

Research Article

A Data Transmission Scheme Based on Time-Evolving Meeting Probability for Opportunistic Social Network

Fu Xiao,^{1,2,3} Guoxia Sun,¹ Jia Xu,^{1,2,3} Lingyun Jiang,^{1,2,3} and Ruchuan Wang^{1,2,3}

¹ College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu 210003, China

² Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, Jiangsu 210003, China

³ Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications), Ministry of Education Jiangsu Province, Nanjing, Jiangsu 210003, China

Correspondence should be addressed to Fu Xiao; xiaof@njupt.edu.cn

Received 6 May 2013; Accepted 17 June 2013

Academic Editor: Shukui Zhang

Copyright © 2013 Fu Xiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With its widespread application prospects, opportunistic social network attracts more and more attention. Efficient data transmission strategy is one of the most important issues to ensure its applications. As is well known, most of nodes in opportunistic social network are human-carried devices, so encounters between nodes are predictable when considering the law of human activities. To the best of our knowledge, existing data transmission solutions are less accurate in the prediction of node encounters due to their lack of consideration of the dynamism of users' behavior. To address this problem, a novel data transmission solution, based on time-evolving meeting probability for opportunistic social network, called TEMP is introduced, and corresponding copy management strategy is given to reduce the message redundancy. Simulation results based on real human traces show that TEMP achieves a good compromise in terms of delivery probability and overhead ratio.

1. Introduction

Driven by the emergence and application of large number of mobile devices, which are characterized with low-cost, powerful, and short-range communication capabilities, wireless ad hoc network has acquired rapid development. With further research, people started to pay attention to the mobile ad hoc networks, especially those whose communication equipments are deployed on the moving object, such as wildlife tracking network [1], vehicle network [2], and pocket switched network [3]. Traditional communication mode is no longer applicable in these practical application scenarios, due to the regular disruption caused by the sparse deployment, quick movement, and strict constraint both on storage and energy of nodes. Opportunistic network [4], which achieves data transmission via node mobility, appeared in such a situation. As a more natural ad hoc network style, opportunistic network transfers messages through a storage-carry-forward hop-by-hop strategy.

To achieve reliable data transmission for opportunistic network, multicopy technique is usually adopted when

the real-time path cannot be guaranteed, for instance, EPIDEMIC [5] and PROPHET [6]; in other words, there are multiple copies of the same message in the network. This will cause data redundancy which affects the network performance. Therefore, efficient data transmission strategy requires effective copy management strategy. In a typical opportunistic network, nodes move randomly and quickly; nevertheless, for opportunistic social network, the mobility of nodes is controlled by human social activities, and the encounters between nodes are more stable and regular, so we can use the history activities of nodes to predict the encounter of nodes in the future in opportunistic network as paper [7] elaborated.

Combined with the characteristic of the opportunistic network as well as the need of copy management, a data transmission based on node time-evolving meet probability which consists of message forwarding and management is proposed in this paper. Message forwarding is divided into three steps: find the periodic neighbor of the destination node, find the appropriate time slot for forwarding, and forward the message to the node which has higher meet probability

with the destination in the right slot. There are two cases for the message management: multicopy strategy is adopted to establish quick contact with the destination in the first step of message forwarding, and for the last two steps message is forwarded in a single copy way to reduce network overhead.

The rest of this paper is structured as follows. Section 1 briefly analyses the related work and our work is elaborated in Section 2. Section 3 presents the simulation and the evaluation and future work of this paper is presented in Section 4.

2. Related Works

A variety of data transmission strategies for opportunistic social network communication is proposed. LABEL [8] proposed by Hui and Crowcroft is the earliest work; the authors think that nodes that belong to the same community have higher encounter opportunity, and they assume that each node has a label to identify their communities; the message is forwarded to the destination directly or by the relay nodes that belong to the same community with the destination. It is inefficient unless the source node can be met directly with the node that is in the same community with destination node. On the basis of LABEL, a community-based data transmission strategy, called Bubble Rap, is proposed in paper [3]. Bubble Rap relied on community and centrality; each node has a local centrality that describes the popularity of the node with its local community and a global centrality across the whole network; it first bubbles the message up based on the global centrality, until the message reaches a node which is in the same local community as destination. The related community detection algorithms, SIMPLE, k -CLIQUE, and MODULARITY are given in paper [9]. Node similarity is defined in [10] to describe the neighborhood relationship between nodes, according to the history of node encounter. Based on the neighborhood relationship, a distributed community detection algorithm is given, as well as a community-based epidemic forwarding. Based on the literature [10], a new social pressure metric (SPM) is introduced in [11] to accurately detect the quality of friendship; this approach considers both direct friendship and indirect friendship to construct its community. It can help to make smarter decisions. Nevertheless, the calculation of metric needs the whole contact information which may be unrealistic for opportunistic network.

