Front. Electr. Electron. Eng. China 2007, 2(3): 260–267
DOI 10.1007/s11460-007-0048-1

RESEARCH ARTICLE

GUO Yang, QU Yugui, BAI Ronggang, ZHAO Baohua

Contact modeling and prediction-based routing in sparse mobile networks

Abstract Mobile ad-hoc networks (MANETs) provide highly robust and self-configuring network capacity required in many critical applications, such as battlefields, disaster relief, and wild life tracking. In this paper, we focus on efficient message forwarding in sparse MANETs, which suffers from frequent and long-duration partitions. Asynchronous contacts become the basic way of communication in such kind of network instead of data links in traditional ad-hoc networks. Current approaches are primarily based on estimation with pure probability calculation. Stochastic forwarding decisions from statistic results can lead to disastrous routing performance when wrong choices are made. This paper introduces a new routing protocol, based on contact modeling and contact prediction, to address the problem. Our contact model focuses on the periodic contact pattern of nodes with actual inter-contact time involved, in order to get an accurate realization of network cooperation and connectivity status. The corresponding contact prediction algorithm makes use of both statistic and time sequence information of contacts and allows choosing the relay that has the earliest contact to the destination, which results in low average latency.

Simulation is used to compare the routing performance of our algorithm with three other categories of forwarding algorithm proposed already. The results demonstrate that our scheme is more efficient in both data delivery and energy consumption than previously proposed schemes.

Keywords contact modeling, prediction, routing protocol, sparse mobile ad-hoc network

1 Introduction

In one common sparse mobile ad-hoc networks (MANETs) scenario, nodes are mobile with wireless communication capabilities. Since the transmission range is much less than the network radius, most of the time, only fragments of the network are connected, and topology of the network is changing frequently because of the nodes' mobility.

This paper studies the problem of efficient data forwarding in sparse MANETs. Both sparse distribution of nodes and high ratio of speed to communication range can increase the period and frequency of partitions in such kind of network. This makes it a typical delay tolerant scenario [1] which determines the sparse MANETs routing works in an extended store-and-forward way. Messages may have to be buffered for a long time by relays, and moving nodes function not only as a source or a destination but also as a data link with high delay during a transfer. In this situation, contacts become the basic way of communication between nodes instead of settled links in traditional ad hoc networks. Just like traditional ad-hoc networks can be abstracted as an aggregation of nodes and fixed links among them, sparse MANETs can be described as an aggregation of mobile nodes and contact sequences with time stamps [2,3]. Due to the lack of knowledge of the nodes' contact dynamics, routing becomes a very challenging problem in which the traditional ad-hoc network forwarding tends to be not working.

Many existing approaches make use of the statistic information in the mobility of contact patterns of nodes. These approaches implying the statistic patterns in the node's historical contact information can be exploited to estimate the node's connectivity to a specific destination. Either mobile patterns [3] or the encounter probability [4] are used to help find the best relay node in statistic sense.

In this paper, we examine how contact patterns with time sequence information can be modeled and exploited to improve routing performance in sparse MANETs and propose a contact prediction-based routing protocol (CPRP). In typical sparse MANETs, nodes exhibit some degree of regularity in their contact patterns. For example, a bus traveling on a road is likely to arrive at the stations by some kind of order with probably constant intervals, and a student is likely to meet the math teacher every Tuesday morning in the semester when he has math classes. By exploiting these non-random

Received September 8, 2006; accepted March 13, 2007

GUO Yang, QU Yugui, BAI Ronggang, ZHAO Baohua (✉)
Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China
E-mail: ianyao@mail.ustc.edu.cn

contact patterns, we can predict the state of the nodes' contacts and network connectivity in the future. The basic idea of the CPRP is using contact prediction to help choose the node as a relay, which has the earliest contact to destination. This is a partial greed strategy that improves the message delivery performance. In the meantime, compared to flooding as many copies over relays as possible, only one carrier with the origin message exists in the network and also reduces the network overhead to a small constant.

The rest of this paper is structured as follows. Section 2 introduces the related work. The contact prediction-based protocol (CPRP) is presented in Sect. 3 in detail. The simulation results are shown in Sect. 4. Section 5 concludes this paper.

2 Related work

Routing in such a kind of delay tolerant MANETs has been studied by a growing number of researchers. The Delay Tolerant Network Research Group[1] has proposed an architecture to support messaging that could be used by some wireless mobile applications in such a context. Kevin Fall [1] proposed knowledge oracle-based routing in DTN, also called Deterministic Routing, which defines several oracles to provide different information of the network on which minimal-cost path selection relies. When no oracle is available, flooding-based protocols like epidemic routing [5], which have the highest probability (or least latency) and the most overheads in terms of bandwidth and energy [6], is a possible solution. On the contrary, the source in first-contact protocols [7] generates and holds the message till it comes in contact with the destination. In this scheme, only one copy of a specific message exits in the network before it reaches the destination or discarded. Obviously, it takes low overhead but may incur considerable delays.

