

# Prediction-Based Routing Methods in Opportunistic Networks

Sanfeng Zhang<sup>1</sup>, Di Huang<sup>1</sup> and Yin Li<sup>2</sup>

<sup>1</sup>Key Laboratory of Computer Network and Information Integration of Ministry of Education  
Southeast University, Nanjing, 211189 - China  
[e-mail: sfzhang@seu.edu.cn, seusofthd@gmail.com]

<sup>2</sup>College of Software Engineering  
Southeast University, Nanjing, 211189 - China  
[e-mail: yinli@seu.edu.cn]

\*Corresponding author: Sanfeng Zhang

*Received July 10, 2014; revised April 2, 2015; accepted August 12, 2015;  
published September 25, 2015*

*Received July 10, 2014; revised April 2, 2015; revised June 3, 2015; revised July 21, 2015;  
accepted August 12, 2015; published October 31, 2015*

---

## Abstract

The dynamic nature of opportunistic networks results in long delays, low rates of success for deliveries, etc. As such user experience is limited, and the further development of opportunistic networks is constrained. This paper proposes a prediction-based routing method for opportunistic networks (PB-OppNet). Firstly, using an ARIMA model, PB-OppNet describes the historical contact information between a node pair as a time series to predict the average encounter time interval of the node pair. Secondly, using an optimal stopping rule, PB-OppNet obtains a threshold for encounter time intervals as forwarding utility. Based on this threshold, a node can easily make decisions of stopping observing, or delivering messages when potential forwarding nodes enter its communication range. It can also report different encounter time intervals to the destination node. With the threshold, PB-OppNet can achieve a better compromise of forwarding utility and waiting delay, so that delivery delay is minimized. The simulation experiment result presented here shows that PB-OppNet is better than existing methods in prediction accuracy for links, delivery delays, delivery success rates, etc.

---

**Keywords:** opportunistic network, optimal stopping, link prediction, time series

---

This research was supported by a grant of National Natural Science Foundation of China [No. 61300200 (Research on Key Technologies for Wireless Mesh Networks Based on Network Coding and Opportunistic Forwarding) and 61472080 (Architecture and Key Technologies for the Complementary Dual-structural Future Network)].

## 1. Introduction

Opportunistic Networks (OppNet) [1] are new network architectures derived from Ad hoc networks. They can transfer the packets through multiple hops while the nodes are moving around, and can finally reach the destination node when in a separate network. Contact refers to a connection between the nodes. When the nodes (for instance, a vehicle equipped with an 802.11p communication device or a mobile smartphone user) enter the communication range of each other, then the link is established and the communication occurs. When leaving the communication range, the link and the communication are disrupted. In opportunistic networks, contact is opportunistic rather than deterministic. Due to the mobility of the user, the end-to-end link is no longer reliable, which makes traditional routing ineffective. Instead, opportunistic networks perform a “store-carry-forward” model to deliver messages via multiple hops.

Opportunistic networking is low-cost and easy to deploy. These characteristics make it a basic network architecture in Vehicular Ad Hoc Networks (VANET), mobile data offloading [2], cooperative content sharing [3][4], mobile computing offloading [5], etc. With opportunistic networks, message delivery, content distribution, resource sharing and other functions can be realized regardless of the infrastructures and only need of a small number of infrastructures in the remote highways, urban transport, mobile social networking and other scenes. In conclusion, these opportunistic networks application systems have many prospects. Their performance and the experience of users are largely dependent on the network transmission service provided by opportunistic networks.

However, the dynamics of the network topology is still a key constraint of opportunistic networks' service quality. The NetSense dataset [6] is a collection of the Bluetooth and WiFi connection logs of two hundred students in The University of Notre Dame. The result [6] shows that the average encounter time interval is about 16 hours. Thirty percent of the node pairs have an encounter time interval no more than tens of minutes, while twenty percent of the nodes pairs have an encounter time interval over a week. From the perspective of link duration, forty percent of node pairs have link durations no more than one minute. The dynamics make it difficult to determine a forwarding path. Therefore, in order to improve the performance of transmissions, most of the state-of-art methods use multiple copies of the message. Besides, they choose suitable next hops according to the criteria of forwarding utility [7]. However, there are still several problems. (1) The success rate of delivery is low. Messages are discarded while transferring because of timeouts or buffer overflow. The success rate of delivery is normally 50% [1]. (2) The delays of deliveries are long. Because of the dynamics, the store-carry is aimless. It may take hours or days to successfully deliver the message. (3) Multiple copies stored and transferred in the network are bound to require a lot of storage space and consume lots of energy. (4) Due to the lack of real-time end-to-end paths in networks, the response of network service requests is unknown and the waiting time is unpredictable, which significantly affects the user's experience.

In many opportunistic networks, the movement of nodes and the contacts between nodes is a reflection of people's daily movements and activities. Individual activities always follow a routine. Based on this assumption, we believe that it is possible to predict the dynamic changes of links in opportunistic networks. In this paper, we propose a prediction-based routing method in opportunistic networks. Firstly, we use the predicting algorithm to obtain the average encounter time interval of the nodes pairs, and regard these as the forwarding utility to

the destination node. Each node learns this forwarding utility of the nodes that it contacted in the historical time series, and then predicts the forwarding utility of the nodes that it will contact in the future. At last, the forwarding utility and the waiting delay are compromised to obtain the minimum delivery delay.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 describes the framework of the prediction based routing algorithm. Section 4 describes the optimal stopping decision method for routing. Section 5 describes the link prediction algorithm. Section 6 is a simulation experiment with a performance evaluation. The last part is a conclusion.

## 2. Related Work

Routing schemes in opportunistic networks can be classified into several types. The most typical is a flooding scheme, with or without a limited buffer size, such as the work of Spyropoulos et al. [1]. Too many replications will greatly degrade the performance of this type of routing method. The prediction-based routing algorithm, including algorithms that predict forwarding utilities using historical contact information such as Prophet [7] and OPF[8], can't describe periodical mobility of nodes properly. Recently many social-based routing protocols have been proposed, such as Bubble[9] and SDM[10]. Bubble builds up cumulative contact length of time for community detection. SDM overcomes this deficiency but we can see that many nodes in the real world like taxis do not have this societal property. Therefore, social-based routing may become inefficient in some cases.

