Evaluating Opportunistic Routing Protocols with Large Realistic Contact Traces

Libo Song and David F. Kotz
Institute for Security Technology Studies (ISTS)
Department of Computer Science, Dartmouth College, Hanover, NH, USA 03755

ABSTRACT

Traditional mobile ad hoc network (MANET) routing protocols assume that contemporaneous end-to-end communication paths exist between data senders and receivers. In some mobile ad hoc networks with a sparse node population, an end-to-end communication path may break frequently or may not exist at any time. Many routing protocols have been proposed in the literature to address the problem, but few were evaluated in a realistic "opportunistic" network setting. We use simulation and contact traces (derived from logs in a production network) to evaluate and compare five existing protocols: direct-delivery, epidemic, random, PRoPHET, and Link-State, as well as our own proposed routing protocol. We show that the direct delivery and epidemic routing protocols suffer either low delivery ratio or high resource usage, and other protocols make tradeoffs between delivery ratio and resource usage.

Categories and Subject Descriptors

C.2.4 [**Computer Systems Organization**]: Computer Communication Networks—*Distributed Systems*

General Terms

Performance, Design

Keywords

Opportunistic Networks, Routing, Simulation

1. INTRODUCTION

Mobile opportunistic networks are one kind of delay-tolerant network (DTN) [6]. Delay-tolerant networks provide service despite long link delays or frequent link breaks. Long link delays happen in networks with communication between nodes at a great distance, such as interplanetary networks [2]. Link breaks are caused by nodes moving out of range, environmental changes, interference from other moving objects, radio power-offs, or failed nodes. For us, mobile opportunistic networks are those DTNs with sparse node population and frequent link breaks caused by power-offs and the mobility of the nodes.

Mobile opportunistic networks have received increasing interest from researchers. In the literature, these networks include mobile sensor networks [25], wild-animal tracking networks [11], "pocket-switched" networks [8], and transportation networks [1, 14]. We expect to see more opportunistic networks when the one-laptop-per-child (OLPC) project [18] starts rolling out inexpensive laptops with wireless networking capability for children in developing countries, where often no infrastructure exits. Opportunistic networking is one promising approach for those children to exchange information.

One fundamental problem in opportunistic networks is how to route messages from their source to their destination. Mobile opportunistic networks differ from the Internet in that disconnections are the norm instead of the exception. In mobile opportunistic networks, communication devices can be carried by people [4], vehicles [1] or animals [11]. Some devices can form a small mobile ad hoc network when the nodes move close to each other. But a node may frequently be isolated from other nodes. Note that traditional Internet routing protocols and ad hoc routing protocols, such as AODV [20] or DSDV [19], assume that a contemporaneous end-to-end path exists, and thus fail in mobile opportunistic networks. Indeed, there may never exist an end-to-end path between two given devices.

In this paper, we study protocols for routing messages between wireless networking devices carried by people. We assume that people send messages to other people occasionally, using their devices; when no direct link exists between the source and the destination of the message, other nodes may relay the message to the destination. Each device represents a unique person (it is out of the scope of this paper when a device maybe carried by multiple people). Each message is destined for a specific person and thus for a specific node carried by that person. Although one person may carry multiple devices, we assume that the sender knows which device is the best to receive the message. We do not consider multicast or geocast in this paper.

Many routing protocols have been proposed in the literature. Few of them were evaluated in realistic network settings, or even in realistic simulations, due to the lack of any realistic people mobility model. Random walk or random way-point mobility models are often used to evaluate the performance of those routing protocols. Although these synthetic mobility models have received extensive interest by mobile ad hoc network researchers [3], they do not reflect people's mobility patterns [9]. Realising the limitations of using random mobility models in simulations, a few researchers have studied routing protocols in mobile opportunistic networks with realistic mobility traces. Chaintreau et al. [5] theoretically analyzed the impact of routing algorithms over a model derived from a realistic mobility data set. Su et al. [22] simulated a set of routing

CHANTS'07, September 14, 2007, Montréal, Québec, Canada.

protocols in a small experimental network. Those studies help researchers better understand the theoretical limits of opportunistic networks, and the routing protocol performance in a small network (20–30 nodes).

Deploying and experimenting large-scale mobile opportunistic networks is difficult, we too resort to simulation. Instead of using a complex mobility model to mimic people's mobility patterns, we used mobility traces collected in a production wireless network at Dartmouth College to drive our simulation. Our message-generation model, however, was synthetic.

To the best of our knowledge, we are the first to simulate the effect of routing protocols in a *large-scale* mobile opportunistic network, using realistic contact traces derived from real traces of a production network with more than $5,000$ users.

Using realistic contact traces, we evaluate the performance of three "naive" routing protocols (direct-delivery, epidemic, and random) and two prediction-based routing protocols, PRoPHET [16] and Link-State [22]. We also propose a new prediction-based routing protocol, and compare it to the above in our evaluation.

2. ROUTING PROTOCOL

A routing protocol is designed for forwarding messages from one node (source) to another node (destination). Any node may generate messages for any other node, and may carry messages destined for other nodes. In this paper, we consider only messages that are unicast (single destination).