Most of the work concerning routing in sparse MANETs has been performed within the network architectures in which nodes are not moving extremely randomly, such as the protocol presented by Timur Friedman et al. [3], which assumes that nodes contacting some specific nodes is long-duration stationary. These contacts form a virtual space, and the two nodes with similar sets of contacts are considered close in the space. Similarly, in Refs. [4,8], the authors use the delivery probability calculated by historical contacts to indicate which relay is to forward the message so that the destination will get a high chance of receiving it.

Also out of interest, work by Jain et al. [9] on inter-village communication and work by Akyildiz [10] on inter-planetary networking are both schemes with scheduled contacts and try to improve the connectivity between villages/planets within the sparse MANETs architecture.

3 Contact prediction-based routing protocol (CPRP)

In this section, we first describe a contact model attempting to mimic human (node) movement behavior, and a contact prediction metric built on it. Then we present our contact prediction-based protocol (CPRP). Our objective is to reduce the latency without any replication packets in the network.

3.1 Contact model

We built this model that considers prediction using history contact patterns, and tries to extract the potential rules in the node's contact context. It is motivated by the fact that most mobile nodes exhibit some regularity in their routine contacts.

1) Duty-cycle

A node's behavior presents periodical characteristic. The length of the behavior cycle is represented by a variable *C*, which is either varied between nodes or not, but stays constant for a specific node. *C* can either be set before the node's participation in the network or gained as the network runs on.

2) Contact

A contact happens when the distance between two nodes becomes smaller than the communication range. To a specific node, contact is described as a two-dimensional vector (ID,*t*). The ID shows to whom it contacts and *t* when the contact happens. We assume that the nodes have infinite bandwidth, so there is no need to consider the continuity of the contact.

3) Node's actual contact (NAC)

We define the NAC as a sequence of nodes' actual contacts gathered in a given cycle

$$\mathrm{NAC} = (\mathrm{ID}_1, t_1)(\mathrm{ID}_2, t_2)\ldots(\mathrm{ID}_{i-1}, t_{i-1})(\mathrm{ID}_i, t_i)(\mathrm{ID}_{i+1}, t_{i+1})\ldots (\mathrm{ID}_n, t_n)$$

We have

$$\begin{cases} t_0 > 0 \\ t_n < C \\ t_{i_1} > t_{i_2}, \text{when } i_1 > i_2 \end{cases}$$

NAC is increased by the up-to-date contact information detected by the node's Radio Frequency (RF) device while the network is running. When a cycle ends, the previous NAC is released for the newly coming NAC of the next cycle.

4) Node's contact pattern (NCP)

The contact sequence that a node may take in a given cycle is denoted as NCP. NCP is also a sequence of contacts which is saved in the node's contact pattern table (CPT). The available NCPs on a node describe its history contact characteristic and are expected to determine the future contacts. The

---

[1] Delay Tolerant Network Research Group (DTNRG). www.dtnrg.org

contact pattern table has a finite space of *k* NCPs. As they become abundant, the newly generated NCPs will displace the older ones.

As we know, NAC and NCP are both described by contact sequences, so we model the node's periodical contact as an edited NCP by allowing the following legal operations.

1) Insert a contact with ID *D* at position *i* of the NCP, and give NAC: $(\mathrm{ID}_1,t_1)(\mathrm{ID}_2,t_2)...(\mathrm{ID}_{i-1},t_{i-1})\left(D,\dfrac{t_{i-1}+t_i}{2}\right)$ $(\mathrm{ID}_i,t_i)(\mathrm{ID}_{i+1},t_{i+1})...(\mathrm{ID}_n,t_n)$. As shown above, inserting is defined on the first dimensional of a contact and the corresponding contact time is set at $\dfrac{t_{i-1}+t_i}{2}$ by default.

2) Change the contact time t_i at position *i* by another time *T* of the NCP, and give NAC: $(\mathrm{ID}_1,t_1)(\mathrm{ID}_2,t_2)...(\mathrm{ID}_{i-1},t_{i-1})$ $(\mathrm{ID}_i,T)(\mathrm{ID}_{i+1},t_{i+1})...(\mathrm{ID}_n,t_n)$. Changing is defined on the second dimensional of a contact. Stated in another way, time changing is making an increment

$$M=\left|T-t_i\right| \tag{1}$$

on t_i.