All the above work is community-based data transmission strategy. It generates considerable network traffic for community information maintenance overhead. Moreover, most community detection algorithm may lead to the formation of monotonically increasing cluster due to lack of time information; that is, more and more nodes are added to the community with time elapsed, but the outdated nodes cannot be removed from the community timely. On the other hand, community detection takes a long time and brings a “slow start” problem to the network. In summary, community-based data transmission strategy makes good use of community feature of opportunistic social network, but it requires

an efficient community detecting algorithm to improve the performance.

At the same time, researchers also proposed a series of data transmission strategy based on node meeting opportunity predicting. PROPHET is a kind of multicopy transmission strategy that can be applied to the opportunistic social network. In the protocol, each node maintains its own transmission probability to the destination and message is forwarded to the node which has a greater meet probability with destination when two nodes meet. There is a “lag” problem when forecasting the encounter probability; furthermore, the excessive copy of the message causes larger overhead. A novel strategy based on node sociability is given in [12]; the key idea is that of assigning to each node a time-varying scalar parameter which captures its social behavior in terms of frequency and types of encounters, and then node forwards message only to the most social nodes. In [13] Mei et al. found that people with similar interests tend to meet more often and then proposed SANE, a social-aware and stateless routing for opportunistic social network; the interest profile of an individual is represented as a k -dimensional vector. The cosine similarity is defined to express the interest similarity between two nodes; a message should be forward to nodes whose interest similar to destination. PeopleRank [14] ranks the node according to the node importance using a similar algorithm as PageRank; node forwards message to destination or a more important node. The author of dLife [15] believes that opportunistic social network should consider the dynamism of users’ behavior resulting from their daily routines; each node has two functions: TECD that captures the evolution of social interaction among pairs of users in the same daily period of time over consecutive days; and TECD $_i$ that captures the node’s importance. The message is forwarded to the encounter if its TECD to destination is bigger than that of the carrier, or its TECD $_i$ is higher than that of the carrier when the relationship to destination is unknown. And the literature [16] explored how much delay has to be tolerated for the message delivery from the source to the destination.

3. The TEMP Strategy

3.1. Problem Description. In a typical opportunistic network, node mobiles follow the same pattern (such as random model), so nodes are very similar in terms of data transfer. However, nodes in opportunistic network are mainly human-carried devices whose mobility is controlled by people; therefore, node mobility is distinct, but their encounter is more regular and stable. As described in [15], nodes have different encounter relationship with different nodes at different time periods of the same cycle because of its daily routines. As shown in Figure 1, A has 5 directly meeting nodes, B, C, D, E, and F, if two nodes meet each other in the related time slot, there is a line between them, and the number of encounters is represented by value w on the line. In summary, nodes have similar encounter relation within the same time slot in different cycles but different encounter relation at different time frames of the same cycle; these features should be considered when designing data transmission strategy.

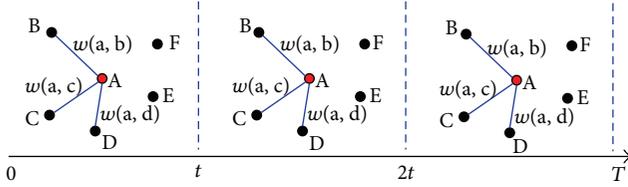


FIGURE 1: The contact information of A at different time slots within the same cycle.

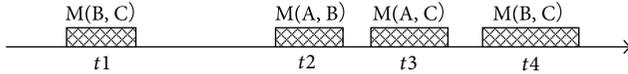


FIGURE 2: A common scenario of node contact for illustration of the lag problem.

Next, we will first analyze the deficiencies of the existing typical solutions and then present our solution, TEMP forwarding strategy.

