Journal of Computers Vol. 31 No. 2, 2020, pp. 197-211
doi:10.3966/199115992020043102017

Opportunistic Mobile Network Routing Protocol Based on Multiple-level Social Network

Bi-Tao Peng[1*], Juan Wang[1]

[1] School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou 510006, China
pengbitao@gdufs.edu.cn, misisipiwj@126.com

Received 21 August 2018; Revised 1 January 2019; Accepted 6 March 2019

Abstract. Based on the mobile contact and storage-carry-forward model, opportunistic mobile network can realize communication between nodes. However, establishing an efficient routing protocol is one of the key problems in opportunistic mobile network research. Based on the sociality of the opportunistic mobile network, many researchers have proposed a routing protocol to select the next packet relay node. However, the social features exploited from the history contact information take time to collect. Moreover, the protocol is unstable and cannot reflect the global contact information in the future, whereas an online social network of nodes can exploit the relationship between nodes all around and it is a stable relationship. By analyzing the stable online social network graph and the relationship between the online social network and offline contact network, we propose an opportunistic mobile network routing protocol based on multiple-level social relationship that combines the online friendship between nodes, the interests of nodes, and offline contact information to select the next hop node for the packet. Simulation results show that the algorithm significantly improves the success rate of data packet transmission and reduces the transmission time compared with the classical algorithms.

Keywords: offline contact network, online social network, opportunistic network, routing protocol

1 Introduction

The opportunistic mobile network [1-4], a type of delay tolerant network (DTN) [5] that utilizes the nodes' mobile opportunistic communication and does not require a fixed communication infrastructure [6-7], has recently received more and more research attention [8-9]. In an opportunistic network, there is no reliable end-to-end path between the source node and the destination node, and its routing mechanism mainly relies on the cooperation of the mobile node to work in the "storage-carry-forward" mode [10-11]. Opportunistic mobile networks have many applications, such as the ZebraNet project for wildlife tracking [12], the DakNet project for Indian rural communications [13], the pocket-switched file-sharing network that makes use of nodes' movement through handheld devices [14], and the Haggle project [15] for content-sharing applications based on mobile devices.

In opportunistic mobile network applications, designing an efficient network routing protocol is one of the core issues; that is, a protocol must be established for how the mobile node forwards data to the encountering nodes to achieve as much data transmission as possible with minimal network overhead and short network delays. Opportunistic mobile networks communicate based on smart devices with short-range wireless communication capabilities. They use Bluetooth, Zigbee, and other networking methods to implement data exchange and provide network services. To a large extent, the nature of network operations depends on the owner of the device: people. Researchers guide the forwarding of data packets by acquiring social relationships between people and improve the efficiency of data transmission.

For example, various researchers [16-19] have analyzed the contact records and contact patterns

---

* Corresponding Author

between nodes to obtain the social relations—contact social relationships (CSRs), i.e., offline social relationships, to guide the forwarding of data packets. In the opportunistic mobile network, there is no central server, and the nodes are constantly moving, so the nodes cannot obtain the complete network structure, and the contact social relationships are incomplete and unstable. In addition to the offline social relations obtained in real time, there are many other ways to obtain social relationships of nodes. Social software such as Facebook, WeChat, Twitter, or QQ provide a good way to obtain the social relationships between nodes—online social relationships (OSRs), and OSRs are more stable than CSRs. A stable online social relationship includes a variety of social attributes such as friend relationships between nodes, interests and hobbies between nodes, professional careers of nodes, and residential communities. Abderrahmen et al.'s [20] study shows that the more information that is gathered, the better they can design the data packet forwarding strategy.

Although several researchers use both CSRs and OSRs to design the opportunistic routing strategy, they exploited only two layers social network. However, in this paper we will combine multiple-layer social network, and analyze the structural characters relationship between different layers. We will research the opportunistic network routing strategy combined with CSRs and OSRs which includes multiple social relationships of nodes. **The key research point of this work is to demonstrate whether it can improve the opportunistic mobile network routing efficiency using the information extracted from multiple-level social networks.** The main technical contributions of this paper are summarized as follows:

(1) We give the multiple-level social network model for an opportunistic mobile network and analyze the structural features.

(2) We propose efficient routing protocols for an opportunistic mobile network based on the multiple-level social network.

(3) We perform extensive simulations based on real traces. The results show that the proposed algorithms can achieve better performance than other classical algorithms.

The article is organized as follows: Section 2 gives the related research work of network routing protocols based on social relations in opportunistic mobile networks. Section 3 discusses the multiple-level social network model, related datasets, and network structure analysis. Section 4 gives the specific algorithm and implementation. Section 5 part sets up the experiment to carry on the simulation analysis. The last part makes the conclusion to the full text, and discusses the future work.

2 Related Research Work

Epidemic algorithm [21], based on data packet replication, is the simplest opportunistic mobile network routing algorithm. The algorithm copies the data packets to the encountered nodes and floods the data. The Spray and Wait scheme [22] also uses flooding to deliver messages, but it limits the amount of flooding. It forwards a fixed number of data packet copies to the encounter nodes. When the number of copies is one, it directly delivered to the destination node. The flooding-based algorithm has a higher transmission success rate, but the disadvantage is high resource consumption. In contrast to algorithms based on data packet replication, the Prophet algorithm [23], based on the historical path information, assumes that the node movement is nonrandom. If the node has historically recorded a certain area, the probability of passing through the area in the future will be higher. Based on this assumption, packets are always passed to nodes that have a greater probability of encountering the target node.

Because equipment is held by people, researchers consider the social attributes of the equipment holders in the mobile network and conduct research on the message forwarding strategies based on the social relationships of the nodes. In the study of forwarding algorithms based on social relations, there are mainly two categories—those based on offline social contact and those based on online social relationships.