3) Delete the contact at position *i* of the NCP, and give $\mathrm{NAC}:(\mathrm{ID}_1,t_1)(\mathrm{ID}_2,t_2)...(\mathrm{ID}_{i-1},t_{i-1})(\mathrm{ID}_{i+1},t_{i+1})...(\mathrm{ID}_n,t_n)$. Different from the two operations above, erasure has effects on both dimensions.

In order to reflect the relationship between NCP and NAC for both demonstrates, we define two kinds of edit distance: ED_i and ED_t. ED_i is the edit degree on 'ID' demonstrate and ED_t on another. Varied from operations, we assign a nonnegative number to each operation as the spatial weight and the edit distance becomes the sum of weights of the operations taken on each demonstrate, which are chosen to gain the possible smallest total weight that can make NAC and NCP alike. Specifically, we define the weights as follows.

1) ED_i calculation

The cost of inserting a contact with ID *D* at position *i*

$$W_{\mathrm{I}i}=\begin{cases}\infty, & i \text{ is at the end of an UAC}\\ 1, & \text{otherwise}\end{cases} \tag{2}$$

The cost of deleting a contact at position *i*

$$W_{\mathrm{D}i}=\begin{cases}0, & \mathrm{ID}_1,...,\mathrm{ID}_{i-1} \text{ have already been deleted}\\ 1, & \text{otherwise}\end{cases} \tag{3}$$

2) ED_t calculation

The cost of changing contact time t_i to *T* by an increment of *M*

$$W_{\mathrm{CT}_i}=\begin{cases}M, & t_{i-1}<T<t_{i+1}\\ \infty, & \text{otherwise}\end{cases} \tag{4}$$

3.2 Metric for contact prediction

We assume that there are a number of NCPs describing periodical contact patterns in a node's CPT. The prediction problem turns to finding the NCP that best describes the current NAC. As NAC increases with real-time data during a cycle, it is possible that match result differs with time. When such an approximately matched NCP is found, the remaining sequence in NCP becomes the prediction result of future contacts. For the sake of clarity and without losing generality, the ED_i is the priority while taking different operation sets in comparison.

Suppose that

$$\begin{aligned}\mathrm{NAC}&=(a_1^{\mathrm{I}},a_1^{\mathrm{T}}),...,(a_m^{\mathrm{I}},...,a_m^{\mathrm{T}})\\ \mathrm{NCP}&=(b_1^{\mathrm{I}},b_1^{\mathrm{T}}),...,(b_n^{\mathrm{I}},...,b_n^{\mathrm{T}}).\end{aligned}$$

We can compute ED_i by constructing an $(m+1)\times(n+1)$ $[d(a_1^{\mathrm{I}},...,a_m^{\mathrm{I}},b_1^{\mathrm{I}},...,b_n^{\mathrm{I}})]$ matrix, which only involves the first dimension, for consideration that ED_i is affected only by ID changing with (i,j) ranging from (0, 0) to (m, n). Then the calculation can be reduced to the following dynamic programming taking place on the matrix

$$\begin{aligned}&d(1,1)=0\\ &d(a_1^{\mathrm{I}},...,a_i^{\mathrm{I}},1)=\begin{cases}\infty, & 1<i<m\\ 0, & i=1\end{cases}\\ &d(1,b_1^{\mathrm{I}},...,b_j^{\mathrm{I}})=0\end{aligned}$$

$$\begin{aligned}&d(a_1^{\mathrm{I}},...,a_i^{\mathrm{I}},b_1^{\mathrm{I}},...,b_j^{\mathrm{I}})\\ &=\begin{cases}d(a_1^{\mathrm{I}},...,a_{i-1}^{\mathrm{I}},b_1^{\mathrm{I}},...,b_{j-1}^{\mathrm{I}}), a_i^{\mathrm{I}}=b_j^{\mathrm{I}}\\ \min\{d(a_1^{\mathrm{I}},...,a_{i-1}^{\mathrm{I}},b_1^{\mathrm{I}},...,b_{j-1}^{\mathrm{I}})+W_{Db_j^{\mathrm{I}}}+\mathrm{W}_{\mathrm{I}a_i^{\mathrm{I}}},\\ d(a_1^{\mathrm{I}},...,a_i^{\mathrm{I}},b_1^{\mathrm{I}},...,b_{j-1}^{\mathrm{I}})+W_{Db_j^{\mathrm{I}}},\\ d(a_1^{\mathrm{I}},...,a_{i-1}^{\mathrm{I}},b_1^{\mathrm{I}},...,b_j^{\mathrm{I}})+W_{\mathrm{I}a_i^{\mathrm{I}}}\}, \quad \text{otherwise}\end{cases}\end{aligned} \tag{5}$$

To be efficient, we set an ED_i threshold ω in NCP-resembling process to avoid most of the computations. Whenever a $d(a_1^{\mathrm{I}},...,a_m^{\mathrm{I}},b_1^{\mathrm{I}},...,b_j^{\mathrm{I}})$ is found that has the property $d(a_1^{\mathrm{I}},...,a_m^{\mathrm{I}},b_1^{\mathrm{I}},...,b_j^{\mathrm{I}})<\omega$, an interceptive NCP ending at the *j*th contact is found.