3.2. Problem Analysis of Existing Solutions. PROPHET may be first thought of when we try to reflect node time-varying meet probability; in PROPHET the update method mainly includes the increase and decay in the probability. The probability increases according to formula (1) when A meets B and decays in time as formula (2):

$$P_{(A,B)} = P_{(A,B)\text{old}} + (1 - P_{(A,B)\text{old}}) \times P_{\text{init}}, \quad (1)$$

$$P_{(A,B)} = P_{(A,B)\text{old}} \times \gamma^k. \quad (2)$$

There is a reaction lag problem in this update method for the prediction of node meeting probability. Consider a common situation shown in Figure 2, at time t_3 ; for node B and C, their meet probability decays to a small value after $t_3 - t_1$, but the probability of A and B increases recently because they meet a while back, then there is $P(A, B) > P(C, B)$ according to PROPHET. Therefore when node A and C encounter at t_3 , C forwards the message, whose destination is B, to A. However we find that it is node C meets node B instead of A in the near future. The message is forwarded toward an incorrect direction just because of the lag problem.

To describe the dynamic probability of contact between nodes, dLife uses TECD and TECD i , two time-varying parameters. First it acquired the average contact length of nodes for each time slot; then for a certain time slot, the meeting probability of two nodes is calculated by the weighted average contact length, with each weighed by a certain coefficient. The main problem of dLife is shown in Figure 3; according to dLife we have $\text{TECD}(A, D) > \text{TECD}(A, C)$ when C meets D in the first sample slot, so if C has message destination for A, dLife will forward the message to D; however, later we will find that C has more opportunity to meet A in the rest of time slot, so message should be still carried by C. For opportunistic social network, the contact number of nodes is limited, and the more often nodes met before, the less chance they will meet in the future for a certain time slot. Note that, this is different with different cycles; it

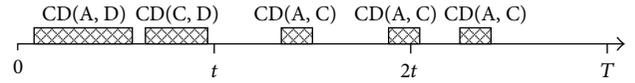


FIGURE 3: Sketch for illustrating the problem of dLife.

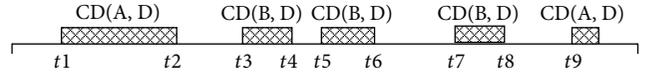


FIGURE 4: Sketch for elaborating node's contact information.

is universally thought that nodes have chance to meet each other, if they met often in the previous cycles.

3.3. Our Solutions. The research results in [7] show that nodes encounter each other with periodic regularity, and node's contact information of last cycle can be used to predict node's meet probability for the next cycle. At the same time dLife pointed out that nodes have different contact relationships for different time slots within the same cycle. Based on the above facts, we make the following assumptions: nodes have varying encounter relationships during different time slots of the same cycle, but for the same time slot in different cycles, node's encounter is relatively stable.

3.3.1. Node's Encounter Probability at Time t within Slot i . First, if the encounter interval of any two nodes is available, the time length between t and the time that node meets destination is acquired; the shorter the length is, the more suitable the node is chosen as the relay node, so we can use formula (3) to represent the meeting probability:

$$P(t)_{(A,B)m} \propto \frac{1}{(d_{(A,B)m} - (t - t_{(A,B)m-1}))} \quad (3)$$

t is the current time, $t_{(A,B)m-1}$ represents the number $m - 1$ meeting time of A and B, and the time interval between number $m - 1$ and number m meeting time of node A and B is $d_{(A,B)m}$. But it is unrealistic to predict the exact time that any two nodes meet at an opportunistic network, even if the information can be obtained; it needs high storage overhead to store this information. Next we will give an approximate solution which is computational and has low storage cost.

We can obtain the contact duration of any two nodes in an arbitrary time slot according to the history information. For a specific time slot i , the contact duration of A and B is relatively stable, which is used to estimate the meet probability by dLife. To make a better prediction, the number and sequence of node's encounter also should be considered. For example, as shown in Figure 4, the shadow rectangles represent the contact duration, the total contact duration of A and B is $\text{TCD}(A, B)$, and then we have $\text{TCD}(A, D) > \text{TCD}(B, D)$. However, for the probability of node's encounter, we should have $P(A, D) > P(B, D)$ before time t_1 , $P(B, D) > P(A, D)$ during t_2 to t_3 and $P(A, D) > P(B, D)$ between t_8 and t_9 .

Based on the above analysis, the method of calculation of node's encounter probability at any time t within the slot i is given as formula (4)

$$P_i(t)_{(A,B)} \propto (\text{TCD}(A, B)_i - \text{CD}(t)_{(A,B)}) \quad (4)$$

$\text{CD}(t)_{(A,B)}$ is the contact duration of A and B before time t within slot i , and their total contact duration in slot i is $\text{TCD}(A, B)_i$; take the deviation into consideration; if $\text{CD}(t)_{(A,B)} > \text{TCD}(A, B)_i$, $P_i(t)_{(A,B)}$ is set 0. $\text{TCD}(A, B)_i$ is calculated based on the average contact duration in slot i of history cycles, as formula (5), j is the number of historical cycles, and $\text{CD}(A, B)$ represents the contact duration of A and B in slot i of cycle k as follows:

$$\text{TCD}(A, B)_i = \frac{\sum_{k=1}^j \text{CD}(A, B)_{k,i}}{\sum_{k=1}^j k} \quad (5)$$