DTN routing protocols could be described in part by their *transfer probability* and *replication probability*; that is, when one node meets another node, what is the probability that a message should be transfered and if so, whether the sender should retain its copy. Two extremes are the direct-delivery protocol and the epidemic protocol. The former transfers with probability 1 when the node meets the destination, 0 for others, and no replication. The latter uses transfer probability 1 for all nodes and unlimited replication. Both these protocols have their advantages and disadvantages. All other protocols are between the two extremes.

First, we define the notion of contact between two nodes. Then we describe five existing protocols before presenting our own proposal.

A *contact* is defined as a period of time during which two nodes have the opportunity to communicate. Although we are aware that wireless technologies differ, we assume that a node can reliably detect the beginning and end time of a contact with nearby nodes. A node may be in contact with several other nodes at the same time.

The contact history of a node is a sequence of contacts with other nodes. Node i has a contact history $H_i(j)$, for each other node j, which denotes the historical contacts between node i and node j. We record the start and end time for each contact; however, the last contacts in the node's contact history may not have ended.

2.1 Direct Delivery Protocol

In this simple protocol, a message is transmitted only when the source node can directly communicate with the destination node of the message. In mobile opportunistic networks, however, the probability for the sender to meet the destination may be low, or even zero.

2.2 Epidemic Routing Protocol

The epidemic routing protocol [23] floods messages into the network. The source node sends a copy of the message to every node that it meets. The nodes that receive a copy of the message also send a copy of the message to every node that they meet. Eventually, a copy of the message arrives at the destination of the message.

This protocol is simple, but may use significant resources; excessive communication may drain each node's battery quickly. Moreover, since each node keeps a copy of each message, storage is not used efficiently, and the capacity of the network is limited.

At a minimum, each node must expire messages after some amount of time or stop forwarding them after a certain number of hops. After a message expires, the message will not be transmitted and will be deleted from the storage of any node that holds the message.

An optimization to reduce the communication cost is to transfer *index messages* before transferring any data message. The index messages contain IDs of messages that a node currently holds. Thus, by examining the index messages, a node only transfers messages that are not yet contained on the other nodes.

2.3 Random Routing

An obvious approach between the above two extremes is to select a transfer probability between 0 and 1 to forward messages at each contact. We use a simple replication strategy that allows only the source node to make replicas, and limits the replication to a specific number of copies. The message has some chance of being transferred to a highly mobile node, and thus may have a better chance to reach its destination before the message expires.

2.4 PRoPHET Protocol

PRoPHET [16] is a *Probabilistic Routing Protocol using History of past Encounters and Transitivity* to estimate each node's delivery probability for each other node. When node i meets node j, the delivery probability of node i for j is updated by

$$p'_{ij} = (1 - p_{ij})p_0 + p_{ij}, \tag{1}$$

where p_0 is an initial probability, a design parameter for a given network. Lindgren et al. [16] chose 0.75, as did we in our evaluation. When node i does not meet j for some time, the delivery probability decreases by

$$p'_{ij} = \alpha^k p_{ij}, \tag{2}$$

where α is the aging factor ($\alpha < 1$), and k is the number of time units since the last update.

The PRoPHET protocol exchanges index messages as well as delivery probabilities. When node i receives node j's delivery probabilities, node i may compute the transitive delivery probability through j to z with

$$p'_{iz} = p_{iz} + (1 - p_{iz})p_{ij}p_{jz}\beta, \tag{3}$$

where β is a design parameter for the impact of transitivity; we used $\beta = 0.25$ as did Lindgren [16].

2.5 Link-State Protocol

Su et al. [22] use a link-state approach to estimate the weight of each path from the source of a message to the destination. They use the median inter-contact duration or exponentially aged inter-contact duration as the weight on links. The exponentially aged inter-contact duration of node i and j is computed by

$$w'_{ij} = \alpha w_{ij} + (1 - \alpha)I, \tag{4}$$

where I is the new inter-contact duration and α is the aging factor.

Nodes share their link-state weights when they can communicate with each other, and messages are forwarded to the node that have the path with the lowest link-state weight.

3. TIMELY-CONTACT PROBABILITY

We also use historical contact information to estimate the probability of meeting other nodes in the future. But our method differs in that we estimate the contact probability within a period of time. For example, what is the contact probability in the next hour? Neither PRoPHET nor Link-State considers time in this way.

One way to estimate the "timely-contact probability" is to use the ratio of the total contact duration to the total time. However, this approach does not capture the frequency of contacts. For example, one node may have a long contact with another node, followed by a long non-contact period. A third node may have a short contact with the first node, followed by a short non-contact period. Using the above estimation approach, both examples would have similar contact probability. In the second example, however, the two nodes have more frequent contacts.