When ED_i calculation is done, a set of NCPs are selected as candidates. ED_t calculation is simpler for only a searching process taking on both NAC and the fitter edited NCP versions. For each NCP piece, we have

$$\mathrm{ED}_t=\begin{cases}\infty, & \exists b_j^{\mathrm{T}}\leqslant a_{j+1}^{\mathrm{T}}, j=1,...,n-1\\ \displaystyle\sum_{i=1}^{m}W_{Ca_i^{\mathrm{T}}}, & \text{otherwise}\end{cases} \tag{6}$$

If one of the NCPs has the minimum ED_t and *j* is where it ends, the remaining sequence of the NCP: $(b_{j+1}^{\mathrm{I}},b_{j+1}^{\mathrm{T}}),...,$ $(b_n^{\mathrm{I}},b_n^{\mathrm{T}})$ is the future contact as a prediction result. If none of the NCPs is found to have a candidate or an ED_t under ∞ to a specific NAC, then that NAC will be identified as a new gathered NCP and will be added to CPT.

Figure 1 demonstrates an example of contact prediction, where a node experiences certain cycles with two NCPs in CPT.

UCP_1	(10,31)(8,36)(9,98)(7,124)(8,152)(7,181)(8,242)(7,294)
UCP_2	(6,21)(5,84)(6,147)(4,174)(6,194)(7,221)(6,254)(7,271)

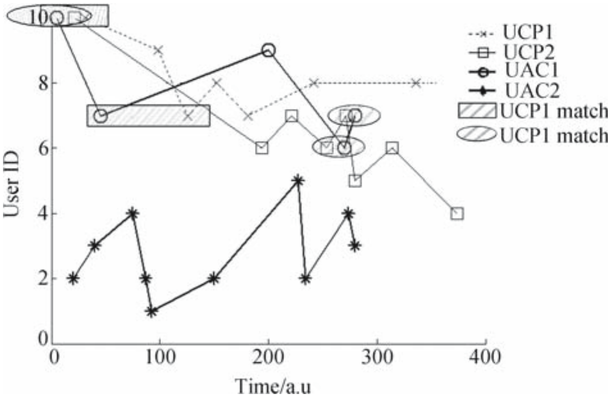


Fig. 1 Actual and predicted node contacts for $\mathrm{UAC}_1 = (10,5)(7,45)(9,200)(6,270)(7,280)$ $\mathrm{UAC}_2 = (2,20)(3,40)(4,75)(2,87)(1,92)(2,150)...$

We use an NCP matching algorithm with the ED_i threshold $\tau = 4$. NAC1 is one of the actual contact sequences which are tracked. The prediction results along the contacts' taking place are given in Table 1.

Table 1 Prediction results for NAC1

Current contact	NCP1		NCP2		Prediction output
	ED_i	ED_t	ED_i	ED_t	
2nd (7, 45)	1	105	1	192	(8, 152)(7, 181)(8, 242)(7, 294)(8, 36)
3rd (9, 200)	1	∞	2	290	(6, 254)(7, 271)(5, 280)(6, 314)(4, 374)
4th (6, 270)	2	∞	2	245.5	(7, 271)(5, 280)(6, 314)(4, 374)
5th (7, 280)	2	∞	2	254.5	(5, 280)(6, 314)(4, 374)

For NAC2, the ED_i with both NCPs exceeds 4 after its 4th contact, so no match is found in the NCP table and NAC2 will be taken into the NCP table as a newly gathered NCP after the end of that cycle.

3.3 Contact prediction-based routing protocol

As stated earlier, the basic idea of our scheme is to create routes on the nodes on which a quick contact with destination will take place. In this way, we can significantly reduce the delivery latency, which is proved by simulation results in Sect. 4.

Initially, messages are generated by source nodes and put into their forwarding buffers immediately. In order to avoid circuits and deal with buffer overflow, every message has a unique ID and a time stamp indicating its birth time. Considering unpredictable long duration disconnections caused by high-degree geographical isolation of the source or prediction errors, we use birth time stamp to pick up these long delay messages and drop them to save buffer for newly generated messages.