Chaintreau et al. [24] and Karagiannis et al. [25] have analyzed the people contact rule. Chaintreau et al. found that the interval between nodes adheres to a power-law distribution. Karagiannis et al. found the same law but point out that, when certain values are exceeded, the data obey an exponential distribution. At the same time, other researchers have found that there are significant social rules in the activities of nodes, and there are community structures for nodes in the network [24-26]. By calculating the community and friend relationships of the nodes [27], the data packets are only forwarded to the nodes that are in the same community as the target nodes. In the same community, only the packets that have a stronger friend relationship with the target nodes are sent. Similarly, Bubble RAP [16] collects historical contact information of nodes, statistically analyzes the social relationships between nodes, and constructs the CSR of nodes. Finally, it designs the packet routing strategy. In Bubble RAP, by calculating the global centrality and local centrality of nodes, packets are always forwarded to nodes with greater global centrality before they reach the destination community; when data are grouped into the final destination community, they are forwarded to the node with a large local center until they are finally sent to the destination node. Liu et al. [26] have analyzed the MIT Reality dataset [28] and found that the complementary cumulative distribution function of the node contact time also conforms to a power-law distribution, indicating that node activity has significant sociality and that the nodes in the network have a community structure. On this basis, the backbone nodes of the opportunistic mobile network are also constructed through the mediation centrality to the design algorithm, and the backbone nodes serve as the relay nodes of the packet. This is another way of thinking about the design of a routing algorithm by predicting future contact opportunities of nodes through contact social relationships of nodes [29-30]. Ihler et al. [30] found that the contact patterns of nodes adhere to a Poisson distribution, use hidden Markov chains to predict the future contact of the nodes in time series, and guide data packets forwarded to nodes with greater chances of contact with future target nodes [30]. Based on the social contact information, Jedari et al. [31] proposed a social-based watchdog system to reduce the loss of network resources, thus improving data delivery performance. Their system uses watchdog nodes to analyze messages received from their encountered nodes with respect to their social tie information to identify the nodes' selfish behavior in message relaying.

By recording the contact data of the nodes, the contact social relationships can partly improve forwarding performance. However, there are uncertainties in the detection of social relationships, and it takes a long time for the network to detect these. With the popularity of social software, such as Facebook, WeChat, Twitter, QQ, etc., there is a stable online social network relationship among people. The more information that is mastered, the more conducive it is for analyzing the activity patterns between nodes. Therefore, the online social relationship of nodes has also been introduced into the study of mobile networks. Mtibaa et al. [32] deployed a Bluetooth-based mobile social network application. By analyzing the contact information of the nodes, they found that the friend relationship between the nodes helps to establish the forwarding strategy of packets. Hossmann et al. [33] further analyzed the node contact information, social relations, and data of nodes communicating with each other, revealing the following: (1) Online social relationships are positively related to node contact; i.e., online social relationships increase the chance of node contact. (2) Nodes' intercommunication and contact times are positively related; that is, if there are more opportunities for nodes to contact, they are more likely to communicate with each other. (3) Online social relationships and internode communications are positively related; for example, people prefer to communicate with friends when they are in contact. Based on the historical contact information of the node and the online social relationship of the node, Socievole et al. [34] computed the centrality of the node and the future contact probability between nodes. When the node is contacted, the packets are forwarded to a high utility value node. In 2016, Socievole et al. [35] presented a detailed analysis of a set of six different mobility traces for opportunistic network environments including nodes' Facebook friendships using a social network approach to analyze the similarity between online and offline networks. Similarly, using the online friend relationship of nodes, Bigwood and Henderson [36] used social network analysis methods to classify nodes into different roles and calculate the measurement distance between different roles. When two nodes contact each other, if the nodes belong to the same role or if the roles are close, the packet will be forwarded between them. Zeng et al. [37] considered the online friend relationship to establish clusters, and the data packets are always copied and forwarded to nodes in the same cluster as the target node. Inspired by Google's PageRank algorithm, Abderrahmen et al. [20] used the online social relationship of nodes from different perspectives and designed the PeopleRank algorithm. When two nodes contact, they calculated the Rank value of each node based on the online social relationship, and the data packets are forwarded to the node with high Rank value upon contact. Allen et al. [38] used social information concerning the interests of network nodes as well as the frequency of encounters with them to design a forwarding strategy. This information is collected and shared dynamically, as nodes initially encounter each other and exchange their preferences, and the forwarding of micro-blog updates is directed across the network.

In fact, there can are multiple relationships between nodes, such as friendship, common interests, and common affiliation, which are abstracted from networks and referred to as multiple-layer social networks [39]. A multiple-layer social network gives the relationships between the nodes from multiple different perspectives. Magnani and Rossi [39] proposed a multiple-layer social network model that includes a fixed set of nodes, and there are multiple sets of edges between the nodes. Piotr et al. [40] analyzed the cross-layer clustering coefficient, cross-layer degree centrality, and neighborhoods in multiple-layer social networks. They give five definitions of multiple-layered neighborhoods that consist of users whose activities result in multiple-layered relations. Based on these definitions, several network structural characteristics were defined and measured. From a collaborative filtering perspective, Chen et al. [41] observed the cross-dependency inference in a multiple-layered social network and performed extensive evaluations on real datasets to evaluate the superiority of the proposed approaches. Bindu et al. [42] proposed a pioneering approach called ADOMS (Anomaly Detection On Multilayer Social networks), an unsupervised, parameter-free, and network feature-based methodology, that automatically detects anomalous users in a multiple-layer social network and ranks them according to their anomalousness. It considers the two well-known anomalous patterns of clique/near-clique and star/near-star anomalies in social networks, and users are ranked according to the degree of similarity of their neighborhoods in different layers to stars or cliques. Although some research work has been done in the field of multiple-layer social networks, to the best of our knowledge, there has been little work on the opportunistic mobile network.

By combining the contact social relationship with the online social relationship, we can design algorithms to improve the data forwarding performance of the opportunistic mobile network. In fact, there can be multiple online network relationships between nodes, such as Facebook friends, WeChat friends, the same interests and hobbies, and so on. Therefore, in this article we will design multiple online social relationship networks and combine these with a contact social network to design a mobile network routing protocol.