3.3.2. Message Forwarding Strategy. TEMP divides a node's active cycle into multiple sampling slots according to node's day-to-day itinerary form. Each node maintains its own contact information of each slot. If two nodes can meet directly, they are neighbor node for each other. A's neighbor node set in slot i is $N_i(A)$, and the total neighbor of A in the entire cycle is $N(A) = \cup N_i(A)$. TEMP forwarding strategy mainly comprises three stages: (1) find the periodic neighbor of the destination node; (2) find the appropriate time slot for forwarding; (3) forward the message to the node which has higher meet probability with the destination in the right slot. Next we will explain the three stages in detail.

(1) Find the periodic neighbor of the destination node.

If the destination node of message does not belong to the neighbor set of the message carrier, the message needs to be forwarded to nodes which have neighbor relationship with the destination as soon as possible. That is, the message carrier node A meets with B, if B satisfies the condition:

$$\frac{(|N(B) \setminus N(A)|)}{|N(B)|} \geq \lambda. \quad (6)$$

A forwards the message to B, λ is an adjustable parameter, and it is set to 1 in order to reduce the message redundancy in this paper). Messages are forwarded by this way until they reach a node that has neighbor relationship with the destination. Then the message forwarding moves into the second stage.

(2) Find the appropriate time slot for forwarding.

The main purpose of this stage is to find the most recent time slot that the message can be delivered. We define the distance between slot i and j as $\text{dis}(i, j)$, which is calculated according to formula (7) as follows:

$$\text{dis}(i, j) = \begin{cases} j - i & j \geq i \\ j + S - i & j < i \end{cases} \quad (7)$$

S is the number of the time slot in a cycle. Note that, $\text{dis}(i, j)$ is different from $\text{dis}(j, i)$. Assume the destination of message is D. After the previous stage, the message's current carrier A must have neighbor relationship with D; that is, D belongs to

A's neighbor collection of a certain time slot i , $D \in N_i(A)$. when A meets with B at slot k , if there exist $D \in N_i(B)$ and $\text{dis}(k, i) > \text{dis}(k, j)$, A forwards the message to B. And so it goes on, until the most recent delivery time slot arrives. Then message forwarding goes to step 3.

(3) Forward the message to the node which has higher meet probability with the destination in the right slot.

Message forwarding reaches this stage; it means that the message current carrier can meet with the message's destination D at the current slot i . So if the message's carrier A meets B at time t , if the condition $P_i(t)_{(B,D)} > P_i(t)_{(A,D)}$ is met, A forwards the message to B.

3.3.3. Copy Management. Copy management aims to reduce the message redundancy and the network load. Copy management includes the following two aspects.

- (1) In the message diffusion stage, namely, the first step of the forwarding policy, the destination's information is unavailable. In this condition, the message is forwarded based on multicopy strategy; that is, the message carrier still save a copy of the message after it forwards the message to another node. The goal here is to establish contact with destination as soon as possible by the multicopy strategy.
- (2) In the last two steps of message forwarding, message carrier has established contact with the destination, so the message is forwarded in a single copy way. The carrier deletes the message once it forwards the message to an appropriate node to reduce the redundancy.

4. Simulation and Evaluation

4.1. Simulation Settings. We develop the TEMP on the DTN simulation platform, ONE1.4.1 [17], developed by the Helsinki University, and also give the performance evolution based on the simulation results. The experimental scenario is based on real human movement trajectory data sets, Cambridge Traces [18]. In order to evaluate the performance of each algorithm, we generate 5000 messages in advance using createCreates.pl, a Perl script of ONE. The source and destination of message are randomly selected, and the message size is evenly distributed between 10 KB and 100 KB. Table 1 shows the main parameters of the simulation.

To evaluate the performance of TEMP, we will compare it to dLife and PROPHET in terms of delivery rate, overhead ratio, and network delay and discuss the results.

4.2. Effect of TTL. In this experiment message buffer size is set to 2 MB. In Figure 5, we can see that the delivery ratio of each protocol develops with the increasing of TTL. dLife and PROPHET achieve high delivery ratio when TTL is less than 1 day, but once the TTL is greater than 1 day, the delivery ratio of TEMP improves significantly and eventually much higher than the other two protocols. In the implementation of TEMP, node's activity cycle is set to 1 day, so when the message TTL is less than 1 day, the performance of TEMP is poor. At the same time, we note that while node's activity cycle in dLife is

TABLE 1: Default parameter value.