We design a method to capture the contact frequency of mobile nodes. For this purpose, we assume that even short contacts are sufficient to exchange messages.[1]

The probability for node i to meet node j is computed by the following procedure. We divide the contact history $H_i(j)$ into a sequence of n periods of ΔT starting from the start time (t_0) of the first contact in history $H_i(j)$ to the current time. We number each of the n periods from 0 to $n-1$, then check each period. If node i had any contact with node j during a given period m, which is $[t_0 + m\Delta T, t_0 + (m+1)\Delta T)$, we set the contact status I_m to be 1; otherwise, the contact status I_m is 0. The probability $p_{ij}^{(0)}$ that node i meets node j in the next ΔT can be estimated as the average of the contact status in prior intervals:

$$p_{ij}^{(0)} = \frac{1}{n} \sum_{m=0}^{n-1} I_m. \tag{5}$$

To adapt to the change of contact patterns, and reduce the storage space for contact histories, a node may discard old history contacts; in this situation, the estimate would be based on only the retained history.

The above probability is the direct contact probability of two nodes. We are also interested in the probability that we may be able to pass a message through a sequence of k nodes. We define the k-order probability inductively,

$$p_{ij}^{(k)} = p_{ij}^{(0)} + \sum_{\alpha} p_{i\alpha}^{(0)} p_{\alpha j}^{(k-1)}, \tag{6}$$

where α is any node other than i or j.

3.1 Our Routing Protocol

We first consider the case of a two-hop path, that is, with only one relay node. We consider two approaches: either the receiving neighbor decides whether to act as a relay, or the source decides which neighbors to use as relay.

3.1.1 Receiver Decision

Whenever a node meets other nodes, they exchange all their messages (or as above, index messages). If the destination of a message is the receiver itself, the message is delivered. Otherwise, if the probability of delivering the message to its destination through this receiver node within ΔT is greater than or equal to a certain threshold, the message is stored in the receiver's storage to forward to the destination. If the probability is less than the threshold, the receiver discards the message. Notice that our protocol replicates the message whenever a good-looking relay comes along.

3.1.2 Sender Decision

To make decisions, a sender must have the information about its neighbors' contact probability with a message's destination. Therefore, meta-data exchange is necessary.

When two nodes meet, they exchange a meta-message, containing an unordered list of node IDs for which the sender of the meta-message has a contact probability greater than the threshold.

After receiving a meta-message, a node checks whether it has any message that destined to its neighbor, or to a node in the node list of the neighbor's meta-message. If it has, it sends a copy of the message.

When a node receives a message, if the destination of the message is the receiver itself, the message is delivered. Otherwise, the message is stored in the receiver's storage for forwarding to the destination.

3.1.3 Multi-node Relay

When we use more than two hops to relay a message, each node needs to know the contact probabilities along all possible paths to the message destination.

Every node keeps a contact probability matrix, in which each cell p_{ij} is a contact probability between to nodes i and j. Each node i computes its own contact probabilities (row i) with other nodes using Equation (5) whenever the node ends a contact with other nodes. Each row of the contact probability matrix has a version number; the version number for row i is only increased when node i updates the matrix entries in row i. Other matrix entries are updated through exchange with other nodes when they meet.

When two nodes i and j meet, they first exchange their contact probability matrices. Node i compares its own contact matrix with node j's matrix. If node j's matrix has a row l with a higher version number, then node i replaces its own row l with node j's row l. Likewise node j updates its matrix. After the exchange, the two nodes will have identical contact probability matrices.

Next, if a node has a message to forward, the node estimates its neighboring node's order-k contact probability to contact the destination of the message using Equation (6). If $p_{ij}^{(k)}$ is above a threshold, or if j is the destination of the message, node i will send a copy of the message to node j.

All the above effort serves to determine the transfer probability when two nodes meet. The replication decision is orthogonal to the transfer decision. In our implementation, we always replicate. Although PRoPHET [16] and Link-State [22] do no replication, as described, we added replication to those protocols for better comparison to our protocol.

4. EVALUATION RESULTS

We evaluate and compare the results of direct delivery, epidemic, random, PRoPHET, Link-State, and timely-contact routing protocols.

4.1 Mobility traces

We use real mobility data collected at Dartmouth College. Dartmouth College has collected association and disassociation messages from devices on its wireless network wireless users since spring 2001 [13]. Each message records the wireless card MAC address, the time of association/disassociation, and the name of the access point. We treat each unique MAC address as a node. For

---

[1] In our simulation, however, we accurately model the communication costs and some short contacts will not succeed in transfer of all messages.

more information about Dartmouth's network and the data collection, see previous studies [7, 12].

Our data are not contacts in a mobile ad hoc network. We can approximate contact traces by assuming that two users can communicate with each other whenever they are associated with the same access point. Chaintreau et al. [5] used Dartmouth data traces and made the same assumption to theoretically analyze the impact of human mobility on opportunistic forwarding algorithms. This assumption may not be accurate,[2] but it is a good first approximation. In our simulation, we imagine the same clients and same mobility in a network with no access points. Since our campus has full WiFi coverage, we assume that the location of access points had little impact on users' mobility.

We simulated one full month of trace data (November 2003) taken from CRAWDAD [13], with $5,142$ users. Although prediction-based protocols require prior contact history to estimate each node's delivery probability, our preliminary results show that the performance improvement of warming-up over one month of trace was marginal. Therefore, for simplicity, we show the results of all protocols without warming-up.