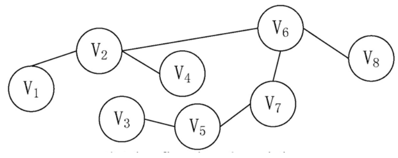
3 Multiple-level Social Network Models and Related Datasets

3.1 Multiple-level Social Network Models

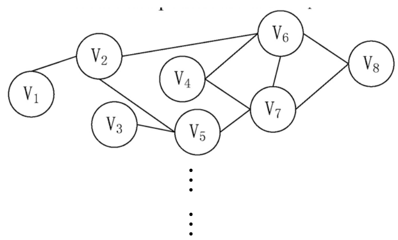
The multiple-level social network (MSN) can be defined as a triplet $<V, E, L>$, where $V=\{v_i \mid i=1,...,n\}$ represents a set of nodes, n represents the number of nodes, and $E=\{(v_i, v_j, l_k) \mid i!=j, i=1,..., n, j=1,...,n, k=1,...,m\}$ represents a set of edges. For any two triple elements $<v_i, v_j, l_k>$ and $<v_i', v_j', l_k'>$, if $v_i = v_i'$ and $v_j = v_j'$, then $l_k = l_k'$; that is, they are not in the same network layer. $L=\{l_k \mid k=1,...,m\}$ represents different layers of the network, and there are a total of m layers of networks.

In an MSN, owing to the existence of different social relationships among nodes, such as Facebook friends, WeChat friends, and common interests, we can construct one social network map according to one social relationship. In all levels of the social network, nodes are the same, but the edges between nodes are unique. Edge (v_i, v_j, l_k) indicates that there is such a relationship between nodes v_i and v_j in the k-level social relationship.

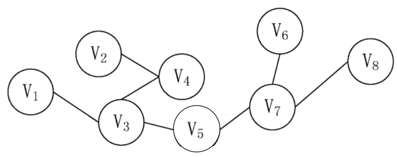
Fig. 1 shows an example of a model diagram for multiple social networks. The nodes of the set $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$ are connected on m different social network layers. In each layer, the edges connecting the nodes are different; this denotes the relationship between nodes in this layer. Through researching the structural feature of each layer, we can master the characteristics and understand the social behaviors of nodes in each layer. Through analyzing the relationships between different layers, and comparing the network features, we can get the characteristics and global rules.



(1) The first level social relationship network—MSN l_1



(2) The second level social relationship network—MSN l_2



(M) The m^{th} level social relationship network—MSN l_m

Fig. 1. A multi-level social relationship network

In this article, we will extract a multiple-layer social network from real datasets, analyze the constructural features, compare the centrality, and compute the correlation. We will then combine the multiple-layer social relationship network to design a message forwarding mechanism in the opportunistic mobile network. While two nodes meet, how do they select the next relay, and forward the data packet.

3.2 Datasets

SocialBlueConn [43] and Hyccups [44] are two open datasets in the field of multiple-level social relationship opportunistic mobile networks. They are available for download in the CRAWDAD data community.

The SocialBlueConn dataset contains Bluetooth device proximity data collected by an ad hoc Android application called SocialBlueConn. This application was used by 15 students at the University of Calabria campus in Rende, Cosenza, Italy, and logged both the internal contacts between the participants and contacts with 20 other external mobile nodes. The dataset includes the social profiles (Facebook friends and self-declared interests) of the participants.

The Hyccups dataset, collected at the University Politehnica of Bucharest in the spring of 2012, uses an application entitled HYCCUPS Tracer, with the purpose of collecting contextual data from Android smartphones. It was run in the background and collected availability and mobile interaction information, including information about a device's encounters with other nodes or with wireless access points. Encounter collection was performed using AllJoyn. The data were collected by constructing and deleting wireless sessions using the AllJoyn framework based on WiFi. The duration of the tracing experiment was 63 days, between March and May 2012, and had 72 participants. Similar to SocialBlueConn, the experimental participants' Facebook friends, interests, and other information are collected at the same time.

Table 1 lists the basic characteristics of the two datasets. Because both SocialBlueConn and Hyccups datasets contain online social relations and offline contact data, it is possible to construct a multiple-layered social relationship network. By analyzing the participants' Facebook profiles, the users' interests, and the offline contact data, the social connections relationship was extracted. Based on the results, we can design an opportunistic mobile network routing protocol to select the appropriate next relay nodes to transfer the data packet.

Table 1. Basic characteristics of the dataset

Experimental dataset	SocialBlueConn	Hyccups
Device type	Smart phone	Smart phone
Network type	Bluetooth	WI-FI
Duration (days)	7	63
Total contact times	20,100	8427
Number of devices	15	72

3.3 Network Structure Analysis

First, in this section we analyze the distribution and characteristics of data contacts in multiple-layer social relations and then analyze the correlation between online social relationships and offline social relationships. Finally, we analyze the network structure of multiple-layer social networks and explore the guiding relationship between online social networks and offline social networks.

3.3.1 Network Model

For the online network relationship, a network model is established based on the two types of social relationships of the node: (1) the friend relationship and (2) the interest relationship, which are, respectively, called the friend relationship network and the interest association network. In the network of friends, if two nodes are friends with each other, there are directly connected edges between the two nodes. In the interest association network, since the node has multiple different interests, all interest is first listed as a one-dimensional array, such as six interests $(I_1, I_2, I_3, I_4, I_5, I_6)$, with each node creating the corresponding interest vector. For example, if node v_x is interested in I_1, I_4, and I_5, the interest vector is (1 0 0 1 1 0). Finally, the cosine correlation between node interest vectors is calculated. If the cosine correlation between the two is 0, then there is no directly connected edge between the pair of nodes; otherwise, there is a connected edge between the pair of nodes, and the weight of the edge is the cosine correlation value, which represents the degree of interest relatedness.

Based on the results in [45], the network is constructed. The weight is the average duration of historical contact intervals for the nodes. Therefore, when constructing the offline contact network between nodes, in this paper we construct the contact network graph of the nodes. The contact interval time between the nodes is the weight of the edge. If there is no contact between the two nodes, the weight of the edge is infinite.

3.3.2 Correlation Analysis of Online Network Relationships and Offline Node Contact Times

Because the online network relationship represents a stable relationship between nodes, the offline contact network is dynamic and unstable. Therefore, does the topology structure of the online network have certain guiding significance for offline network contact? That is, do the nodes with the same hobby or nodes that are close friends more likely to touch each other in real contact? To explore these contents, we analyze the following: (1) the relevance of the node's friends and the number of contacts and (2) the relevance of the node's interests and the number of contacts.