Parameter	Value
Simulation time	990000 s
Update interval	60 s
Number of messages	5000
Message size	10 kB–100 kB
Message TTL	2 day
Buffer size	2 MB
Transmit speed	11 Mbps
ProphetRouter.secondsInTimeUnit	10 s
ProphetRouter.p_init	0.75
ProphetRouter.beta	0.25
ProphetRouter.gamma	0.98
Dlife.numberofslot	24
λ	1

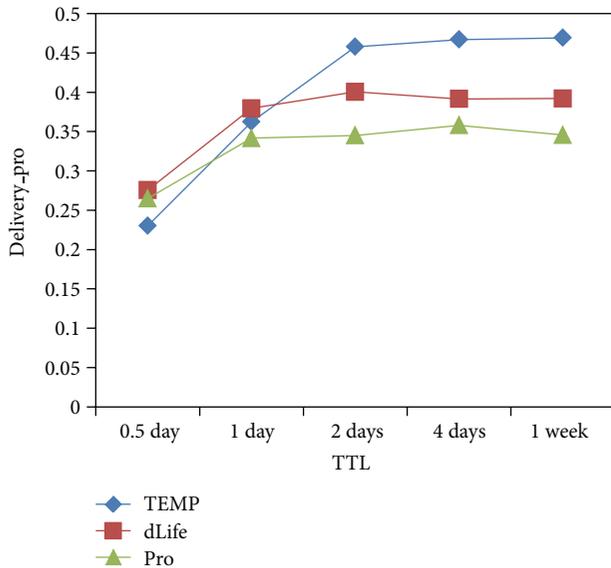


FIGURE 5: Delivery ratios under varying TTL.

also set to 1 day, it achieves better performance than TEMP when message TTL is less than 1 day. The reason is that dLife uses the weighted sum of node’s contact duration of every slot to calculate $TECD$ and $TECD_i$, and the coefficient of each slot is very close to each other which makes dLife more likely to use the total contact duration to predict node’s encounter probability. In a word, it makes little sense to divide the slot for dLife.

The definition of overhead ratio in Figure 6 adopts ONE’s default value; that is, $overhead_ratio = (relayed - delivered) / delivered$. Figure 6 shows that compared with ROPHET TEMP and dLife can reduce the overhead ratio significantly. When message TTL is less than about 1 day, the message forwarding mainly uses multicopy strategy for TEMP, so the overhead ratio of TEMP is much higher than dLife, but it decreases with the addition of message TTL and remains stable finally.

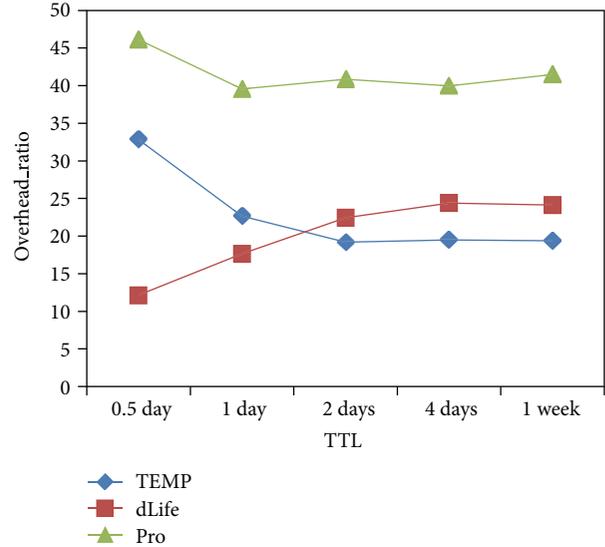


FIGURE 6: Overhead ratios under varying TTL.

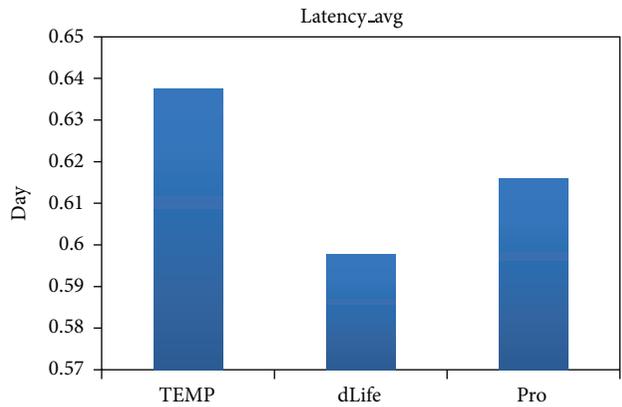


FIGURE 7: Average network delays.