In a variety of routing algorithms in opportunistic networks, the key technique is about how to take full advantage of the historical evolution of the network topology and predict the changes of links. In the early design of routing in opportunistic networks, routing features that do not change over time are used to indicate the probability of the next link's establishment, such as the average contact length of time, contact frequency, etc. There is a prediction method based on contact length of time in history[7]. Nodes that have more frequent contacts in the past have a larger contact probability in the future. If two nodes haven't contacted each other for a long time, the probability of a link establishment between them will age as time goes by. Using average contact intervals in history is also a simple and effective link prediction method[8]. Each node in the network dynamically maintains the average contact interval with other nodes. The smaller the time interval is, the more frequently they meet, and the higher the probability of a link establishment is in the future. The duration of the link is also an effective measurement. In conclusion, this kind of link prediction method uses the historical information of a link. They can be regarded as the simplest predicting methods based on a time series. They do not need global information and have a low computational complexity, and it's easy to realize these methods. However, they lack a theoretical basis, can't discern internal rules of link changes, and can't predict links that never occurred in history.

Opportunistic networks are consistent with the statistical properties of complex networks, so we can use methods of complex networks for link prediction in opportunistic networks. Link prediction based on similarity [11] refers to predicting the probability of links by the similarity between nodes. If two people have the same topology features such as common neighbors, or they have the same age, gender, profession, interests and so on, they must be very similar. Then the probability of a link establishment between these two people is high. Stochastic block modeling (SBM) [12] is a method based on maximum likelihood. Its basic idea is to divide the nodes in networks into groups using the modularity characteristic of networks. Whether there is a connected link between the two nodes or not is determined by the groups they belong to. Link predictions based on complex networks are actually methods that

regard the historical link information as static, known, global topological information to predict unknown or impending links, rather than reflecting the rules of topologies evolving over time. It also lacks a consideration of the given date, place and other known conditions. We need to combine some methods of the time series analysis and the parameter estimation to improve the application efficiency of link predictions based on complex networks.

Mobility prediction is also explored in VANETs [14] and mobile communication [15][16][17]. ERBA [14] is an energy-efficient routing protocol. It predicts movement trends and makes routing recommendations using information including current directions, vehicular category information, driving behavior patterns and road map information. The strategy in Ref.[15] forecasts user mobility based on telecom calls and implements a cloud-based location related service that can make online mobility predictions for value-added telecom services. NextCell [16] is a novel algorithm that aims to enhance the location prediction by harnessing the social interplay revealed in cellular call records.

### 3. Opportunistic Network Model and Routing Framework

#### 3.1 Opportunistic network model

All nodes moving in a limited area, of which the number is  $NUM$ , constitute a node set  $V$ . Each node  $i \in V$  is equipped with an omni-directional antenna. According to their given mobility model, nodes belong to different kinds have different movement patterns but have the same communication range. Nodes within each other's transmission range can only perform single-hop data transmission directly while indirect links of multi-hops do not exist. We suppose that all nodes have the same constant bandwidth and can generate messages with certain destinations. Buffers of all nodes are the same and all messages are stored in it. Assume that the network runs in time slots with length  $T$ . Nodes in idle channels inform others of their configuration information and detect the configuration of their neighboring nodes. We also suppose that two nodes can replicate all the messages that satisfy the forwarding policy when they meet with each other. Each message has two attributes: the remaining hop count  $H$  and the time-to-live  $TTL$ . When replication occurs in two nodes, the hop counts of the copies in these two nodes become  $H-1$ . If the  $H$  value in one replica decreases to 0, the replica cannot be replicated by other nodes except for the destination node.  $TTL$  decreases with time slots and the message is deleted from the buffer if  $TTL=0$ .

#### 3.2 Routing algorithm Framework

$$\begin{array}{ccc}
 \overline{TS}_j & & \overline{T}_{all} \\
 \left[ \begin{array}{ccc} 100 & 350 & \dots \\ 20 & 60 & \dots \\ \dots & \dots & \dots \\ 1500 & 1000 & \dots \end{array} \right] & & \left[ \begin{array}{cccc} 0 & t_{1 \rightarrow 2} & \dots & t_{1 \rightarrow NUM} \\ t_{2 \rightarrow 1} & 0 & \dots & t_{2 \rightarrow NUM} \\ \dots & \dots & 0 & \dots \\ t_{NUM \rightarrow 1} & t_{NUM \rightarrow 2} & \dots & 0 \end{array} \right] \\
 \text{(a) interval series matrix} & & \text{(b) interval matrix}
 \end{array}$$

$$\begin{array}{c}
 \overline{M}_j \\
 \text{slots} \\
 \left[ \begin{array}{ccc} 0 & t_{2 \rightarrow 1} & \dots \\ t_{1 \rightarrow 2} & 0 & \dots \\ \dots & \dots & \dots \\ t_{1 \rightarrow NUM} & t_{2 \rightarrow NUM} & \dots \end{array} \right] \\
 \text{(c) observed value of } T_d
 \end{array}$$

Fig. 1. Data structures maintained by node  $j$

In order to run the routing algorithm, each node maintains several data structures shown in Fig.1. Fig. 1(a) shows an encounter time interval sequence matrix noted as  $\overline{TS}_j$  where the  $i$ -th row records the encounter time interval sequence between node  $j$  and node  $i$ . This interval

sequence matrix will be used by the link prediction algorithm based on time series analysis to deduce the next time interval between node  $j$  and other nodes. Fig. 1(b) shows a predicted time interval matrix noted as  $\bar{T}_{\text{all}}$  where element  $t_{ij}$  represents the predicted average encounter time interval between node  $i$  and node  $j$ . Fig. 1(c) is an observation matrix denoted by  $\bar{M}_j$  where one column is added for each time slot, and each column records the predicted encounter time interval between one reporting node and all the other nodes, so the elements of the  $d$ -th row form a sequence, recording encounter time intervals between potential forwarding nodes in turn and node  $d$ .

Data structures are maintained as follows. Node  $i \in V$  checks the channel at each time slot. If the channel is an empty node  $i$ , it will broadcast a configuration message containing the node ID and an vector of the average encounter time intervals  $\bar{T}_i = (t_{i \rightarrow 1}, t_{i \rightarrow 2}, \dots, t_{i \rightarrow \text{NUM}})$ , where the variable  $t_{i \rightarrow d}, (d \in V)$  represents the average encounter time interval between node  $i$  and another node  $d$ . If node  $j \in V$  successfully receives the configuration message from node  $i$ , node  $j$  will first update matrix  $\bar{TS}_j$ , adding an element recording the time interval between the last encounter of the node pair  $(i, j)$  and the new one. Subsequently, node  $j$  will replace the  $i$ -th row of  $\bar{T}_{\text{all}} (NUM \times NUM)$  with vector  $\bar{T}_i$ . Finally, node  $j$  adds  $\bar{T}_i$  as a column to  $\bar{M}_j$ . Node  $j$  regards all the  $t_{i \rightarrow d}$  as the observed values of  $T_d$  and fits a distribution of random variables  $T_d$  on these values for subsequent routing decisions.