Table 2 lists the friend relationship and contact number between the nodes, the Pearson correlation coefficient, and the Spearman rank correlation coefficient of the friend relationship and contact number. In denotes the degree of interest, F means friend, and C means contact number.

Table 2. Online network relationships and node contact times

	SocialBlueConn		Hyccups	
	In & C	F & C	In & C	F & C
Pearson	0.421053	0.314285	0.052221	0.041551
Spearman	0.402045	0.561595	0.561825	0.725711

As shown in Table 2, for both test datasets, there is a positive correlation between the number of contacts and interest or between the number of contacts and friends. That is, if two people have the same interests or they are friends, the number of contacts will be greater.

3.3.3 Correlation Analysis of the Centrality of the Online Network Relationship Diagram and the Offline Contact Network Diagram

In the network diagram, centrality measures the association degree between the node and the other nodes in the graph, which is an important indicator of the importance of the node. In the opportunistic network, if the centrality of the node is high, this means that the node has more opportunities to contact with other nodes; when the data packets are forwarded between nodes, the message is forwarded to the node with high centrality to increase the success rate of packet delivery. The relationship between the online network and the offline contact network is analyzed by comparing the node centrality of the online interest network, the friend relationship network, and the offline contact network. In the offline contact network, the average contact interval of the node is its weight. In the online interest network, the weight correlation is opposite to the interest correlation between the nodes, which is 1 minus its interest correlation. In the online friend relationship, the weight between nodes with a friend relationship is 1; otherwise, the value is infinity.

Table 3. Centrality relevance of online network relationship and offline contact network

	SocialBlueConn				Hyccups			
	In & C		F & C		In & C		F & C	
	CC	EC	CC	EC	CC	EC	CC	EC
Pearson	0.590612	0.666241	0.537559	0.373500	0.234949	0.214947	0.342749	0.357558
Spearman	0.353571	0.532143	0.461607	0.464286	0.303931	0.288134	0.442784	0.371383

The analysis in Table 3 shows that, for the two test datasets, both the centrality of the online friend graph and the centrality of the contact network graph as well as the centrality of the interest correlation graph and the centrality of the contact network graph are significantly positively correlated. That is, if someone is in an important position in the friend relationship diagram or the interest association diagram, that person is also in an important position in the contact network diagram.

4 Message Transmission Algorithm Based on a Multiple-layer Social Relationship

Offline contact information can effectively guide data packet forwarding between nodes; however, offline contact information has some limitations. On the one hand, it belongs to historical information. On the other hand, it only contains contact information of some nodes, and there is no global contact information. According to the analysis in the previous section, the online social relationship of nodes based on interest and friend relationship has a strong positive correlation with the offline contact social relationship between the contact strength and the global centrality of the nodes. Moreover, the online social relationship is relatively stable. Therefore, we combined the offline contact information and the stable online social relationship to design the message forwarding strategy and guide the forwarding of data packets between nodes.

4.1 Algorithm

4.1.1 Offline Contact Social Relationship Parameters

By analyzing the actual datasets, Lindgren et al. [23] found that the contact interval time of the two nodes satisfies an exponential distribution. The node pairs of each contact in the experimental dataset were tested by using a chi-square test to determine whether the contact time interval between the pair of nodes obeys an exponential distribution. The experimental results show that, when the test interval is large enough, the contact time interval is follows an exponential distribution. In the literature [24, 27], the contact interval of the two nodes is also considered to satisfy an exponential distribution. Based on the above research results, we also assume that the contact time interval between pairs of nodes in the opportunistic mobile network obeys an exponential distribution. Therefore, the contact between any two nodes v_i and v_j becomes a homogeneous Poisson process, in which the contact frequency is λ_{ij}, so the probability density function of X is

$$f(x) = \begin{cases} \lambda_{ij} e^{-\lambda_{ij} x}, x \geq 0 \\ 0, x < 0 \end{cases}. \quad \textbf{(1)}$$

It is assumed that the expected contact interval time is E_{ij}, which is equal to $1/\lambda_{ij}$. In each node v_x, there is a data field-node contact record table, which records the contact time points with other nodes. According to the node contact record table, each node can calculate the average contact interval time between node v_i and other nodes; we set E_{ij} as the average contact interval time. When node v_i meets node v_j, the two nodes record the contact time and modify their node contact record tables, respectively. Through this operation, each node in the network can calculate the expected contact time interval with other nodes. When the network is initially running, a default initial value is given to the average expected contact time of each pair of nodes. As time progresses, different nodes meet each other, record their contact time, and calculate the expected contact interval time according to the historical dynamic contact information.

4.1.2 Comprehensive Social Relationship Parameters

The number of contacts between nodes reflects the contact strength of the node. According to the above analysis, there is a strong positive correlation between the number of contacts and the friend relationship of the node and the interest similarity between the nodes. On the one hand, if the nodes are friends, the number of contacts between the nodes is significantly more than that of non-friend nodes. On the other hand, if there are more same interests between the nodes, the number of contacts is also higher. At the same time, in terms of centrality, the relationship between the topological graph of friends and the contact topological graph is obviously consistent at the central degree of the node. The topology graph established based on the interest similarity between nodes is obviously consistent with the node centrality of the offline contact network. Therefore, we predict offline contact with network nodes based on an online stable social network to guide the data packets transfer.

Based on the friend relationships between nodes, we define the relationship between nodes v_i and v_j as

$$VF(v_i, v_j) = \begin{cases} 1, & v_i, v_j \text{ is friend} \\ 0, & \text{else} \end{cases}. \quad \textbf{(2)}$$

Based on the interest relationship between nodes, we use the cosine similarity of the interest vector as the relationship value between the nodes v_i and v_j, which is

$$VI(v_i, v_j) = \sum_{b=1}^{T} (v_{ib} * v_{jb}) / (\sum_{b=1}^{T} v_{ib}{}^2 * \sqrt{\sum_{b=1}^{T} v_{jb}{}^2}). \quad \textbf{(3)}$$

where v_{ib} represents the value of node v_i on the *b*th interest, and, if there is such interest, the value is 1; otherwise, it is 0.