We also observed the delay of each protocol when the TTL changes. In the experiments, we found that when the TTL increases, the delay of each protocol changes slightly. So we use the average of network delay in different TTL settings for evaluation and plot them in Figure 7. We find that even though the average delay of TEMP is higher than that of dLife and PROPHET, the difference among them is less than 0.04 day (about an hour) which is acceptable in opportunistic network.

4.3. *Effect of Buffer Size.* We studied the effect of buffer size on the performance in this part; TTL value is set to 2 days. The results of the experiment are showed in Figures 8–13.

The plot shows that TEMP outperforms all the other forwarding schemes on delivery ratio. In normal conditions, the usage of multicopy strategy can improve message delivery ratio. However, network resources, such as storage, energy, are limited in opportunistic social network; a large number of message copies bring huge resource consumption which will reduce network performance and bring high overhead

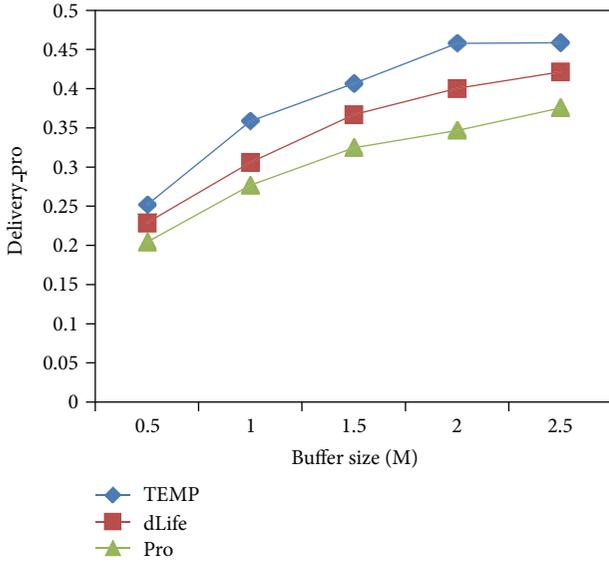


FIGURE 8: Delivery ratios under varying buffer sizes.

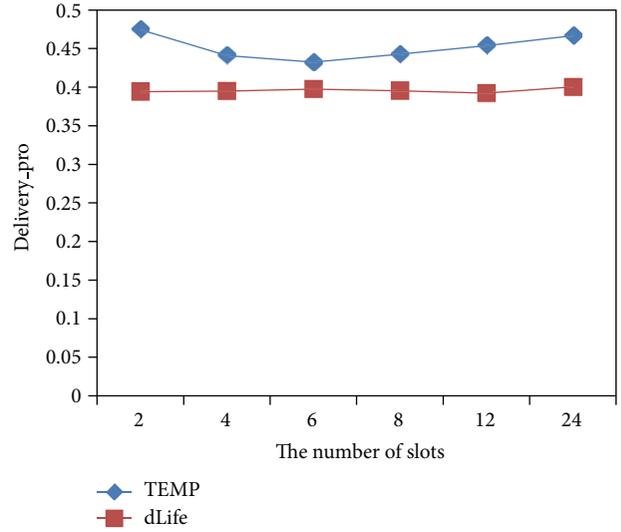


FIGURE 11: Delivery ratios under varying number of slots.

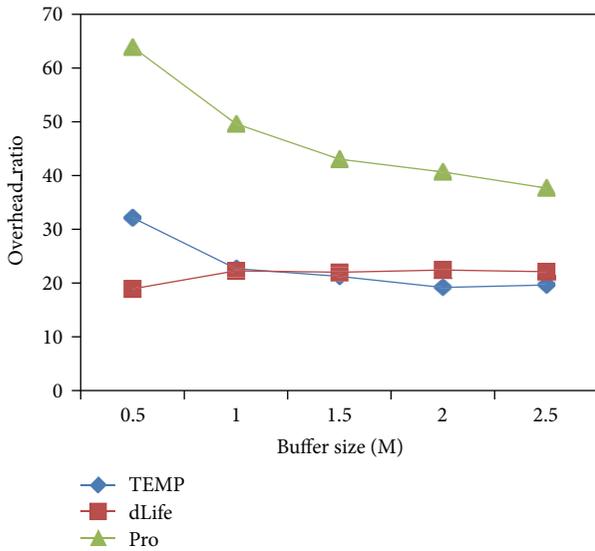


FIGURE 9: Overhead ratios under varying buffer sizes.