When the RF devices on a node *A* and another node *B* detect each other in the communication range, they both refresh their NACs with this new contact information and rerun the prediction process to get the latest prediction result. For the sake of clarity and without losing generality, we assume that node *A* is the only message carrier in this pair-wise with a message destined for node *C*. The message forwarding process contains the following steps.

1) *A* challenges *B* with the message ID and message destination

$$A \rightarrow B : (\mathrm{ID}, C)$$

2) After receiving *A*'s message list, *B* responds with its contact information to A with a message ID and the next contact time to *C*(NCT(*B*,*C*))

$$B \rightarrow A : (\mathrm{ID}, \mathrm{NCT}(B, C))$$

in which

$$\mathrm{NCT} = \begin{cases} 0, & B = C \\ \infty, & B \text{ has already been passed by message(ID)} \\ \text{predicted next contct time to } C, \text{otherwise} \end{cases} \quad (7)$$

3) *A* compares NCT(*B*,*C*) with its own next contact time to node *C*: NCT(*A*,*C*). If $\mathrm{NCT}(A,C) < \mathrm{NCT}(B,C)$, *A* forwards the message to *C*. Otherwise, it does nothing.

4 Simulation results

4.1 Methodology

We have implemented an isolated simulator to evaluate the CPRP presented in this paper. This simulator only implements the transport and network layers. Several assumptions are made with regard to the following layers.

1) Infinite bandwidth

Since delay is mostly caused by network partitions, we only take into consideration the interval between contacts. Thus, an infinite bandwidth of each node's transmitter is assumed to neglect the transmission delay and propagation delay.

2) Neighborhood detection

We assume that the MAC layer could detect another node as soon as it comes into the communication range.

3) Limited ability as a message carrier

Since there is no permanent route in sparse MANETs, identity as a source doesn't make sense after message generation and flow if the exits are taken as separate messages. Both messages originated and received from others share the same finite forwarding buffer on a node.

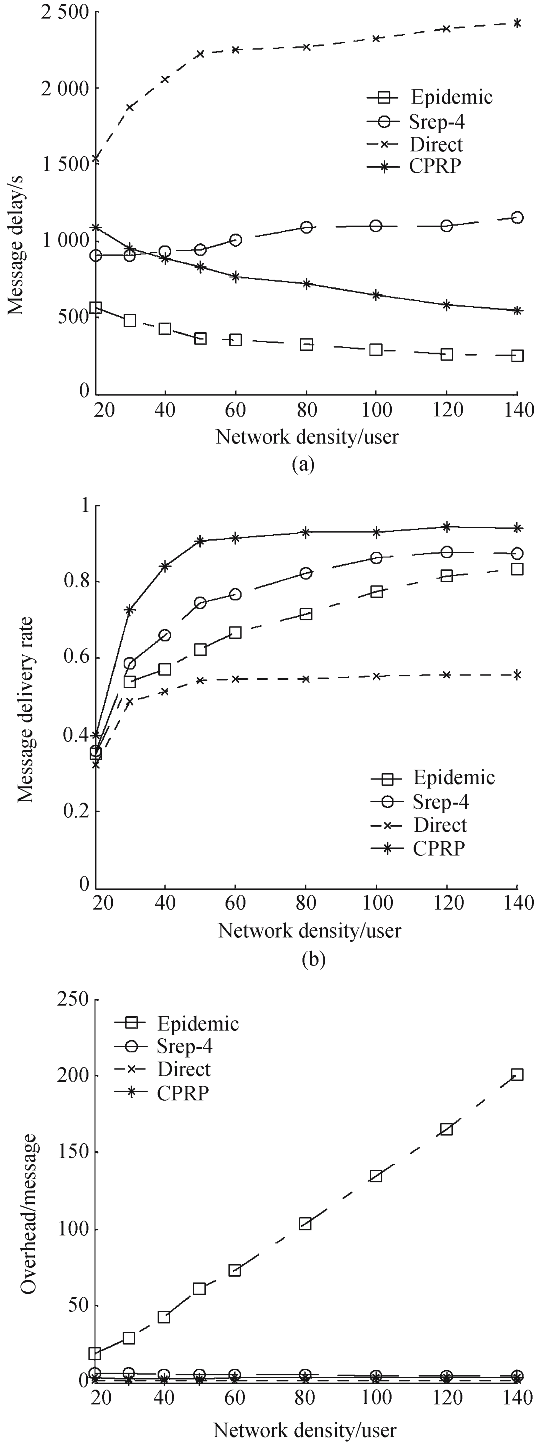
We also implemented the following routing algorithms for comparison: epidemic routing (epidemic) [5], direct contact routing (direct) [7], and simple replication routing with replication factor *k* (srep_*k*) [7]

We are interested in several metrics of the algorithm: message delivery rate, latency, and routing overhead. These metrics are measured over the network density, the buffer size, and the CPT size. The message delivery rate is defined as the ratio of the number of successfully delivered messages to the total number of messages generated. The message delay is defined on the successful delivered messages as the average interval from their generation to their arrival. The routing overhead is the ratio of the number of messages transmitted to the number of messages successfully delivered.