If node  $j$  receives a configuration message from  $i$  in some time slot, it will decide which messages to replicate to  $i$  according to the routing decision method based on the optimal stopping theory. If node  $j$  copies the message with remaining hop  $H$  to  $i$ , the hop counts in both  $i$  and  $j$  will become  $H-1$ . If  $H=0$ , messages can only be copied to destination node  $d$ . Thus the maximum number of replicas of a message in the network are  $2^H$ .

## 4. Optimal Stopping Decision Methods for Routing

### 4.1 Decision problem of the minimum expected delay

As described before, when node  $j$  takes a message destined to  $d$  and receives the configuration message from node  $i$  in slot  $N$ , it will decide whether to copy the message to  $i$  according to the observed value  $t_{i \rightarrow d}$ . Two cases exist in the decision process: one is that  $t_{x \rightarrow d}$  of node  $x$  that it observes in the following time-slots are all larger than current observed value  $t_{x \rightarrow d}$ , thus  $j$  should replicate the message to  $i$  immediately. The other case is that  $t_{y \rightarrow d}$  of node  $y$  that it met in time slot  $N+1$  satisfies  $t_{y \rightarrow d} + T < t_{x \rightarrow d}$ , thus delivering this message to node  $y$  in slot  $N+1$  with a lower delay. Therefore, node  $j$  needs to select an appropriate slot to obtain a lower delay. This problem can be modeled as an optimization of minimizing expected delivery delays.

As messages are copied among multiple nodes, nodes holding the message  $m$  form a node set  $V_m$ . Thus, the routing decision of node  $j$  ( $j \in V_m$ ) aims to minimize the expectation of the minimum average delay  $E(\min_{s \in V_m \cup \{i\}} \{T_{s \rightarrow d}\})$  in set  $V_m \cup \{i\}$  destined to  $d$  by choosing a slot  $N$  and copying the message to  $i$  that  $j$  meets in  $N$ . We describe this problem as **MED** (Minimum Expected Delay) and solve it based on the optimal stopping rule.

### 4.2 Selection of forwarding slots using optimal stopping rules

Define a sequence of random variables  $X_1, X_2, \dots$ , with a known joint distribution and reward functions  $y_0, y_1(x_1), y_2(x_1, x_2), \dots, y_\infty(x_1, x_2, \dots)$ . After observing  $X_1=x_1, X_2=x_2, \dots, X_n=x_n$ , we can

either stop and receive the reward  $y_n(x_1, x_2, \dots, x_n)$ , or continue to observe  $x_{n+1}$ . Optimal stopping rules choose a stopping time  $N^*$  to maximize the expected reward  $E[Y_{N^*}]$ .

**Assumption 1.** The random sequence, which consists of the contact time interval between destination node  $d$  and node  $i$  encountered in each time slot, is independent and identically distributed.

For the MED problem, the optimal rule is to choose a slot  $N^*$  to satisfy:

$$N^* = \arg E \left( \frac{\min}{N=0,1,2,\dots} \{T_{s \rightarrow d} \mid s \in V_m\} \cup \{N \cdot T + T_{i \rightarrow d}\} \right) \quad (1)$$

or :

$$N^* = \arg \frac{\max}{N=0,1,2,\dots} E(T_{s \rightarrow d} - T_{i \rightarrow d} - N \cdot T) = \arg \frac{\max}{N=0,1,2,\dots} E(Y_N) \quad (2)$$

In the formula,

$$Y_N = T_{s \rightarrow d} - T_{i \rightarrow d} - N \cdot T, \quad N = 1, 2, \dots \quad (3)$$

represents the reward function of MED problem. According to assumption 1,  $T_{i \rightarrow d}$  is also independent and identically distributed.

As the above definitions, the contact interval  $T_{i \rightarrow d}$  between  $i$  and  $d$  is the random variable, and  $T_{s \rightarrow d}$ ,  $s \in V_m$  is known. For computational convenience, we consider that the node  $s$  observes itself and  $T_{i \rightarrow d} = T_{s \rightarrow d}$  if no other node is observed in a time-slot. In addition, we believe that in most cases a node observes only one or zero nodes in a time slot. This has been proved by the simulation results. Therefore, we can select the best node whose  $T_{i \rightarrow d}$  is the smallest as the observed value for the approximation if the node observes several nodes.

#### Proposition-1

For the MED problem, there exists a stopping rule  $N^*$ , and:

$$N^* = \min \{N \geq 1 : (T_{s \rightarrow d} - T_{i \rightarrow d}) \geq V^*\} \quad (4)$$

where  $V^*$  is the solution of the following equation:

$$E[T_{s \rightarrow d} - T_{i \rightarrow d} - V^*]^+ = T \quad (5)$$

The notation  $[*]^+$  means to select only positive values.

Proposition-1 shows that messages can be replicated to the current observed node if the observed value is within a range.

### 4.3 Existence proof and solutions of the optimal stopping rule

We use optimal stopping theory to prove Proposition-1.

The description in the reward function (3) indicates that every node needs to probe at least one other node at least one time if it wants to replicate the message to others. Thus,  $N$  is a non-zero natural number. Next, we obtain the optimal rule through a general method.

#### First Step: prove the existence of the optimal stopping rule.

According to Theorem 3.1[13], the optimal stopping rule exists when satisfying the following two conditions:

$$\begin{aligned} \text{A1. } & E\{\sup_N Y_N\} < \infty \\ \text{A2. } & \limsup_{N \rightarrow \infty} Y_N \leq Y_\infty \end{aligned} \quad (6)$$

From the definition of our reward function  $Y_N$ , we can easily find that  $\limsup_{N \rightarrow \infty} Y_N = -\infty$  and  $Y_\infty = -\infty$ . That is because  $-N \cdot T \rightarrow -\infty$  when  $N \rightarrow \infty$  and  $(T_{s \rightarrow d} - T_{i \rightarrow d})$  has a range

whose  $\max(T_{s \rightarrow d} - T_{i \rightarrow d})$  and  $\min(T_{s \rightarrow d} - T_{i \rightarrow d})$  exist and have specific values in a network. Therefore  $\limsup_{N \rightarrow \infty} Y_N \leq Y_\infty = -\infty$  and A2 are proved. Also, for any  $N=1, 2, 3, 4, \dots$ ,  $\sup_N Y_N < \infty$  so  $E\{\sup_N Y_N\} < \infty$  and A1 is proved.