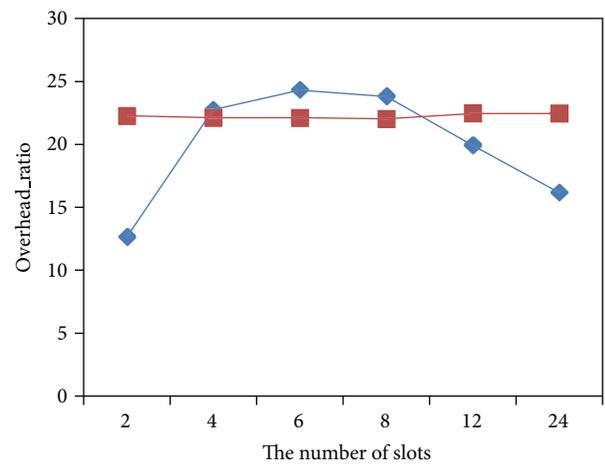


FIGURE 12: Overhead ratios under varying number of slots.

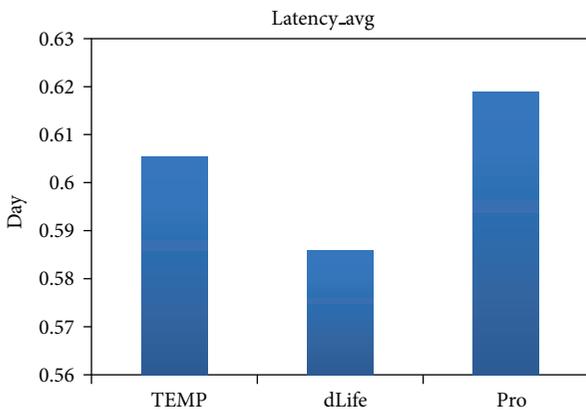


FIGURE 10: Average network delays.

ratio as is shown in Figure 9; therefore, the delivery ratio of dLife and PROPHET is poor. Moreover, the “lag problem” of PROPHET further decreases the delivery ratio.

Figure 10 shows the average delay of the three forwarding schemes under different buffer sizes. dLife takes node’s global importance into consideration and uses TECDi (similar to PageRank algorithm) to make message forwarding decision, so it achieves the best performance on delay. Note again, the calculation of a node’s importance metric TECDi relied on the prior obtaining of its neighbor’s importance which is very complex. TEMP only uses node’s local information and avoids this disadvantage. Overall, the difference of delay, less than 0.03 day (about 0.72 hour), among them is very slightly for opportunistic social network with delay tolerance capacity.

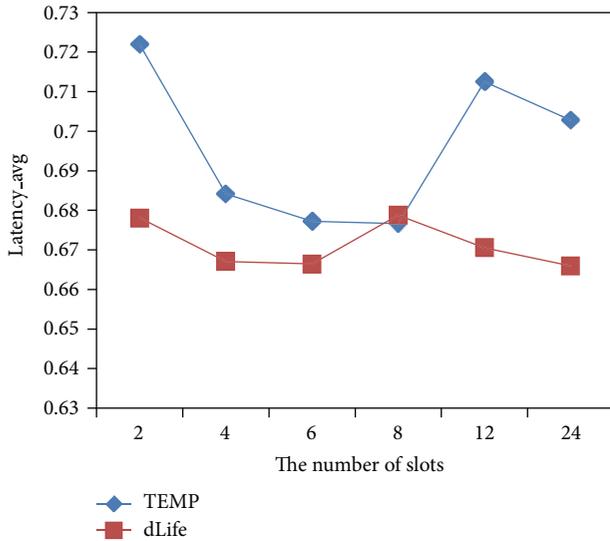


FIGURE 13: Average delays under varying number of slots.

4.4. Analysis of Time Slot Division. In paper [15], the author pointed out that dLife achieved the best performance when the cycle was divided into 24 time slots. In all experiments above, we used 24 time slots in the implementation of dLife. But for TEMP, we adopt 4 time slots, it should be noted that this division is not optimal for TEMP, and we adopted 4 time slots because we take the two main periods, namely, 6:00 am–12:00am and 12:00 pm–18:00 pm, of human activities into consideration. In fact, the division of time cycle should be based on a large number of observations, but in this paper we divided the time cycle into multiple slots of equal length for convenience. In the following parts, we talk about the effect of time slot division on TEMP and dLife.