Each simulation runs for 40 000 s within a 5 000 m × 5 000 m area. Nodes are randomly chosen as sources, which send messages to randomly chosen destinations every second. As stated above, we believe movements of human and vehicle (e.g. buses and planes) are structured. In order to get a high reality, we use a community waypoint model (CWM) to define the simulation environment as much like biological movements as possible. Nodes are randomly distributed to ten communities that have a radius of 500 m. They move randomly within the community which they belong to. Similar to the traditional waypoint model, each community has a set of five waypoints. The community moves along the path restricted by the waypoints. Every time a community arrives at a waypoint, it refreshes its speed randomly ranging from 5 to 30 m/s (similar to human walking or automobiles) and moves towards the next waypoint. All communities are synchronized by conjunctive cycles of 500 s. If a community reaches its terminus, it stops moving till another life cycle begins. The default settings for other parameters are listed in Table 2.

Table 2 Default parameter settings

Parameter	Value
Transmission range	100/m
Resembling threshold ω	5
CPR size	15/NCPs
Forward buffer size	160/messages



(a)

(b)

(c)

Fig. 2 Performance under different network density
(a) message delay; (b) message delivery rate; (c) total message duplication

4.2 Impact of network density

We first study how network density affects each scheme. Figure 2(a) shows the message latency for different network densities with nodes from 20 to 140. Due to the community waypoint model (CWM), community size ranges form 2 to 14 nodes. The epidemic routing has the least latency which shows the upper bound of the network capability. The CPRP is about twice that on every density but still lower than srep-4. Direct contact scheme has the highest delivery latency because there are fewer chances to meet the destination than any other nodes. For epidemic routing and the CPRP, as the networks' density increases, more nodes are involved in message transmission, thus leading to decreased delay. For the CPRP, the number of contacts during a cycle increases when the density increases, and it takes more relays with shorter waiting time. This decreases the latency in the CPRP. For srep-4 and direct contact routing, the growing number of nodes with a constantly offered load decreases the average buffer usage. Messages with longer delivery delay are held in the network instead of being dropped, thus increasing the average latency.

Figure 2(b) shows that the CPRP outperforms all the other schemes in the message delivery rate. When there are 20 nodes, all the schemes have a delivery rate of about 30%. Most messages carried by a node acting as a relay are hustled by newly generated messages in a node as a source due to high offered load per node. As more nodes are involved in networking, transmission load becomes the primary cause of message loss instead of the offered load. When the network density reaches 50 nodes, the CPRP achieves a delivery rate of more than 85% while epidemic routing only delivers 50% of the messages. This is because in the CPRP scheme the actual contact pattern and prediction of future contacts decrease the delivery latency when only the original messages are passed through nodes, which leads to low transmission load and high delivery rate. The increase of the delivery rate in the srep-4 scheme and direct contact scheme is because the connectivity among nodes rises with the network density.

Figure 2(c) presents the routing overhead corresponding to each forwarding algorithm. Routing overhead is measured by using the ratio of messages transmitted to messages received. Since the direct contact scheme transmits a fixed amount of messages with respect to the data received, its overhead is constant. The srep-4 scheme forwards a fixed number of copies for each message generated, but due to the loss during delivery, the overhead is above 4, especially in a low network density. On the other hand, in both the CPRP and epidemic routing, relays also forward to other relays. The difference is that multiple copies of the original message are distributed after the first delivery of the original messages in epidemic routing while only the original messages are passed by nodes in the CPRP scheme. As a result, the overhead in epidemic has a linear growth with the network density and does not grow much in the CPRP.

4.3 Impact of buffer size

In this section we evaluate the impact of the node's buffer size on message delivery performance. Figure 3(a) shows the message delay, which tends to increase to the buffer size. For Epidemic routing and srep-4, the increase of delay is because the chance of going through multi-hop paths without being dropped grows. For direct contact routing, as the buffer size increases, a node can buffer messages longer before it meets the destination. The CPRP's average delay is not affected much by the buffer size, and it is near to the higher bound of epidemic routing. This is because CPRP makes partially optimized decisions on delivery latency while forwarding a message.