In summary, the reward function satisfies both A1 and A2. Therefore optimal stopping rules exist in our reward function.

**Second step : obtain the solutions of the optimal stopping rule.**

According to solutions of 4.1 [13] and theorem 3.2, for the reward function  $Y_N = X_N \square N \cdot T$  of the MED problem, if the distribution of  $X_N$  is time-independent, then the optimal stopping rule presents a threshold structure. In other words, let  $V^*$  denote the expected return of the optimal stopping rule. If  $T_N < V^*$ , the node should continue observing, and if  $T_N \geq V^*$  the node should stop and replicate. In this way, combined with Assumption 1, the stopping rule becomes:

$$N^* = \min\{N \geq 1 : X_N \geq V^*\} \quad (7)$$

According to the optimality equation (3.2 in [13]), solutions to stopping rule  $v^*$  are:

$$\begin{aligned} V^* &= E \max\{T_1, V^*\} - T \\ &= \int_{-\infty}^{V^*} V^* \cdot dF(t) + \int_{V^*}^{\infty} t \cdot dF(t) - T \end{aligned} \quad (8)$$

Let:

$$V^* = \int_{-\infty}^{+\infty} V^* \cdot dF(t) \quad (9)$$

Combining (8) and (9):

$$\begin{aligned} \int_{-\infty}^{+\infty} V^* \cdot dF(t) &= \int_{-\infty}^{V^*} V^* \cdot dF(t) + \int_{V^*}^{\infty} t \cdot dF(t) - T \Rightarrow \\ \int_{-\infty}^{V^*} V^* \cdot dF(t) + \int_{V^*}^{\infty} t \cdot dF(t) - \int_{-\infty}^{+\infty} V^* \cdot dF(t) &= T \Rightarrow \\ \int_{V^*}^{\infty} (t - V^*) dF(t) &= T \end{aligned} \quad (10)$$

In these formulas,  $F$  represents the common distribution function of  $T_N$ .

For a discrete variable  $T_N = T_{s \rightarrow d} - T_{i \rightarrow d}$ , we can deduce the following result in a similar way:

$$E(T_N - V^*)^+ = T \quad (11)$$

Therefore, Proposition 1 is proved.

**Calculation methods for the threshold value  $V^*$ .**

There are two methods to calculate the threshold  $V^*$ . The first method is to regard the  $T_N$  as a discrete variable and use the formula (11). It is necessary to sort all the possible values in descending order and try each value interval to see if it satisfies (11). This will cost  $O(n)$  operations. The other method is to regard the  $T_N$  as a continuous variable and use (9). We would have to fit a curve and it is a little bit complex. Thus, we adopt the first method.

#### 4.4 Analysis for Assumptions of the Optimal Stopping Rule

When we use the optimal stopping rule to solve the MED problem, these assumptions must be met: random variables  $X_N$  (in our model  $T_N = T_{s \rightarrow d} - T_{i \rightarrow d}$  are the random variables) must be independent and identically distributed (i.i.d). There is a strong relationship between the validity of the assumption and the granularity of the time slot and the mobility model. If the node movement area is uniform (there are no such different types of area like urban and



suburb) and node velocities fluctuate narrowly, random variables  $T_N$  tend to be identical. If the node movement range is large and the time-slot granularity is big enough,  $T_N$  will have less relativity. For simplicity, we consider time-slot granularity and movement area in our network to satisfy the i.i.d assumption. In practical applications, we can divide the area into several sub-regions to enhance the validity of this assumption. We will illustrate satisfactory situations with statistical data in experimental evaluations.

### 4.5 Analysis of Message Replication Processes

Based on the routing framework and the optimal stopping rule, PB-OppNet works as follows.

In Fig. 2, if maximum hop  $H$  is limited to 2, the forwarding process is as following. For the case where  $V_m = \{a\}$  when  $H=2$ , if in some time-slot, node  $b$  satisfies formula (3) by calculation of formula (4),  $a$  will replicate the message to  $b$ . We must add the common distribution of  $T_{i \rightarrow d}$  to the message for the later calculation of  $v^*$  in formulas (3) and (4). After this is done,  $V_m = \{a, b\}$  and  $H=1$ . The result is shown in Fig. 2.(a). When  $b$  meets  $c$ ,  $b$  will also determine whether to replicate or not. However, this time PB-OppNet needs to update threshold  $v^*$  using the distribution of both a and b because there are two observation points (a and b) for the message in  $V_m$ . Both of these nodes can forward the message, so we can treat them equally and the equality lies in the  $T_{i \rightarrow d}$  distribution updates, which is replaced with an average on a and b's common distribution. If  $c$  gets the replica, the results will be like Fig. 2. (b).

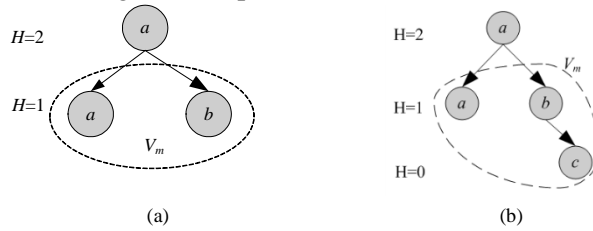


Fig. 2. Copy messages with 2 hops

As Fig. 3 shows, if  $H$  is limited to 3,  $H$  equals 1 after two replications and  $c$  meets  $y$  in some time slot. The following situation may exist in Fig. 3 (a). There,  $c$  knows  $V_m' = \{a, b, c\}$  to be the set that holds the message replicas, but instead the set is  $V_m = \{a, b, c, x\}$ . Due to inaccurate knowledge in  $c$ , the calculation using formula (4) for the optimal stopping rule may lead to differences. This kind of difference grows as the hop count increases. In Fig. 3 (b),  $V_m'$  has two fewer elements than  $V_m$ .