The curves in Figure 11 show that the delivery ratio performance of TEMP is always higher than that of dLife under different time cycle divisions. From all the three figures above, that is, Figures 11, 12, and 13, we can find that the performance of dLife changes slightly. This further proves that it makes little sense for dLife to divide time slot although its original intention is to reflect node's different encounter relationship by dividing time slot. Compared to dLife, TEMP is more sensitive to slot division which makes it more suitable for opportunistic social network that node's activities are obviously different during different time slots.

5. Conclusion and Future Work

In this paper, we propose a data transmission strategy, called TEMP, for opportunistic social network. It consists of message forwarding and copy management strategy. Simulation results show that TEMP is more efficient in terms of delivery ratio and overhead ratio. Simultaneously it is more suitable for the scene in which node's activities show a significant difference during different time slots. It is well known that the community is a very important feature of opportunistic social network, but the existing community detection algorithm is nonadaptive and complex. Thus, in the future work, we aim

to study a computable and self-adaptive community detection algorithm to assist the message forwarding. Such as, we can use GPS or RFID to locate the community that the node is currently in; of course, the community is a geo-community, but for offline network, node's mobility preference is always associated with the geographic information.

Acknowledgments

This work is sponsored by National Natural Science Foundation of China (61003236, 61170065, 61100199), Scientific & Technological Support Project of Jiangsu (BE2012755), Scientific Research & Industry Promotion Project for Higher Education Institutions (JHB2012-7), and a Project funded by Priority Academic Program Development of Jiangsu Higher Education Institutions (Information and Communication, YX002001).

References

- [1] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet," in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 96–107, October 2002.
- [2] U. G. Acer, P. Giaccone, D. Hay, G. Neglia, and S. Tarapiah, "Timely data delivery in a realistic bus network," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1251–1265, 2012.
- [3] P. Hui, 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, 2011.
- [4] Y.-P. Xiong, L.-M. Sun, J.-W. Niu, and Y. Liu, "Opportunistic networks," *Journal of Software*, vol. 20, no. 1, pp. 124–137, 2009.
- [5] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep. CS-2000-06, Duke University, 2000.
- [6] L. Anders and D. Avri, "Probabilistic routing in intermittently connected networks," *ACM Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.
- [7] N. Eagle and A. S. Pentland, "Eigenbehaviors: identifying structure in routine," *Behavioral Ecology and Sociobiology*, vol. 63, no. 7, pp. 1057–1066, 2009.
- [8] P. Hui and J. Crowcroft, "How small labels create big improvements," in *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 65–70, Washington, DC, USA, March 2007.
- [9] P. Hui, E. Yoneki, S. Y. Chan, and J. Crowcroft, "Distributed community detection in delay tolerant networks," in *Proceedings of the 2nd ACM International Workshop on Mobility in the Evolving Internet Architecture (MobiArch '07)*, Article no. 7, New York, NY, USA, August 2007.
- [10] F. Li and J. Wu, "LocalCom: a community-based epidemic forwarding scheme in disruption-tolerant networks," in *Proceedings of the 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '09)*, Rome, Italy, June 2009.
- [11] E. Bulut and B. K. Szymanski, "Friendship based routing in delay tolerant mobile social networks," in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM '10)*, Troy, NY, USA, December 2010.

- [12] F. Fabbri and R. Verdone, "A sociability-based routing scheme for delay-tolerant networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, Article ID 251408, 13 pages, 2011.
- [13] A. Mei, G. Morabito, P. Santi, and J. Stefa, "Social-aware stateless forwarding in pocket switched networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 251–255, Shanghai, China, April 2011.
- [14] A. Mtibaa, M. May, C. Diot, and M. Ammar, "PeopleRank: social opportunistic forwarding," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '10)*, San Diego, Calif, USA, March 2010.
- [15] W. Moreira, P. Mendes, and S. Sargento, "Opportunistic routing based on daily routines," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pp. 1–6, 2012.
- [16] Y. Zhu, H. Zhang, and Q. Ji, "How much delay has to be tolerated in a mobile social network," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 358120, 8 pages, 2013.
- [17] The ONE 1.4.1 [EB/OL], Nokia Research Center, Helsinki, Finland, 2010, <http://www.netlab.tkk.fi/tutkimus/dtn/theone/>.
- [18] J. Scott and R. Gass, "Crawdad trace cambridge/haggle (v.2006-09-15)," 2006, <http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote/content>.