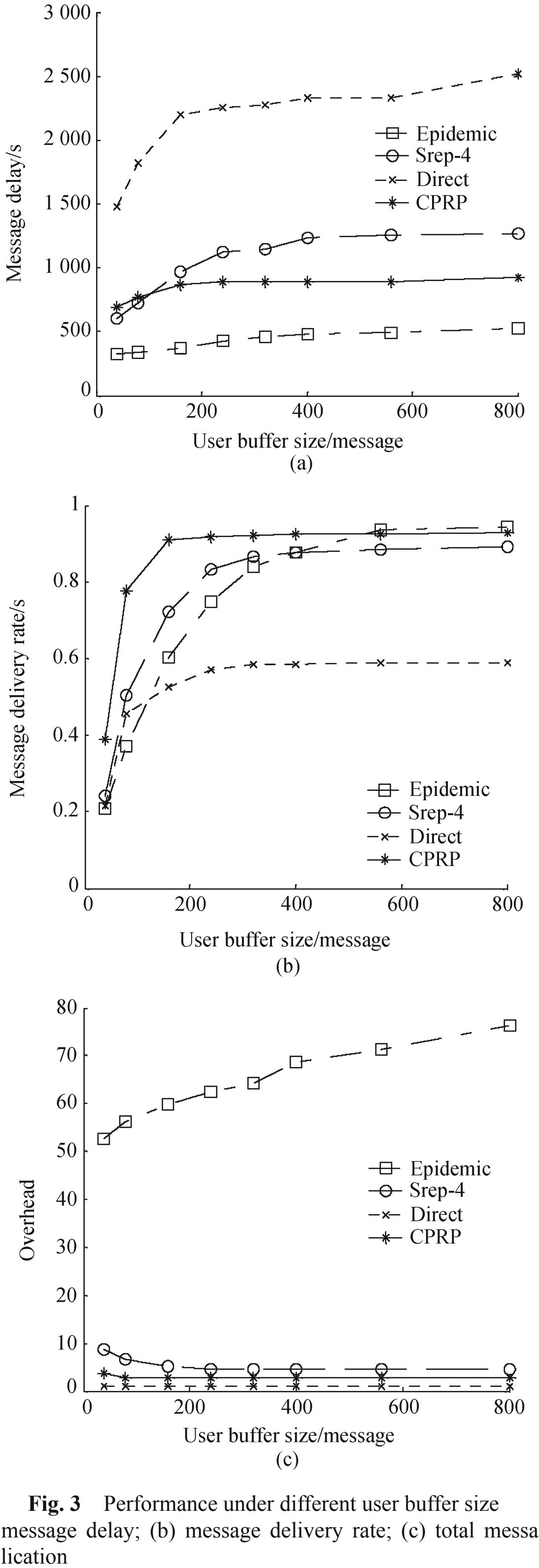


Fig. 3 Performance under different user buffer size
(a) message delay; (b) message delivery rate; (c) total message duplication

Figure 3(b) presents the message delivery rates under different buffer size. The delivery rate of direct contact routing stays low because the chance of direct contact between source and destination is low, and if a message is not delivered in the first cycle after generation, there is a high probability that the carrier will never meet the right destination. The CPRP significantly outperforms the other two epidemic style schemes with regard to all buffer size. It works well to achieve a delivery rate of more than 80% at a small buffer size of 60 messages while the epidemic scheme only delivers 35% of the messages. This reason is that in the CPRP, only one copy is kept in the whole network, and it is easier to be purged out by a successful delivery. Epidemic routing distributes a large number of copies to the network. Undelivered messages are hustled by new clones as soon as an origin is generated somewhere, which causes problems for the buffer contentions. As the node's buffer size increases, their message delivery rate increases, but it is still lower than the CPRP.

Figure 3(c) shows the total message duplications, which also indicates the energy efficiency. The direct contact scheme achieves the best energy efficiency, the CPRP as the second because messages are just forwarded via one hop to reach their destinations in direct routing but may be passed through a number of hops in the CPRP. They both greatly outperform the epidemic scheme.

4.4 Impact of the size of contact pattern table (CPT)

In this section, we evaluate the performance of the CPRP with different CPT sizes. Figure 4(a) and (b) shows the message delay distribution for simulations using CPT of 15 NCPs and of 20 NCPs respectively. Table 3 lists the average delay, delivery rate, and overhead. Generally, a small ED_i threshold τ will both enhance the prediction accuracy and the number of NCPs picked during the matching process. We can see that in Fig. 4(a) and (b) a τ of 10 has distributed delay in a lower area than that of 12, which also achieves better delivery rates and overheads. However, a lower τ of 8 does not show that advantage with a CPT of 15 NCPs. This is because an over constrained threshold will cause frequent drops of existing NCPs with inadequate CPT space. When the database grows to 20 NCPs, a τ of 8 fits and outperforms other bigger τ. We found that an appropriate τ is taken depending on the network density and the CPR size. A small threshold makes more accurate predictions especially in busy-contacted networks, and also a higher requirement for the CPT size.