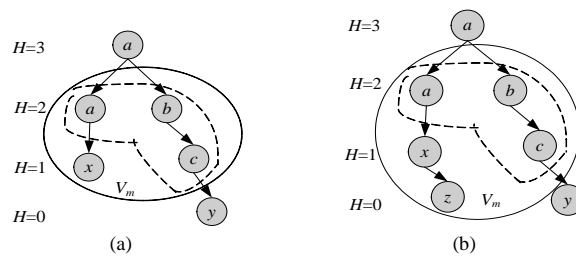


Fig. 3. Copy messages with 3 hops



In conclusion, when  $H=2$ , we can calculate all stopping moments accurately every time the node performs a replication according to (4), but this also limits the maximum number of replicas to four. When  $H>2$ , the stopping rule calculation may be inaccurate due to the lack of global knowledge of set  $V_m$ . In this process, we can see that it will cost  $O(n)$  time if nodes dynamically calculate the threshold for every message.

## 5. Prediction Algorithm of Contacting Intervals

The changes of network topology over time are divided into several snapshots of equal length. Each snapshot may include one or more slots. According to the links in each snapshot, we obtain a weighted undirected graph and an adjacency matrix as shown in Fig. 4. The edge weight or the elements of the matrix represents the number of link establishments in the snapshot. Using all the link changes in the history, we can obtain a series of snapshots represented by  $(G_1, G_2, G_3, \dots, G_t)$ . Thus, the link prediction refers to predicting  $G_{t+1}$  according to the given snapshots series  $(G_1, G_2, G_3, \dots, G_t)$ .

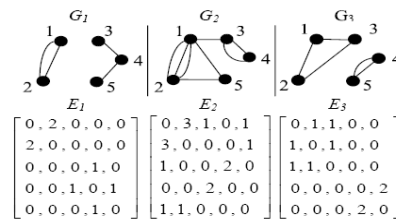


Fig. 4. Network snapshots

If we use the link prediction algorithm based on complex networks and the snapshots series in Fig. 4, we need to know the global topology information. Then we can grasp the dependencies between links and predict the impending links that are absent in the history. Another link prediction algorithm is based on time series, thus only needs the contacting frequency in history of one pair of nodes and can predict the link establishment of this pair. These two algorithms are conceptually orthogonal and can complement each other to improve the link prediction accuracy. However, according to the routing algorithm framework described in section 3.2, each node only completely stores the topology information of links ever related to itself. Furthermore, considering the complexity of the algorithm, we choose to use the univariate time series analysis method for link prediction.

Firstly, according to the  $\overline{TS}_j$  matrix of node  $j$  as shown in Fig. 1, row  $i$  indicates the historical contacting interval between node  $i$  and node  $j$ , which is represented as  $(M_1(i,j), M_2(i,j), \dots, M_t(i,j))$ . For the next contacting interval  $M_{t+1}(i,j)$  must be predicted, and  $(M_1(i,j), M_2(i,j), \dots, M_t(i,j))$  refers to the background interval series. Then we use the general ARIMA  $(p, d, q)$  model (Auto Regressive Integrated Moving Average Model) for time series prediction. In the formula,  $p$  is the autoregressive order,  $q$  is the order of the moving average, and  $d$  is the difference times in order to make the time series stable.

ARIMA  $(p, d, q)$  is represented as :

$$\varphi_p(B)(1-B)^d M_t(i, j) = \theta_0 + \theta_q(B)w_t \quad (12)$$

$B$  is back operator,  $BM_t(i, j) = M_{t-1}(i, j)$ . The operator  $\varphi_p, \theta_q$  separately represents the polynomial function of  $B$ :  $\varphi_p(B) = (1 - \varphi_1 B - \varphi_2 B^2 - \dots - \varphi_p B^p)$  and  $\theta_q(B) = (1 + \theta_1 B + \theta_2 B^2 + \dots + \theta_q B^q)$ .

The average of  $w_t$  is zero. The variance is the white noise sequences of  $\delta_w^2$ . The sequences  $\varphi_1, \varphi_2, \dots, \varphi_p$  and  $\theta_1, \theta_2, \dots, \theta_q$  are constants. By the function  $\theta_0 = u(1 - \varphi_1 - \varphi_2 - \dots - \varphi_p)$ ,  $u$  is the average.

In the model fitting process, we choose optimal models from spaces of  $p = 0, 1, 2, 3$ ,  $d = 0, 1$ ,  $q = 0, 1, 2, 3$ . We choose the minimum information criterion, AIC (Akaike Information Criterion). Then we use its optimal model to predict the interval of the  $T+1$  contacting  $M_{T+1}(i, j)$ . AIC is calculated as follow:

$$AIC = \ln\left(\frac{\sum_{t=\max(p,q)+1}^T (M_t(i, j) - M_t(i, j))^2}{T - \max(p, q)}\right) + \frac{2(p + q + 2)}{T - \max(p, q)} \quad (13)$$

## 6. Simulation and Performance Evaluation

### 6.1 Scenario and Parameter Settings

Simulations and evaluation of PB-OppNet are implemented in ONE (Opportunistic Network Environment) [18], which is specially designed for opportunistic networks. In order to simulate real life opportunistic networks, we introduce the data set provided by the ‘‘Cab-spotting’’ project. This data set [19] records more than 500 taxis for 30 days in San Francisco and each taxi reports the location every 60 seconds. We select different times in the data set as simulation seeds to ensure simulation differences if we want to do the experiment several times. Other parameters are set in Table 1.

**Table 1.** Parameters for simulations

Parameter	Value
Number of node	500
Transmit range (m)	30
Transmit speed (Mbps)	10
Buffer size (MB)	5-50
Message size (KB)	500~1024
Message creation interval (s)	30
Slot time (s)	60
Message time to live (h)	333
Simulation time (h)	333

We must first verify the optimal stopping model assumptions and introduce the internal tuning process of PBR-OppNet, covering factors including the interval prediction, the slot length, the global information and local information, and then the optimized PBR-OppNet is compared with several methods.

We compare PBR-OppNet with Epidemic, SprayAndWait and Prophet. Epidemic is a pure flooding scheme where each node copies messages to any node it encounters. SprayAndWait [1] is a flooding scheme with a limited buffer size, where each node copies messages to limited early encounter nodes and then waits for the target node to appear. Prophet [7] is a prediction based routing algorithm, which predicts forwarding utility using historical contact information. Comparison metrics include delivery rate, forwarding cost, and average delay. Delivery rate is the number of delivered messages/number of all messages. Forwarding cost (overhead ratio) = (number of forwarding – number of delivered messages)/ number of delivered messages.

Average delay = the mean time of the messages delivered from the source node to the destination node.