4.2 Simulator

We developed a custom simulator.[3] Since we used contact traces derived from real mobility data, we did not need a mobility model and omitted physical and link-layer details for node discovery. We were aware that the time for neighbor discovery in different wireless technologies vary from less than one seconds to several seconds. Furthermore, connection establishment also takes time, such as DHCP. In our simulation, we assumed the nodes could discover and connect each other instantly when they were associated with a same AP. To accurately model communication costs, however, we simulated some MAC-layer behaviors, such as collision.

The default settings of the network of our simulator are listed in Table 1, using the values recommended by other papers [22, 16]. The message probability was the probability of generating messages, as described in Section 4.3. The default transmission bandwidth was 11 Mb/s. When one node tried to transmit a message, it first checked whether any nearby node was transmitting. If it was, the node backed off a random number of slots. Each slot was 1 millisecond, and the maximum number of backoff slots was 30. The size of messages was uniformly distributed between 80 bytes and 1024 bytes. The hop count limit (HCL) was the maximum number of hops before a message should stop forwarding. The time to live (TTL) was the maximum duration that a message may exist before expiring. The storage capacity was the maximum space that a node can use for storing messages. For our routing method, we used a default prediction window ΔT of 10 hours and a probability threshold of 0.01. The replication factor r was not limited by default, so the source of a message transferred the messages to any other node that had a contact probability with the message destination higher than the probability threshold.

4.3 Message generation

After each contact event in the contact trace, we generated a message with a given probability; we choose a source node and a destination node randomly using a uniform distribution across nodes seen in the contact trace up to the current time. When there were more contacts during a certain period, there was a higher likelihood that a new message was generated in that period. This correlation is not unreasonable, since there were more movements during the day than during the night, and so the number of contacts. Figure 1 shows the statistics of the numbers of movements and the numbers of contacts during each hour of the day, summed across all users and all days. The plot shows a clear diurnal activity pattern. The activities reached lowest around 5am and peaked between 4pm and 5pm. We assume that in some applications, network traffic exhibits similar patterns, that is, people send more messages during the day, too.

Table 1: Default Settings of the Simulation

Parameter	Default value
message probability	0.001
bandwidth	11 Mb/s
transmission slot	1 millisecond
max backoff slots	30
message size	80–1024 bytes
hop count limit (HCL)	unlimited
time to live (TTL)	unlimited
storage capacity	unlimited
prediction window ΔT	10 hours
probability threshold	0.01
contact history length	20
replication	always
aging factor α	0.9 (0.98 PRoPHET)
initial probability p_0	0.75 (PRoPHET)
transitivity impact β	0.25 (PRoPHET)

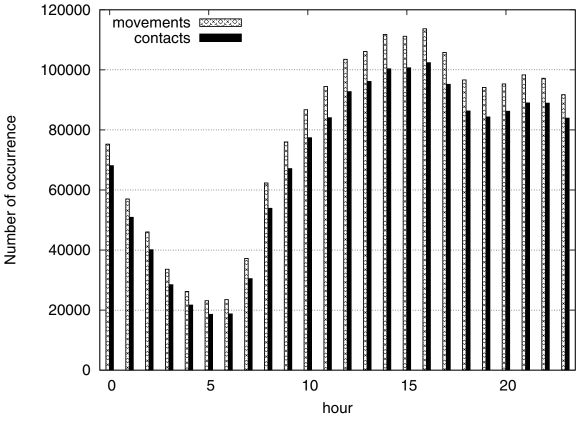

Figure 1: Movements and contacts duration each hour

Messages expire after a TTL. We did not use proactive methods to notify nodes the delivery of messages, so that the messages can be removed from storage.

4.4 Metrics

We define a set of metrics that we use in evaluating routing protocols in opportunistic networks:

- *delivery ratio*, the ratio of the number of messages delivered to the number of total messages generated.
- *message transmissions*, the total number of messages transmitted during the simulation across all nodes.

---

[2] Two nodes may not have been able to directly communicate while they were at two far sides of an access point, or two nodes may have been able to directly communicate if they were between two adjacent access points.

[3] We tried to use a general network simulator (ns2), which was extremely slow when simulating a large number of mobile nodes (in our case, more than 5000 nodes), and provided unnecessary detail in modeling lower-level network protocols.

- *meta-data transmissions*, the total number of meta-data units transmitted during the simulation across all nodes.
- *message duplications*, the number of times a message copy occurred, due to replication.
- *delay*, the duration between a message's generation time and the message's delivery time.
- *storage usage*, the max and mean of maximum storage (bytes) used across all nodes.

4.5 Results

Here we compare simulation results of the six routing protocols.

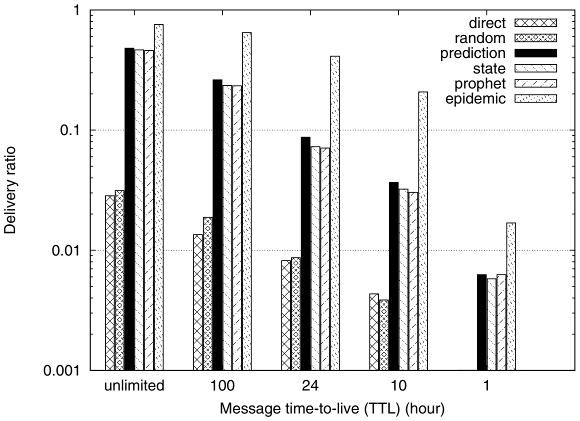

Figure 2: Delivery ratio (log scale). The direct and random protocols for one-hour TTL had delivery ratios that were too low to be visible in the plot.

