing and Networking*, Boston, Massachusetts, 2000, pp. 44-55.

[15] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proc. 5th ACM International Symposium on Mobile ad Hoc Networking and Computing*, Tokyo, Japan, 2004, pp. 187- 198.

[16] W. Zhao, M. Ammar, and E. Zegura, "Controlling the mobility of multiple data transport ferries in a delay tolerant network," in Proc. *25th IEEE International Conference on Computer Communications*, Miami, USA, 2005, pp. 1407-1418.

[17] T. He, K. Lee, and A. Swami, "Flying in the dark: Controlling autonomous data ferries with partial observations," in *Proc. Eleventh ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Chicago, Illinois, USA, 2010, pp. 141-150.

[18] P. Yuan and H. Ma, "Hug: Human gathering point based routing for opportunistic networks," in *Proc. IEEE Wireless Communications and Networking Conference*, Paris, France, 2012, pp. 3024-3029.

[19] J. Wu, M. Xiao, and L. Huang, "Homing spread: Community home-based multi-copy routing in mobile social networks," in *Proc. IEEE International Conference on Computer Communications*, Turin, Italy, 2013, pp. 2319-2327.

[20] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: The multiple-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77-90, February 2008.

[21] P. Yuan, H. Ma, and P. Duan, "On exploiting few strangers for data forwarding in delay tolerant networks," in *Proc. IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks*, Beijing, 2011, pp. 282-288.

[22] R. Qin, Z. Li, and Y. Wang, "An integrated network of roadside sensors and vehicles for driving safety: Concept, design and experiments," in *Proc. IEEE International Conference on Pervasive Computing and Communications*, Mannheim, 2010, pp. 79-87.

[23] W. Ivancic, W. M. Eddy, and D. Stewart, "Experience with delay-tolerant networking from orbit," in *Proc. IEEE 4th Advanced Satellite Mobile Systems*, Bologna, 2008, pp. 173-178.

[24] S. Jain, K. Fall, and S. Patra, "Routing in a delay tolerant network," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2004, pp. 145-158.

[25] C. Liu and J. Wu, "Practical routing in a cyclic MobiSpace," *IEEE/ACMTransactions on Networkin*g, vol. 19, no. 2, pp. 369-382, April 2011.

[26] Q. Yuan, I. Cardei, and J. Wu, "An efficient prediction-based routing in disruption-tolerant networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 19-31, January 2012.

[27] M. Musolesi and C. Mascolo, "CAR: Context-Aware adaptive routing for delay-tolerant mobile networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 2, pp. 246-260, February 2009.

[28] J. Leguay, T. Friedman, and V. Conan, "Evaluating mobility pattern space routing for DTNs," in *Proc. 25th IEEE International Conference on Computer Communications*, Barcelona, Spain, 2006, pp. 1-10.

[29] P. Yuan, L. Fan, P. Liu, and S. Tang, "Recent progress in routing protocols of mobile opportunistic networks: A clear taxonomy, analysis and evaluation," *Journal of Network and Computer Applications*, vol. 62, pp. 163-170, February 2016.

[30] H. Pan, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: Social based forwarding in delay tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576-1589, November 2011.

[31] E. Yoneki, P. Hui, and J. Crowcroft, "Visualizing community detection in opportunistic networks," in *Proc. Second ACM Workshop on Challenged Networks*, New York, NY, USA, 2007, pp. 93-96.

[32] A. Miklas, K. Gollu, and K. Chan, "Exploiting social interactions in mobile systems," in *Proc. ACM 9th International Conference on Ubiquitous Computing, Innsbruck*, Austria, 2007, pp. 409-428.

[33] I. Rhee, M. Shin, and S. Hong, *et al.*, "On the levy-walk nature of human mobility," *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 630-643, June 2008.

[34] P. Yuan and C. Wang, "OPPO: An optimal copy allocation scheme in mobile opportunistic networks," Peer-to-Peer Networking and Applications, pp. 1-8, June 2016.

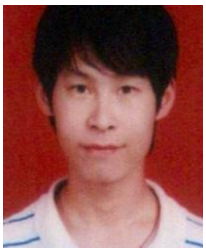
[35] J. Hu, L. L. Yang, L. Hanzo, and K. Yang, "Socially-Aware integrated centralized infrastructure and opportunistic networking: A powerful engine stimulating the content dissemination," *IEEE Communication Magazine*, vol. 56, no. 8, August 2016.

[36] J. Hu, L. L. Yang, and L. Hanzo, "Delay analysis of social group multicast aided content dissemination in cellular system," *IEEE Transactions on Communication*, vol. 64, no. 4, April 2015.

[37] J. Hu, L. L. Yang, and L. Hanzo, "Bridging the social and wireless networking divide: information dissemination in integrated cellular and opportunistic networks," *IEEE Access*, vol. 3, October 2015.

Ping Liu received her B.S. degree in computer science from Henan Normal University, China, and M.S. degree from China University of Mining and Technology. She is currently a Ph.D. candidate in computer science at the Wuhan University of Technology. Her research interests are in the areas of Network architecture and Network services

Peiyan Yuan is an associate professor of Computer Science at the Henan Normal University since 2014, where he received his B.S. degree at Computer Science in 2001. After that, he got his M.S. (2007) and Ph.D. (2014) degree at Computer Science from Wuhan University of Technology and Beijing University of Posts and Telecommunications, respectively. His research interests include future networks and distributed systems ranging from content-centric networks, mobile opportunistic networks to online social networks and crowd sensing applications etc. He is a senior member of the CCF and a member of the ACM.



Chenyang Wang received his B. Eng. Degree in Computer Science and Technology from the Henan Normal University, Xinxiang, Henan province, China. He is now a postgraduate student in the School of Computer and Information Engineering at the Henan Normal University. His research interests include mobile opportunistic networks and D2D wireless networks.