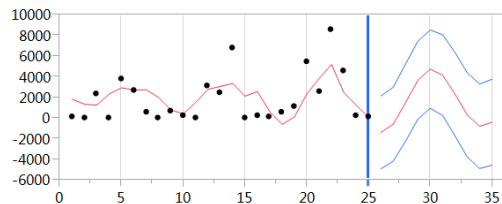
In the comparison, the network scale and the buffering size are two most important parameters. Other parameters may include bandwidth, moving speed and density of nodes etc. The speed of motion is fixed by the real tracing data set. The bandwidth factor is also simplified in our model by assuming that the bandwidth is large enough, so that the communication time can meet the demand for all messages delivered. The density of nodes has been reflected in the scale of the network parameter, because the fixed regions of the nodes distribution and the number of nodes representing density change.

**6.2 Contact Interval Prediction Performance**

A pair of nodes  $i$  and  $j$  are chosen. Thirty previous intervals of this node pair are used to evaluate the interval prediction process. The first 25 intervals are training data. We choose an optimal model from spaces of  $p = 0,1,2,3$ ,  $d = 0,1$ ,  $q = 0,1,2,3$ . Models with  $d=1$  have better stationary behavior than those with  $d=0$ . AIC values for ARIMA models with  $d=1$  are listed in **Table 2**. Therefore, ARIMA(2,1,3) is chosen as the best model to predict the interval of the  $T+1$  contacting  $M_{T+1}(i, j)$ . The curve predicted by ARIMA (2, 1, 3) and the prediction precision are shown in **Fig. 5** and **Table 3**.

**Table 2.** AIC values for ARIMA models

	MA(0)	MA(1)	MA(2)	MA(3)
AR(0)	60.61	61.49	62.27	64.26
AR(1)	61.26	63.25	64.27	63.44
AR(2)	63.22	64.64	64.20	57.04
AR(3)	62.97	62.45	57.53	59.97



**Fig. 5.** predicted curve of ARIMA(2,1,3)

**Table 3.** prediction precision

Time	Real	Predicted	Error
26	12	7	42%
27	180	358	98%
28	24	13	46%
29	96	84	12%
30	9696	5716	41%

**6.3 I.I.d Assumption Explanation of Random Variable XN in Each Time Slot**

Firstly, using the assumption 1, we illustrate the random variables  $T_{i \rightarrow d}$  in each slot satisfy an identical distribution. We set the time-slot to 60s to be consistent with performance evaluating experiments. We run 20 slots each time and repeat the experiment 10000 times. **Fig. 6** shows

the cumulative distribution of  $T_{i \rightarrow d}$  in the first three slots of node 1. We can conclude that  $T_{i \rightarrow d}$  substantially has the same distribution in the first three slots.

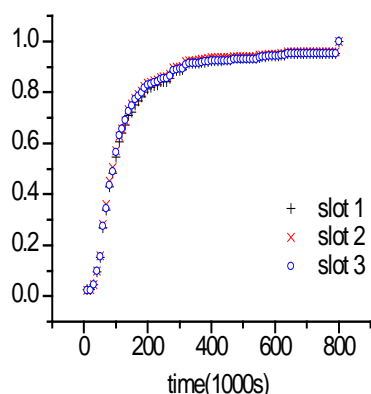


Fig. 6. Distribution of  $T_N$

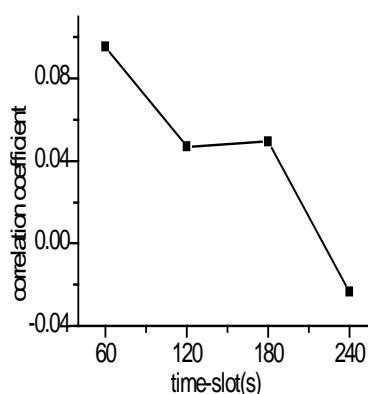


Fig. 7. Correlation of  $T_N$ .

According to probability theory, when the joint distribution of a two-dimensional random variable  $(X, Y)$  is a two-dimensional normal distribution, the independence and irrelevance of  $X$  and  $Y$  are equivalent. Therefore, we can measure their independence by the correlation of two random variables. Fig. 7 gives the relationship between the correlation of two adjacent slots and slot lengths. The figure indicates that when the slot length is 60s, the correlation is lower than 0.1. As the slot length increases, the correlation substantially decreases. When the slot length is over 120s, the correlation will be nearly zero.

#### 6.4 Performance Comparison under Conditions of Global Information and Local Information

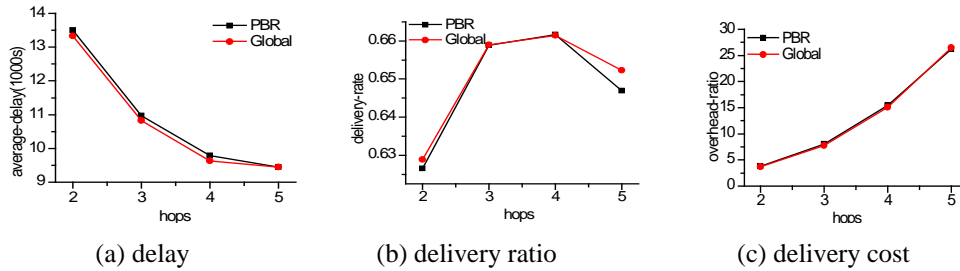
As above, if the node makes a decision only according to the partial information, there may be unnecessary information duplication and storage, thus forming some second best paths besides the theoretical optimal path in the network. It may cause a waste of storage and transmission resources.

Fig. 8 assesses the extent of this problem. The abscissa is the hop count limit, which is the limit of the number of copies. Among them, the overhead ratio is defined as  $(\text{number of copies} - \text{number of messages}) / \text{number of messages}$ . It shows that these two cases have very similar results in delay, delivery rate and overhead. Local information will cause deviation of the optimal stopping rule's result, but the deviation is constrained by two factors: the real number of hops to forward the message and the distribution of the average contacting time  $T_{i \rightarrow d}$  of the nodes.

First, the real number of hops is small when considering factors such as the lifetime of the message, cache space, and network scale. The results of the experiment show that if the upper limit of the number of hops is larger than 4, it will cause a significant increase in the number of copies, the load of transmission and storage. Therefore, the cost of message delivery increases while the success rate decreases. In the case where the overall number of hops is small, the difference between the local and global information collection is limited.