Table 3 Performance under different NCP table sizes

NCP table size/NCP	15			20		
	$\tau = 8$	$\tau = 10$	$\tau = 12$	$\tau = 8$	$\tau = 10$	$\tau = 12$
Average delay/s	973.21	842.1	889.66	785.21	802.3	840.21
Delivery rate	0.88	0.92	0.91	0.92	0.92	0.91
Overhead	2.94	2.54	2.51	2.11	2.14	2.5

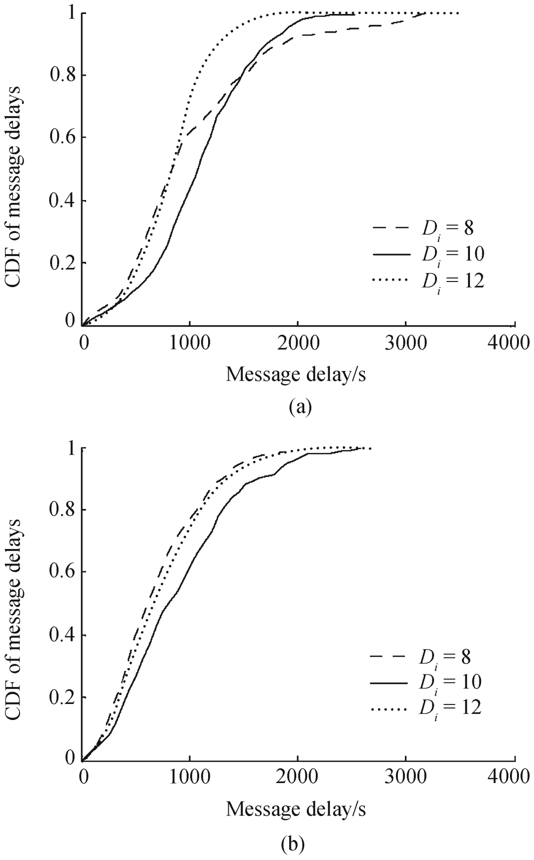


Fig. 4 Performance of CPRP under different NCP buffer size
(a) 15 UCPs; (b) 20 UCPs

5 Conclusions and future work

We have explored the fundamental communication approach of movements and links in MANETs to structured contacts known ahead to nodes. We proposed a novel contact model that describes a connection between a nodes history contact and its contact pattern as an edited version, based on which a prediction algorithm with minimum edit distance is advanced. With this refined algorithm of contact prediction, we developed a prediction-based routing protocol (CPRP) for sparse MANETs. Our scheme reduces delivery latency by passing the original message through a path and decreasing waiting time to get to the destination. Meanwhile, neither the routing information nor the message duplication is flooded in the network, which further leads to low overhead.

Future work includes long-term contact predictions and parameter analysis. Practical lower layer protocols will be investigated to support a testbed built on a campus student network, and more field experiments will be carried out.

References

1. Fall K. A delay-tolerant network architecture for challenged Internets. In: Proceeding of the 2003 Conference on Application, Technologies, Architectures, and Protocols for Computer Communications SIGCOMM'03, 2003, 27–34
2. Wang Y, Jain S, Martonosi M, et al. Erasure coding based routing in opportunistic networks. In: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking WDTN'05, 2005, 229–236

3. Leguay J, Friedman T, Conan V. Dtn routing in a mobility pattern space. In: Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking, 2005, 8: 276−283
4. Lindgren A D, Schelen O. Probabilistic routing in intermittently connected networks. ACM SIGMOBILE Mobile Computing and Communications Review, 2003, 7(3): 19–20
5. Vahdat Becker D. Epidemic routing for partially-connected ad-hoc networks. Duke University Tech Rep, CS-2000-06, 2000, 7: 27–31
6. Juang P, Oki H, Wang Y, et al. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebraNet. In: Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems ASPLOS-X. 2002, Vol. 37, 30, 36, Issue 10, 5, 5
7. Spyropoulos T, Psounis K, Raghavendra C S. Single-copy routing in intermittently connected mobile networks. In: IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks. 2004, 235–244
8. Hussein, Saadawi T. Ant routing algorithm for mobile ad-hoc networks (ARAMA). International Performance Computing and Communications Conference, 2004, 4: 15−17
9. Burleigh S, Hooke A, Torgerson L, et al. Delay-tolerant networking: An approach to interplanetary internet. IEEE Communications Magazine, 2003, 41: 128−136
10. Grossglauser M, Tse D. Mobility increases the capacity of ad-hoc wireless networks. In: Proceedings of INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, 2001, 3: 1 360–1 369