Finally, the comprehensive relationship between the two nodes combines the above three subrelationship values, and the combined result is

$$VC(v_i, v_j) = \frac{1}{E_{ij}} * (1 + VF_{ij} + VI_{ij}). \quad \textbf{(4)}$$

4.2 Comprehensive Algorithm

When two nodes v_i and v_j meet, they exchange relevant control information and data packets by calculating the above node relationship values to guide message forwarding. In the entire data forwarding process, the main data structure in the node is as follows:

(1) Node contact record list. Each item of the data structure includes (v_{id} , historical contact time point).

(2) Expected contact interval time table between nodes. Each item of the data structure includes (v_{id} , expected time interval), and the data structure is calculated from the node contact record list.

(3) Node interest list.

(4) Node friend relationship list.

The specific steps are as follows:

(1) Nodes v_i and v_j exchange their node lists.

(2) Nodes v_i and v_j exchange their friend relationship lists.

(3) Node v_i judges each data packet in the storage space. If the target node of the data packet is v_j , v_i delivers it to node v_j and deletes the packet.

(4) Node v_i updates the internode contact information list, adds the contact time with node v_j to the corresponding entry, and updates the expected time distance E_{ij} with node v_j .

(5) For each data packet stored in node v_i, the comprehensive relationship value of node v_i and the target node v_d of the message and the comprehensive relationship values of nodes v_j and v_d are, respectively, calculated, and the values of the two are compared. If $VC(v_i, v_d) < VC(v_j, v_d)$ and node v_j can store the data packet, the data packet is forwarded to v_j .

The pseudo code of major operations for node v_i is listed as Fig. 2.

5 Experiment

In this section, we employ an opportunistic network routing based on multiple-layer social relationship (ORMLR) algorithm in an opportunistic network environment [46]. For datasets, we use SocialBlueConn and Hyccups datasets; both datasets contain offline contact data, online friend relations, and hobby information. We use ORMLR as a data packet forwarding mechanism for simulation, and the simulation results are compared with the two classical algorithms: Epidemic algorithm and Directly delivery algorithm. The experiment compares the number of data packets successfully transmitted under different routing protocols and the data packet transmission delay, which are two important indicators for measuring the performance of the opportunity routing algorithm.

5.1 Impact of Different Data Packet Size

We first change the data packet size to evaluate the performance, we set the parameters as follows:

In the experiment for the SocialBlueConn dataset, we set the node cache space to 10M, selected data packet sizes as 1k, 1k-10k, 20k-50k, 50k-100k, 100k-200k, etc., and selected one of each interval for each experiment. We randomly generated the data packet size from the selected interval. Because the SocialBlueConn project lasted for 8 days, setting a data packet every 100 s, 7100 data packets were generated, and the data packet lifetime was set to 700,000 s.

Notation:

- ✧ CRT_i [v_x, {t_{ix}}]: A stored contact record table in node v_i, where v_x and {t_{ix}} are a node id and a set of contact times between nodes v_i and v_x, respectively.
- ✧ ETT_i [v_x, ET_{ix}]: An expected contact interval time table in node v_i, where v_x and ET_{ix} are a node id and the expected contact interval time between nodes v_i and v_x, respectively.
- ✧ IL_i [int_k]: A stored interest list in node v_i, where int_k are the interests in which node v_i is interested.
- ✧ FL_i [v_x]: A stored friend relationship list in node v_i, where v_x is the node that is the friend with node v_i.
- ✧ SPT_i [pkt_m, v_d]: A stored packet table in node v_i, where pkt_m and v_d are the data packet stored in node v_i and v_d is the destination node id of data packet pkt_m, respectively.
- ✧ ASS_i: The available storage space in node v_i.
- ✧ $size_m$: The size of data packet pkt_m.

◆ When v_i encounters v_j:

    Exchange IL_i [int_k] and IL_j [int_k];
    Exchange FL_i [v_x] and FL_j [v_x];
    For $pkt_m \in SPT_i$
            if $v_d = v_j$;
                Deliver data packet pkt_m to node v_j;
        Remove data packet pkt_m from node v_i.
                ASS_i := ASS_i + $size_m$;
    For $pkt_m \in SPT_i$
        Compute $VC(v_i, v_d)$ and $VC(v_j, v_d)$
        if $VC(v_i, v_d) < VC(v_j, v_d)$ and $ASS_j > size_m$
          Transfer data packet pkt_m to node v_j
          Remove data packet pkt_m from node v_i
          ASS_i := ASS_i + $size_m$;
          ASS_j = ASS_j - $size_m$;

Fig. 2. Pseudo code of major operations for node v_i

In the experiment for the Hyccups dataset, we set the same buffer space and interval of data packet size. The data packet size was randomly generated from the selected interval. The Hyccups project lasted for 63 days. Therefore, the data packet generation frequency was set to 1000 s to generate a data packet, and 5400 data packets were generated, and the lifetime of the data packet was set to 5,000,000 s.

Fig. 3 shows the number of data packets successfully delivered to the target node using the three routing algorithms Directly, Epidemic, and ORMLR), respectively, under different data packet sizes. Fig. 4 shows the average time it takes to successfully deliver a packet using the three algorithms under different data packet sizes. The dotted lines in the figure represent the results of the Directly algorithm, the dashed lines indicate the results obtained by the Epidemic algorithm, and the solid lines indicate the results obtained by the ORMLR algorithm.