Second, since  $T_{i \rightarrow d}$  is approximately normally distributed, the optimal stopping rule is actually to reduce the value of  $E(\min_{s \in V_m \cup \{i\}} \{T_{s \rightarrow d}\})$  after adding a new node. With the increase in the number of hops, the probability of contacting a better node is reduced, and therefore, the

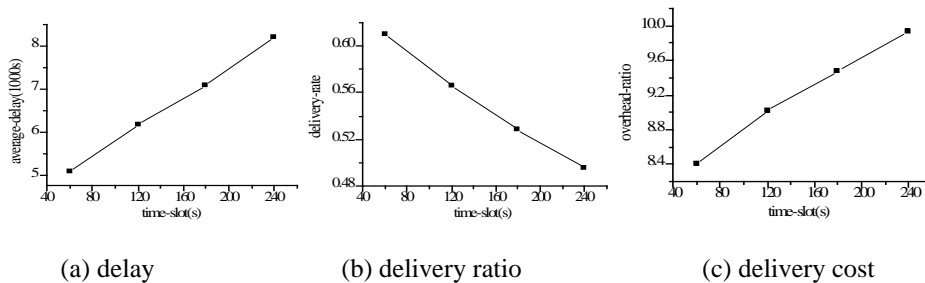
number of second best paths is limited. Taking into account the stochastic nature of the optimization method, the cost of the second best path is not completely wasted, which can be used to reduce the delivery delay. Therefore, using local information to approximate global information has only a small effect on the overall performance of the network.



**Fig. 8.** Performance comparison under conditions of global information and local information

### 6.5 Slot Length Influence on PBR-OppNet Performance

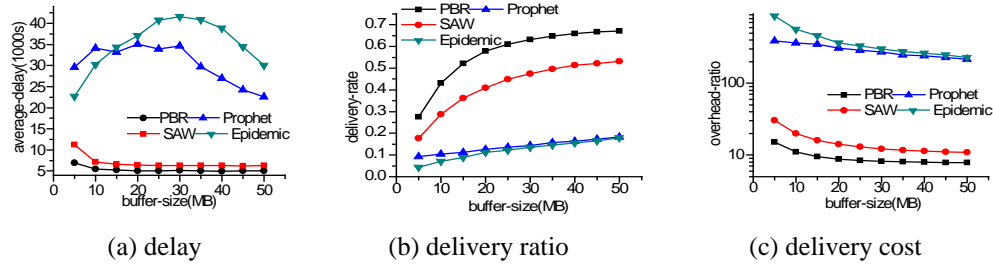
We then evaluate the influence of slot length on PBR-OppNet performance. We can see in [Fig. 9](#) that these three metrics perform better as the slot length decreases, since a smaller slot means a sharper routing decision. However, considering that the data gathering cycle is 60s, a short slot leads to an increase in the decision frequency and calculation costs, and short slots also makes the frequency of contacting no nodes higher, so we choose a length of 60 seconds as the time-slot length in PBR-OppNet algorithm.



**Fig. 9.** Performance comparison of PBR-OppNet with different length of slots

### 6.6 Buffer Size Influence on Performance

We evaluate the performance of PBR-OppNet with different buffer limitations according to the protocols such as SprayAndWait. [Fig. 10](#) clearly shows that PBR-OppNet performs better than the three other algorithms in all three metrics. SprayAndWait is near to PBR-OppNet in delay and overhead but has a delivery rate lower than PBR-OppNet between 10~20% because it adopts a greedy strategy to replicate as soon as it meets other nodes. This leads to fast replication and relatively short delay, but the probability of buffer overflow and deleting messages increases, resulting in an increase in costs and a decrease in delivery success rate.



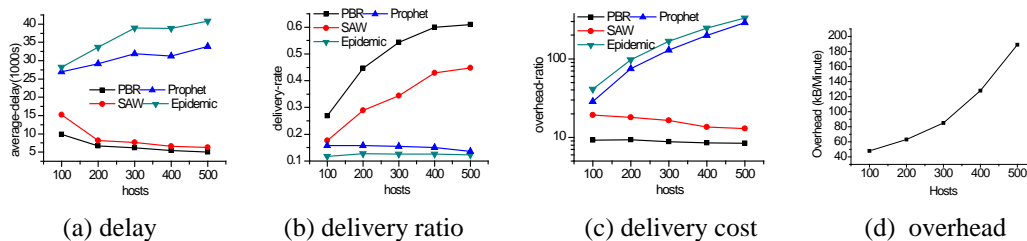
**Fig. 10.** Performance comparison of PBR-OppNet with different size of buffers

### 6.7 Network Size Influence on Performance

Finally, we evaluate performances in different network sizes. From **Fig. 11**, with the nodes increasing, network performances degrade in Epidemic and Prophet, which both have no limitation of copies. SprayAndWait and PBR-OppNet use the same copy limitation, so when the network scale expands and the node density increases, their performances are promoted. The delivery rate of PBR-OppNet is obviously higher than that of SprayAndWait, because PBR-OppNet avoids copying messages blindly. Furthermore, this advantage increases with network size expansion, since it increases the probability of waiting until meeting a node with better forwarding utility.

By the message delivery process, delivery delay is composed of the waiting delay for optimal forwarding nodes and the delay from the optimal forwarding node to the target node. This is the forwarding utility in our model. By comparing results of several methods, we can see that the PBR-OppNet and SprayAndWait have relatively small delivery delays, while the former not only has a smaller delivery delay but also has a higher delivery ratio and a lower delivery cost.

In **Fig. 11** (d), we examined the overhead introduced by control messages in the PBR - OppNet. When the number of nodes increases, the node density and meeting frequency increased, and the number of messages exchanged between encountered nodes also increases. There is a non-linear relationship between the overhead and node number. However, because of the prediction to reduce the blindness of moving forward, the average total cost to successfully deliver a message is low, as can be in **Fig. 11** (c).



**Fig. 11.** Performance comparison of PBR-OppNet under conditions of different scale of networks

## 7. Conclusion

This paper proposes PBR-OppNet, a prediction-based routing method in opportunistic networks. According to the ARIMA model, we first predicted the historical contacting interval of the nodes using a time series to obtain the contact time in the future as the forwarding utility.

Then we used the routing algorithm based on an optimal stopping rule to calculate the forwarding utility's threshold when observation is ceased and the message is delivered. This algorithm actually predicts the future forwarding utility according to the observed distribution of the past forwarding utility. Using a prediction model, PBR-OppNet alleviates the dynamics in opportunistic networks and the blindness when duplicating messages. It reduces the delay, improves the delivery success rate and obtains a relatively good comprehensive performance. From the perspective of predicting forwarding utility, the direction of future work is the time series analysis method, which considers the dependency between multiple links, and the method based on a complex network, which is about the network topology, and to mix various methods to improve prediction accuracy.