Figure 2 shows the delivery ratio of all the protocols, with different TTLs. (In all the plots in the paper, "prediction" stands for our method, "state" stands for the Link-State protocol, and "prophet" represents PRoPHET.) Although we had 5,142 users in the network, the direct-delivery and random protocols had low delivery ratios (note the log scale). Even for messages with an unlimited lifetime, only 59 out of 2077 messages were delivered during this one-month simulation. The delivery ratio of epidemic routing was the best. The three prediction-based approaches had low delivery ratio, compared to epidemic routing. Although our method was slightly better than the other two, the advantage was marginal.

The high delivery ratio of epidemic routing came with a price: excessive transmissions. Figure 3 shows the number of message data transmissions. The number of message transmissions of epidemic routing was more than 10 times higher than for the prediction-based routing protocols. Obviously, the direct delivery protocol had the lowest number of message transmissions – the number of message delivered. Among the three prediction-based methods, the PRoPHET transmitted fewer messages, but had comparable delivery-ratio as seen in Figure 2.

Figure 4 shows that epidemic and all prediction-based methods had substantial meta-data transmissions, though epidemic routing had relatively more, with shorter TTLs. Because epidemic protocol transmitted messages at every contact, in turn, more nodes had messages that required meta-data transmission during contact. The direct-delivery and random protocols had no meta-data transmissions.

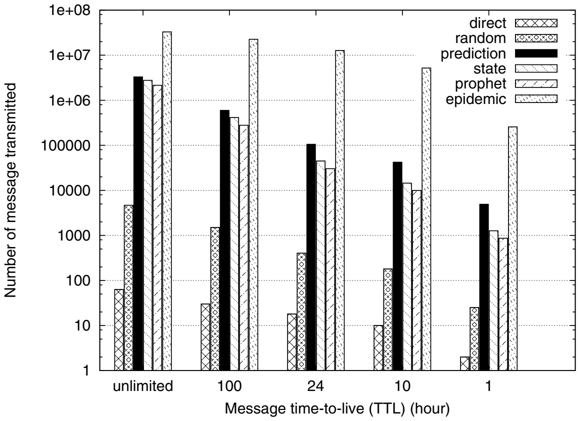

Figure 3: Message transmissions (log scale)

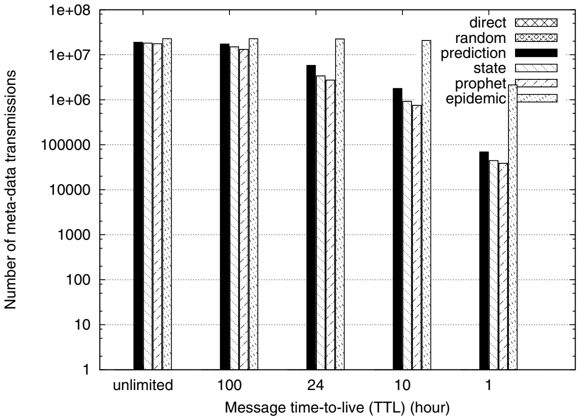

Figure 4: Meta-data transmissions (log scale). Direct and random protocols had no meta-data transmissions.

In addition to its message transmissions and meta-data transmissions, the epidemic routing protocol also had excessive message duplications, spreading replicas of messages over the network. Figure 5 shows that epidemic routing had one or two orders more duplication than the prediction-based protocols. Recall that the direct-delivery and random protocols did not replicate, thus had no data duplications.

Figure 6 shows both the median and mean delivery delays. All protocols show similar delivery delays in both mean and median measures for medium TTLs, but differ for long and short TTLs. With a 100-hour TTL, or unlimited TTL, epidemic routing had the shortest delays. The direct-delivery had the longest delay for unlimited TTL, but it had the shortest delay for the one-hour TTL.

The results seem contrary to our intuition: the epidemic routing protocol should be the fastest routing protocol since it spreads messages all over the network. Indeed, the figures show only the delay time for *delivered* messages. For direct delivery, random, and the probability-based routing protocols, relatively few messages were delivered for short TTLs, so many messages expired before they could reach their destination; those messages had infinite delivery delay and were not included in the median or mean measurements. For longer TTLs, more messages were delivered even for the direct-delivery protocol. The statistics of longer TTLs for comparison are more meaningful than those of short TTLs.