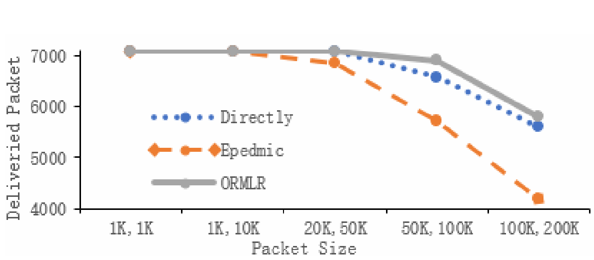


Fig. 3. Deliveried Packets Count

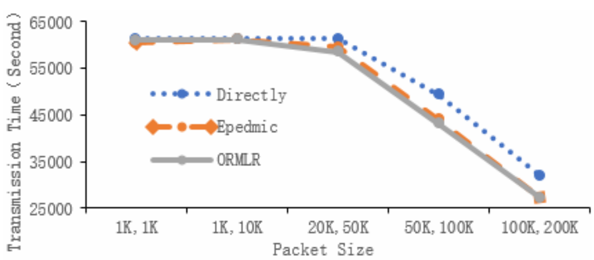


Fig. 4. Average Transmission Time

As shown in Fig. 3, when the packet size is small, the number of successfully delivered data packets obtained by the three algorithms is not much different. In actual data, the number of successful deliveries by the Epidemic algorithm is slightly greater than that of the other two algorithms. This is because, when the data packet size is small, no data packets are deleted owing to insufficient node cache space throughout the experiment, so Epidemic uses the infectious route, yielding the highest number of data packets being successfully delivered. At the same time, because the contact is frequent, the other two routing algorithms can also get better results. As the number of data packets increases, the number of packets successfully delivered becomes fewer and fewer, but the effects of different algorithms differ significantly. When the data packet becomes larger, Epidemic results begin to drop significantly, becoming lower than the results of the other two algorithms. This is because the Epidemic algorithm causes the data packets to be extruded frequently. ORMLR achieved the best results owing to the implementation of the guided group forwarding strategy based on multiple-layer social relationships. In Fig. 4, when the data packet size is small, the packet can always be cached in the node until it is finally delivered to the target node, so the average successful transmission time is long; when the data score size becomes larger, the average successful transmission time decreases. The average successful transmission time obtained by ORMLR is the least of the three routing algorithms.

Fig. 5 and Fig. 6 show the test results for the Hyccups dataset. Fig. 5 shows the number of data packets successfully delivered to the target node using Directly, Epidemic, and ORMLR routing algorithms under different data packet sizes. Fig. 6 shows the average time it takes to successfully transfer a packet using the three algorithms under different data packet sizes. In Fig. 5, when the data packet size is small, the Epidemic algorithm achieves the best results in terms of the number of successful deliveries, and the Directly algorithm yields the worst results. When the data packet becomes larger, the number of Directly successful deliveries trivially decreases, but the numbers of successful ORMLR and Epidemic deliveries decrease significantly, because the cache constraint has little effect on the Directly algorithm. At the same time, under the guidance of the multi-node social relationship, ORMLR has achieved the most successful deliveries. In Fig. 6, the Directly algorithm has the longest average transmission time, and when the node's data packet becomes larger, the average successful delivery time of the data packet eventually becomes shorter, and the packet forwarding is guided, and ORMLR achieved the best results.

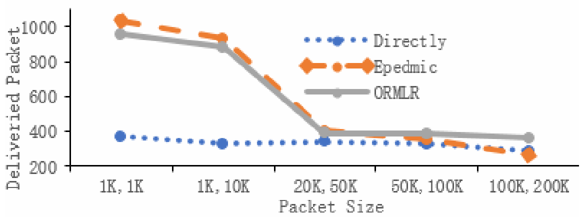


Fig. 5. Deliveried Packets Count

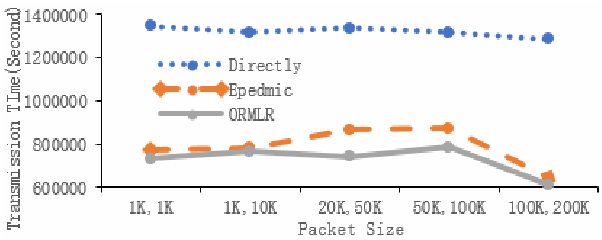


Fig. 6. Average Transmission Time

5.2 Impact of Data Packet Generation Rate

In the second part, we change the data packet generation rate to evaluate the performance. We set the parameters as follows:

In the experiment for the SocialBlueConn dataset, we set the node cache space to 10M and selected the data packet size as 1k-10k. We randomly generated the data packet size from the selected interval. Because the SocialBlueConn project lasted for 8 days, we set the data packet lifetime to 700,000 s. We set the data packet generating interval times to 100, 200, 300, 400, and 500 s, respectively.

In the experiment for the Hyccups dataset, we set the same buffer space and interval of data packet size. The data packet size was randomly generated from the selected interval. Because the Hyccups project last for 63 days, we sett the data packet lifetime to 5,000,000 s. We set the data packet generation interval times to 1000, 2000, 3000, 4000, and 5000 s, respectively.

Fig. 7 shows the number of data packets successfully delivered to the target node using the three routing algorithms (Directly, Epidemic and ORMLR), respectively, under different packet generation rates. Fig. 8 shows the average transmission time it takes to successfully deliver a data packet. As shown in Fig. 7, with the increase of the data packet interval, the total number of data packets created decreases, and the number of successfully delivered data packets decreases accordingly. In Fig. 8, with the increase of the data packet interval, the transmission time decreases. In both figures, the Directly algorithm gets the worst results compared to the two other algorithms. Fig. 9 and Fig. 10 show the test results for the Hyccups dataset under different packet generation intervals. Fig. 9 shows the number of data packet successfully delivered using the three routing algorithms. Fig. 10 shows the average transmission time it takes to successfully deliver a data packet. In Fig. 9, with the increase of the data packet generation interval, the number of successfully delivered data packets decreases, but Epidemic performed better than the two other algorithms. In Fig. 10, the performance of ORMLR is better than that of the two other algorithms. With the increase of the packet generation interval, the number of packets is decreased, so Epidemic can get the best result for the sufficient store space. For both datasets, ORMLR can get results that are very close to those of Epidemic. However, for the average transmission time, it achieves the best performance. Because of the proposed routing algorithm, the node can select the suitable next node to relay the data packet, which can efficiently decrease the transmission time.