## References

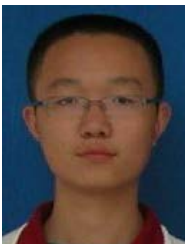
- [1] Spyropoulos Thrasylvoulos, Konstantinos Psounis, and Cauligi S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proc. of ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 252-259, August 22-26, 2005. [Article \(CrossRef Link\)](#)
- [2] Han Bo, Pan Hui, VS Anil Kumar, Madhav V. Marathe, Jianhua Shao, and Aravind Srinivasan, "Mobile data offloading through opportunistic communications and social participation," *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, pp. 821-834, September, 2012. [Article \(CrossRef Link\)](#)
- [3] Anna-Kaisa Pietiläinen, Earl Oliver, Jason LeBrun, George Varghese, and Christophe Diot, "MobiClique: middleware for mobile social networking," in *Proc. of 2nd ACM workshop on Online social networks*, pp. 49-54, August 17, 2009. [Article \(CrossRef Link\)](#)
- [4] Jung, Sewook, Uichin Lee, Alexander Chang, Dae-Ki Cho, and Mario Gerla, "Bluetorrent: Cooperative content sharing for bluetooth users," *Pervasive and Mobile Computing*, vol. 3, no. 6, pp. 609-634, December, 2007. [Article \(CrossRef Link\)](#)
- [5] Shi, Cong, Vasileios Lakafosis, Mostafa H. Ammar, and Ellen W. Zegura. "Serendipity: enabling remote computing among intermittently connected mobile devices," in *Proc. of 13th ACM international symposium on Mobile Ad Hoc Networking and Computing*, pp. 145-154, June 11-14, 2012. [Article \(CrossRef Link\)](#)
- [6] Liu Shu, and Aaron D. Striegel, "Exploring the potential in practice for opportunistic networks amongst smart mobile devices," in *Proc. of 19th annual international conference on Mobile computing & networking*, pp. 315-326, September 20-24, 2013. [Article \(CrossRef Link\)](#)
- [7] Lindgren A, Doria A, and Schelen O, "Probabilistic routing in intermittently connected networks," *Mobile Computing and Communications Review*, vol. 7, no. 3, pp.19-20, July, 2003. [Article \(CrossRef Link\)](#)
- [8] C. Liu and J. Wu, "An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks," in *Proc. of 10th ACM international symposium on Mobile Ad Hoc Networking and Computing*, pp. 105-114, May 18-21, 2009. [Article \(CrossRef Link\)](#)
- [9] P. Hui, J. Crowcroft and E. Yoneki, "Bubble Rap: Social-based Forwarding in Delay Tolerant Networks," in *Proc. of 9th ACM international symposium on Mobile Ad Hoc Networking and Computing*, pp. 241-250, May 18-21, 2008. [Article \(CrossRef Link\)](#)
- [10] W. Gao, Q. Li, B. Zhao and G. Cao, "Multicasting in delay tolerant networks: a social network perspective," in *Proc. of 10th ACM international symposium on Mobile Ad Hoc Networking and Computing*, pp. 299-308, May 18-21, 2009. [Article \(CrossRef Link\)](#)
- [11] David Liben Nowell and Jon Kleinberg, "The link-prediction problem for social networks," *Journal of the American society for information science and technology*, vol. 58, no. 7, pp. 1019-1031, July, 2007. [Article \(CrossRef Link\)](#)
- [12] Guimerà Roger, and Marta Sales-Pardo, "Missing and spurious interactions and the reconstruction of complex networks," in *Proc. of the National Academy of Sciences of the United States of*



- America*, pp. 22073-22078, December 29, 2009. [Article \(CrossRef Link\)](#)
- [13] T. S. Ferguson, *Optimal stopping and applications*, Online available: <http://www.math.ucla.edu/~tom/Stopping/Contents.html>, 2015.
- [14] Daqiang Zhang, Zhijun Yang, Vaskar Raychoudhury, Zhe Chen and Jaime Lloret, "An Energy-Efficient Routing Protocol Using Movement Trends in Vehicular Ad hoc Networks," *The Computer Journal*, vol. 56, no. 8, pp. 938-946, August, 2013. [Article \(CrossRef Link\)](#)
- [15] Daqiang Zhang, Min Chen, Mohsen Guizani, Haoyi Xiong and Daqing Zhang, "Mobility prediction in telecom cloud using mobile calls," *IEEE Wireless Communication*, vol. 21, no. 1, pp. 26-32, January, 2014. [Article \(CrossRef Link\)](#)
- [16] Daqiang Zhang, Daqing Zhang, Haoyi Xiong, Laurence, T. Yang and Vincent Gauthier, "NextCell: Predicting Location Using Social Interplay from Cell Phone Traces," *IEEE Transaction on Computers*, vol. 64, no. 2, pp. 452-463, February, 2015. [Article \(CrossRef Link\)](#)
- [17] Daqiang Zhang, Athanasios V. Vasilakos and Haoyi Xiong, "Predicting location using mobile phone calls," in *Proc. of ACM Special Interest Group on Data Communication*, pp. 295-296, August 13-17, 2012. [Article \(CrossRef Link\)](#)
- [18] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proc. of 2nd International Conference on Simulation Tools and Techniques*, pp. 1-10, March 2-6, 2009. [Article \(CrossRef Link\)](#)
- [19] Cabspotting dataset. [EB/OL]. Available: <http://crawdad.cs.dartmouth.edu/meta.php?name=epfl/mobility>, 2015.



**Sanfeng Zhang** received his Ph.D. degree in Computer Application Technology, from School of Computer Science & Engineering, Southeast University, Nanjing, China, in 2008. Since 2008, Dr. Zhang has been with the College of Software Engineering and School of Computer Science & Engineering, Southeast University, where he is currently an associate professor. His current research interests include network coding and wireless networks. Email: [sfzhang@seu.edu.cn](mailto:sfzhang@seu.edu.cn).



**Di Huang** was born in Jiangsu, China, in 1990. He is currently pursuing for the D. Eng. degree in Computer Science and Technology from Southeast University. His current research interests include opportunistic networks and computing offloading. Email: [seusofthd@gmail.com](mailto:seusofthd@gmail.com)



**Yin Li** was born in Hebei, China, in 1993. She is currently pursuing for the M. Eng. degree in Computer Science and Technology from Southeast University, Nanjing, China. Her current research interests include opportunistic networks and link prediction in complex networks. Email: [yinli@seu.edu.cn](mailto:yinli@seu.edu.cn).