Since our message generation rate was low, the storage usage was also low in our simulation. Figure 7 shows the maximum and average of maximum volume (in KBytes) of messages stored

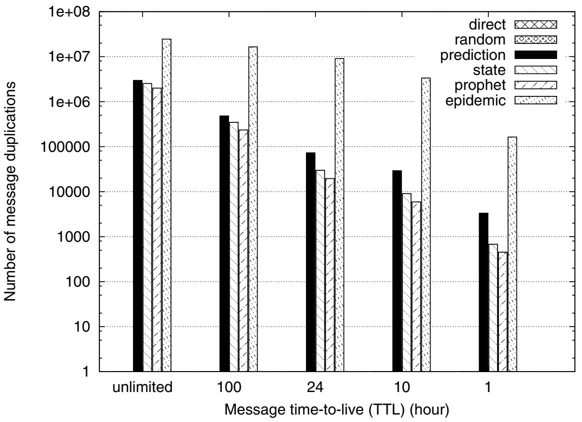

Figure 5: Message duplications (log scale). Direct and random protocols had no message duplications.

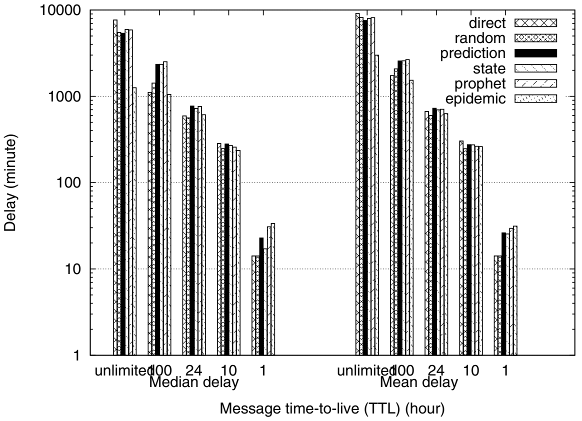

Figure 6: Median and mean delays (log scale).

in each node. The epidemic routing had the most storage usage. The message time-to-live parameter was the big factor affecting the storage usage for epidemic and prediction-based routing protocols.

We studied the impact of different parameters of our prediction-based routing protocol. Our prediction-based protocol was sensitive to several parameters, such as the probability threshold and the prediction window ΔT. Figure 8 shows the delivery ratios when we used different probability thresholds. (The leftmost value 0.01 is the value used for the other plots.) A higher probability threshold limited the transfer probability, so fewer messages were delivered. It also required fewer transmissions as shown in Figure 9. With a larger prediction window, we got a higher contact probability. Thus, for the same probability threshold, we had a slightly higher delivery ratio as shown in Figure 10, and a few more transmissions as shown in Figure 11.

5. RELATED WORK

In addition to the protocols that we evaluated in our simulation, several other opportunistic network routing protocols have been proposed in the literature. We did not implement and evaluate these routing protocols, because either they require domain-specific information (location information) [14, 15], assume certain mobility patterns [17], present orthogonal approaches [10, 24] to other routing protocols.

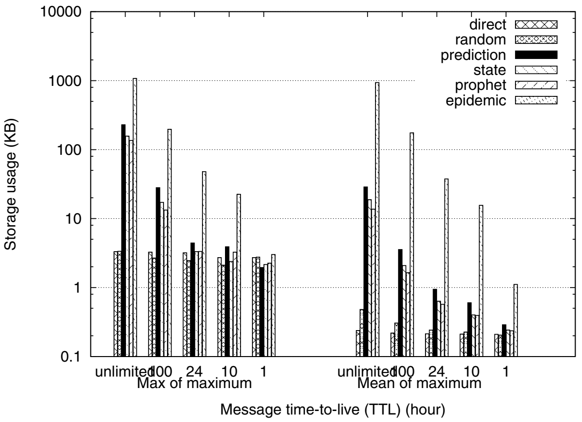

Figure 7: Max and mean of maximum storage usage across all nodes (log scale).

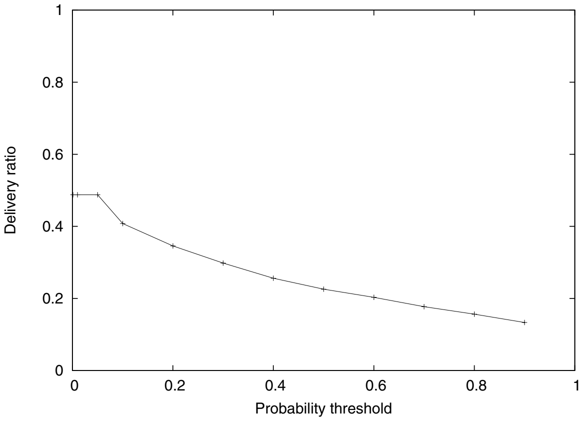

Figure 8: Probability threshold impact on delivery ratio of timely-contact routing.

LeBrun et al. [14] propose a location-based delay-tolerant network routing protocol. Their algorithm assumes that every node knows its own position, and the destination is stationary at a known location. A node forwards data to a neighbor only if the neighbor is closer to the destination than its own position. Our protocol does not require knowledge of the nodes' locations, and learns their contact patterns.

Leguay et al. [15] use a high-dimensional space to represent a mobility pattern, then routes messages to nodes that are closer to the destination node in the mobility pattern space. Location information of nodes is required to construct mobility patterns.