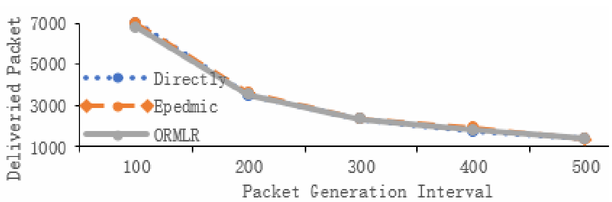


Fig. 7. Deliveried packets count

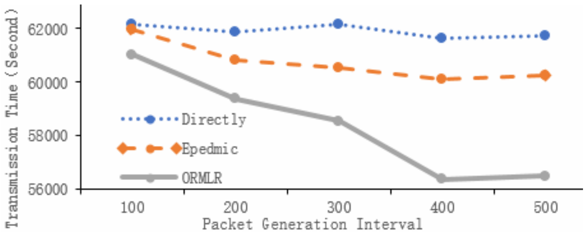


Fig. 8. Average transmission time

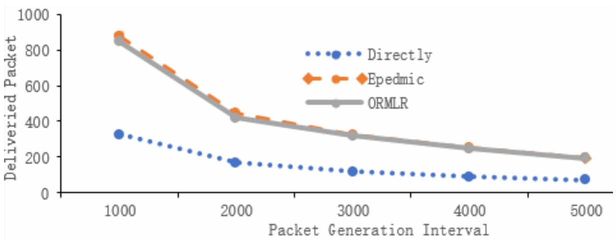


Fig. 9. Deliveried packets count

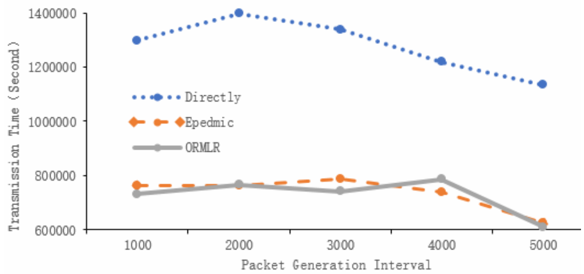


Fig. 10. Average transmission time

6 Conclusion

To effectively guide data packet forwarding in an opportunistic network and improve routing performance, we presented ORMLR, which combines online social relationships with offline social relations to forward data packets. First, we proposed a multiple-level social network model that describes different levels of social relationship between nodes. Based on the datasets SocialBlueConn and Hyccups, which include the friend relationship, interest relationship, and node contact relationship, we designed network relationship graphs and analyzed the correlation of several important factors for the different levels of relationships. The analysis shows a positive correlation between the number of contacts and the friend relationship and interest relationship and a positive correlation between centrality at different network layers. Then we evaluated the performance of ORMLR through experiments. The simulation results show that, compared with the classic Epidemic and Directly routing algorithms, the ORMLR routing algorithm can effectively improve the successful delivery rate of data packets and reduce the average delay of packets.

However, owing to the small number of social relationship opportunistic network datasets, the effectiveness of this algorithm has yet to be verified on more opportunistic network datasets, so the results of this work need further validation. As future work on the routing protocol, we would promote two items. First, the routing protocol that we considered in this work paves the way for further research; the community structure, backbone nodes, and other social characters of networks will be introduced to gain insight into the performance potential of the multiple-level social network routing protocol. Second, we will explore different opportunistic mobile network paradigms, such as publish/subscribe, to design a routing protocol based on a multiple-level social network.

Acknowledgments

This research is supported by the Guangdong University of Foreign Studies Innovation Project (No. 15T26) and The National Social Science Fund of China (No. 18BTQ065).

References

[1] L. Pelusi, A. Passarella, M. Conti, Opportunistic networking: data forwarding in disconnected mobile ad hoc networks, Communications Magazine 44(11)(2006) 134-141.

[2] S. Pal, Evaluating the impact of network loads and message size on mobile opportunistic networks in challenged environments, Journal of Network and Computer Applications 81(C)(2017) 47-58.

[3] Z. Mao, Y. Jiang, G. Min, S. Leng, X. Jin, K. Yang, Mobile social networks: Design requirements, architecture, and state-of-the-art technology, Computer Communications 100(1)(2017) 1-19.

[4] L.F. Liu, R. Wang, J.G. Wu, On the adaptive data forwarding in opportunistic underwater sensor networks using GPS-free mobile nodes, Journal of Parallel and Distributed Computing 122(2018) 131-144.

[5] A.P. Silva, S Burleigh, C.M. Hirata, K. Obraczka, A survey on congestion control for delay and disruption tolerant networks, Ad Hoc Networks 25(Part. B)(2015) 480-494.

[6] N. Wang, J. Wu, Rethink data dissemination in opportunistic mobile networks with mutually exclusive requirement, Journal of Parallel and Distributed Computing 119(2018) 50-63.

[7] A. Socievole, E. Yoneki, F.D. Rango, J. Crowcroft, ML-SOR: message routing using multi-layer social networks in opportunistic communications, Computer Networks 81(2015) 201-219.

[8] B.K. Saha, S. Misra, S. Pal. SeeR: simulated annealing-based routing in opportunistic mobile networks, IEEE Transactions on Mobile Computing 16(10)(2017) 2876-2888.

[9] T. Liu, Y. Zhu, R. Jiang, B. Li, A sociality-aware approach to computing backbone in mobile opportunistic networks, Ad Hoc Networks 24(Part. A)(2015) 46-56.

[10] Y. Li, J.H. Yu, X.H. You, Resource-limited opportunity network node incentive strategy research, Journal of Computer 5(36)(2013) 947-956.

[11] X. Chen, C. Shang, B. Wong, W.Z. Li, S. Oh, Efficient multicast algorithms in opportunistic mobile social networks using community and social features, Computer Networks 111(2016) 71-81.

[12] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.S. Peh, D. Rubenstein, Energy-efficient computing for wild life tracking: design tradeoffs and early experiences with Zebranet, in: Proc. 2002 International Conference on Architectural Support for Programming Languages and Operations Systems, 2002.

[13] A.A. Hasson, R. Fletcher, A. Pentland, DakNet: a road to universal broadband connectivity, in: Proc. Wireless Internet UN ICT Conference Case Study, 2003.

[14] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, C. Diot, Pocket switched networks and human mobility in conference networks, in: Proc. 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking, 2005.

[15] E. Nordstrom, C. Rohner, P. Gunningberg, Haggle: opportunistic mobile content sharing using search, Computer Communications 48(2014) 121-132.

[16] P. Hui. J. Crowcroft, E. Yoneki, Bubble rap: social-based forwarding in delay-tolerant network, IEEE Transaction on Mobile Computing 10(11)(2011) 1576-1589.