Musolesi et al. [17] propose an adaptive routing protocol for intermittently connected mobile ad hoc networks. They use a Kalman filter to compute the probability that a node delivers messages. This protocol assumes group mobility and cloud connectivity, that is, nodes move as a group, and among this group of nodes a contemporaneous end-to-end connection exists for every pair of nodes. When two nodes are in the same connected cloud, DSDV [19] routing is used.

Network coding also draws much interest from DTN research. Erasure-coding [10, 24] explores coding algorithms to reduce message replicas. The source node replicates a message m times, then uses a coding scheme to encode them in one big message. After replicas are encoded, the source divides the big message into k

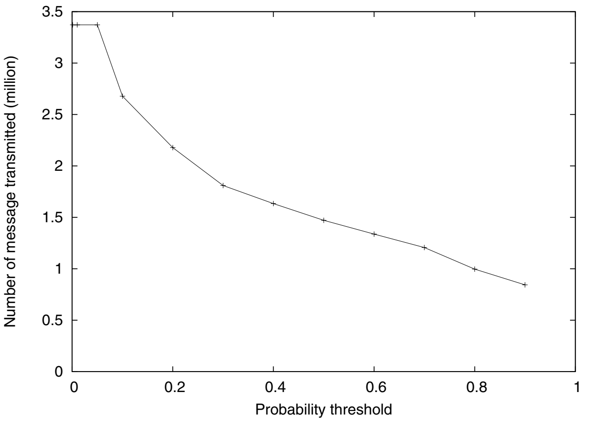

Figure 9: Probability threshold impact on message transmission of timely-contact routing.

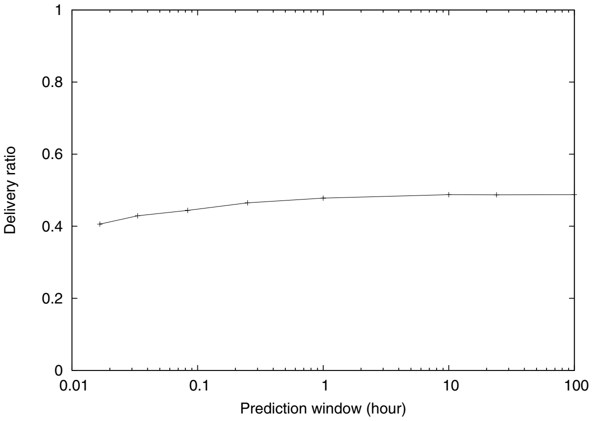

Figure 10: Prediction window impact on delivery ratio of timely-contact routing (semi-log scale).

blocks of the same size, and transmits a block to each of the first k encountered nodes. If m of the blocks are received at the destination, the message can be restored, where $m < k$. In a uniformly distributed mobility scenario, the delivery probability increases because the probability that the destination node meets m relays is greater than it meets k relays, given $m < k$.

6. SUMMARY

We propose a prediction-based routing protocol for opportunistic networks. We evaluate the performance of our protocol using realistic contact traces, and compare to five existing routing protocols.

Our simulation results show that direct delivery had the lowest delivery ratio, the fewest data transmissions, and no meta-data transmission or data duplication. Direct delivery is suitable for devices that require an extremely low power consumption. The random protocol increased the chance of delivery for messages otherwise stuck at some low mobility nodes. Epidemic routing delivered the most messages. The excessive transmissions, and data duplication, however, consume more resources than portable devices may be able to provide.

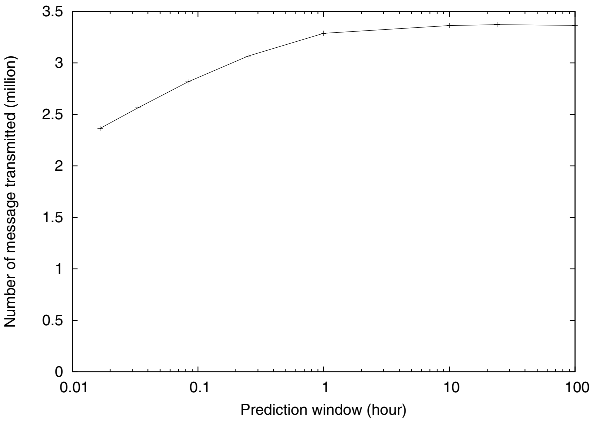

Figure 11: Prediction window impact on message transmission of timely-contact routing (semi-log scale).

None of these protocols (direct-delivery, random and epidemic routing) are practical for real deployment of opportunistic networks, because they either had an extremely low delivery ratio, or had an extremely high resource consumption. The prediction-based routing protocols had a delivery ratio more than 10 times better than that for direct-delivery and random routing, and fewer transmissions and less storage usage than epidemic routing. They also had fewer data duplications than epidemic routing.

All the prediction-based routing protocols that we have evaluated had similar performance. Our method had a slightly higher delivery ratio, but more transmissions and higher storage usage. There are many parameters for prediction-based routing protocols, however, and different parameters may produce different results. Indeed, there is an opportunity for some adaptation; for example, high priority messages may be given higher transfer and replication probabilities to increase the chance of delivery and reduce the delay, or a node with infrequent contact may choose to raise its transfer probability.

We only studied the impact of predicting peer-to-peer contact probability for routing in unicast messages. In some applications, context information (such as location) may be available for the peers. One may also consider other messaging models, for example, where messages are sent to a *location*, such that every node at that location will receive a copy of the message. Location prediction [21] may be used to predict nodes' mobility, and to choose as relays those nodes moving toward the destined location.

Research on routing in opportunistic networks is still in its early stage. Many other issues of opportunistic networks, such as security and privacy, are mainly left open. We anticipate studying these issues in future work.

7. ACKNOWLEDGEMENT

This research is a project of the Center for Mobile Computing and the Institute for Security Technology Studies at Dartmouth College. It was supported by DoCoMo Labs USA, the CRAWDAD archive at Dartmouth College (funded by NSF CRI Award 0454062), NSF Infrastructure Award EIA-9802068, and by Grant number 2005-DD-BX-1091 awarded by the Bureau of Justice Assistance. Points of view or opinions in this document are those of the authors and do not represent the official position or policies of any sponsor.

8. REFERENCES

[1] John Burgess, Brian Gallagher, David Jensen, and Brian Neil Levine. MaxProp: routing for vehicle-based

disruption-tolerant networks. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, April 2006.

[2] Scott Burleigh, Adrian Hooke, Leigh Torgerson, Kevin Fall, Vint Cerf, Bob Durst, Keith Scott, and Howard Weiss. Delay-tolerant networking: An approach to interplanetary Internet. *IEEE Communications Magazine*, 41(6):128–136, June 2003.

[3] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. *Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2(5):483–502, 2002.

[4] Andrew Campbell, Shane Eisenman, Nicholas Lane, Emiliano Miluzzo, and Ronald Peterson. People-centric urban sensing. In *IEEE Wireless Internet Conference*, August 2006.

[5] Augustin Chaintreau, Pan Hui, Jon Crowcroft, Christophe Diot, Richard Gass, and James Scott. Impact of human mobility on the design of opportunistic forwarding algorithms. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, April 2006.

[6] Kevin Fall. A delay-tolerant network architecture for challenged internets. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, August 2003.

[7] Tristan Henderson, David Kotz, and Ilya Abyzov. The changing usage of a mature campus-wide wireless network. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 187–201, September 2004.

[8] Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot. Pocket switched networks and human mobility in conference environments. In *ACM SIGCOMM Workshop on Delay Tolerant Networking*, pages 244–251, August 2005.

[9] Ravi Jain, Dan Lelescu, and Mahadevan Balakrishnan. Model T: an empirical model for user registration patterns in a campus wireless LAN. In *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 170–184, 2005.

[10] Sushant Jain, Mike Demmer, Rabin Patra, and Kevin Fall. Using redundancy to cope with failures in a delay tolerant network. In *Proceedings of the 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pages 109–120, August 2005.

[11] Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li-Shiuan Peh, and Daniel Rubenstein. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet. In *the Tenth International Conference on Architectural Support for Programming Languages and Operating Systems*, October 2002.

[12] David Kotz and Kobby Essien. Analysis of a campus-wide wireless network. *Wireless Networks*, 11:115–133, 2005.

[13] David Kotz, Tristan Henderson, and Ilya Abyzov. CRAWDAD data set dartmouth/campus. http://crawdad.cs.dartmouth.edu/dartmouth/campus, December 2004.

[14] Jason LeBrun, Chen-Nee Chuah, Dipak Ghosal, and Michael Zhang. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In *IEEE Vehicular Technology Conference*, pages 2289–2293, May 2005.

[15] Jeremie Leguay, Timur Friedman, and Vania Conan. Evaluating mobility pattern space routing for DTNs. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, April 2006.

[16] Anders Lindgren, Avri Doria, and Olov Schelen. Probabilistic routing in intermittently connected networks. In *Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR)*, pages 239–254, 2004.

[17] Mirco Musolesi, Stephen Hailes, and Cecilia Mascolo. Adaptive routing for intermittently connected mobile ad hoc networks. In *IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*, pages 183–189, June 2005. extended version.

[18] OLPC. One laptop per child project. http://laptop.org.

[19] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *Computer Communication Review*, pages 234–244, October 1994.

[20] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, February 1999.

[21] Libo Song, David Kotz, Ravi Jain, and Xiaoning He. Evaluating next-cell predictors with extensive Wi-Fi mobility data. *IEEE Transactions on Mobile Computing*, 5(12):1633–1649, December 2006.

[22] Jing Su, Ashvin Goel, and Eyal de Lara. An empirical evaluation of the student-net delay tolerant network. In *International Conference on Mobile and Ubiquitous Systems (MobiQuitous)*, July 2006.

[23] Amin Vahdat and David Becker. Epidemic routing for partially-connected ad hoc networks. Technical Report CS-2000-06, Duke University, July 2000.

[24] Yong Wang, Sushant Jain, Margaret Martonosia, and Kevin Fall. Erasure-coding based routing for opportunistic networks. In *ACM SIGCOMM Workshop on Delay Tolerant Networking*, pages 229–236, August 2005.

[25] Yu Wang and Hongyi Wu. DFT-MSN: the delay fault tolerant mobile sensor network for pervasive information gathering. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, April 2006.