[17] Y. Wang, J. Wu, M. Xiao, Hierarchical cooperative caching in mobile opportunistic social networks, in: Proc. 2014Global Communications Conference (GLOBECOM), 2014.

[18] Z.F. Wang, P.T. Shi, S. Deng, H.B. Huang, Y.H. Wu, Optimal transmission strategy of opportunity network based on node social features, Journal of Communications 37(6)(2016) 163-168.

[19] M. Mordacchini, A. Passarella, M. Conti, A social cognitive heuristic for adaptive data dissemination in mobile opportunistic networks, Pervasive and Mobile Computing 42(2017) 371-392.

[20] A. Mtibaa, M. May, C. Diot, M. Ammar, PeopleRank: social opportunistic forwarding, in: Proc. 29th IEEE International Conference on Computer Communications, 2010.

[21] A. Vahdat, D. Becker, Epidemic routing for partially connected ad hoc networks. Technical Report CS-2000-06.

[22] T. Spyropoulos, K. Psounis, C.S. Raghavendra, Spray and wait: an efficient routing scheme for intermittently connected mobile networks, in: Proc. 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, 2005.

[23] A. Lindgren, A. Doria, O. Shelen, Probabilistic routing in intermittently connected networks, ACM SIGMOBILE Mobile Computing and Communications Review 7(3)(2003) 19-20.

[24] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, J. Scott, Impact of human mobility on opportunistic forwarding algorithms, IEEE Transactions on Mobile Computing 6(6)(2007) 606-620.

[25] T. Karagiannis, J.Y.L. Boudec, M. Vojnovic, Power law and exponential decay of inter contact times between mobile devices, IEEE Transactions on Mobile Computing 9(10)(2010) 1377-1390.

[26] T. Liu, Y. Zhu, R. Jiang, B. Li, A sociality-aware approach to computing backbone in mobile opportunistic networks, Ad Hoc Networks 24(2015) 46-56.

[27] E. Bulut, B.K. Szymanski, Friendship based routing in delay tolerant mobile social networks, in: Proc. 2010 Global Telecommunications Conference (GLOBECOM 2010), 2010.

[28] N. Eagle, A.S. Pentland, CRAWDAD data set mit/reality(v.2005-07-01). <http://crawdad.cs.dartmouth.edu/mit/reality>, 2005.

[29] C. Boldrini, M. Conti, F. Delmastro, A. Passarella, Context- and social-aware middleware for opportunistic networks, Journal of Network and Computer Applications 33(5)(2010) 525-541.

[30] A. Ihler, J. Hutchins, P. Smyth, Learning to detect events with Markov-modulated Poisson processes, ACM Transactions on Knowledge Discovery from Data 1(3)(2007) 13-35.

[31] B. Jedari, F. Xia, H. Chen, S.K. Das, A. Tolba, Z. Al-makhadmeh, A social-based watchdog system to detect selfish nodes in opportunistic mobile networks, Future Generation Computer System 92(2019) 777-788.

[32] A. Mtibaa, A. Chaintreau, J. Lebrun, E. Oliver, a.-K. Pietilainen, C. Diot, Are you moved by your social network application? in: Proc. 2008 the First Workshop on Online Social Networks, 2008.

[33] T. Hossmann, G. Nomikos, T. Spyropoulos, F. Legendre, Collection and analysis of multi-dimensional network data for opportunistic networking research, Computer Communications 35(13)(2012) 1613-1625.

[34] A. Socievole, F.D. Rango, S. Marano, Face-to-face with facebook friends: using online friendlists for routing in opportunistic networks, in: Proc. 2013 IEEE 24 International Symposium on Personal, Indoor and Mobile Radio Communication, 2013.

[35] A. Socievole, F.D. Rango, A. Caputo, Opportunistic mobile social networks: from mobility and Facebook friendships to structural analysis of user social behavior, Computer Communications 87(2016) 1-18.

[36] G. Bigwood, T. Henderson, Bootstrapping opportunistic networks using social roles, in: Proc. 2011 IEEE International Symposium on a World of Wireless and Multimedia Network, 2011.

[37] F. Zeng, N. Zhao, W. Li, Effective social relationship measurement and cluster based routing in mobile opportunistic networks, Sensors 17(5)(2017) 1109-1127.

[38] S.M. Allen, M.J. Chorley, G.B. Colombo, E. Jaho, M. Karaliopoulos, I. Stavrakakis, R.M. Whitaker, Exploiting user interest similarity and social link for micro-blog forwarding in mobile opportunistic networks, Pervasive and Mobile Computing 11(2014) 106-131.

[39] M. Magnani, L. Rossi, The ml-model for multi-layer social networks, in: Proc. 2011 International Conference on Advances in Social Networks Analysis and Mining, 2011.

[40] B. Piotr, K. Przemysław, M. Katarzyna, S. Krzysztof, Analysis of neighborhoods in multi-layered dynamic social networks, International Journal of Computational Intelligence Systems 5(3)(2012) 582-596.

[41] C. Chen, H.H. Tong, L. Xie, L. Ying, Q. He, Cross-dependency inference in multi-layered networks: a collaborative filtering perspective, in: Proc. 2017 ACM Transactions on Knowledge Discovery from Data (TKDD), 2017.

[42] P.V. Bindu, P.S. Thilagam, D. Ahuja, Discovering suspicious behavior in multilayer social networks, Computers in Human Behavior 73(2017) 568-582.

[43] A. Caputo, A. Socievole, F.D. Rango, CRAWDAD dataset unical/socialblueconn (v. 2015 02 08). <http://crawdad.org/unical/socialblueconn/20150208>, 2015.

[44] R.I. Ciobanu, C. Dobre, CRAWDAD dataset upb/hyccups (v. 2016 10 17). <http://crawdad.org/upb/hyccups/20161017>, 2016.

[45] H. Zhou, J. Chen, J. Fan, Y. Du, S. Das, Consub: incentive-based content subscribing in selfish opportunistic mobile networks, IEEE Journal On Selected Areas In Communications 31(9)(2013) 669-679.

[46] A. Keranen, J. Ott, T. Karkkainen, The one simulator for DTN protocol evaluation, in: Proc. 2009 International Conference on Simulation Tools and Techniques, 2009.