Evaluating Contacts for Routing in Highly Partitioned Mobile Networks

Chao Chen
Department of Engineering
Indiana University - Purdue University
Fort Wayne, IN 46805
chen@engr.ipfw.edu

Zesheng Chen
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
zchen@ece.gatech.edu

ABSTRACT

The movement of real users often follows some patterns that can be characterized by certain statistical metrics of the contacts. Such metrics are useful for nodes to make routing decisions, especially in sparse mobile ad hoc networks where instantaneous network connectivity is rare and messages are delivered using store-carry-forward routing. None of the statistical metrics used in existing routing algorithms considers the dependency between neighboring contacts, thus resulting in an inaccurate estimation of message delays over paths. In this paper, a new metric called the *expected dependent delay* that characterizes the expected delay of a contact dependent on the previous hop is proposed. With this new metric, the closed-form expression for the expected delay of a path is given and can be further simplified in highly partitioned mobile networks. Simulation results show that the proposed algorithm has a significant improvement in terms of the message delay than those that consider only the delivery probability or the expected delay.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing Protocols

General Terms

Algorithms, Performance, Theory

Keywords

Delay Tolerant Networks, Routing, Dependent Delay

1. INTRODUCTION

In ad hoc networks where mobile nodes are sparsely distributed, network partitioning is frequent, and nodes connect only intermittently. Such networks also belong to a broader notion of Delay Tolerant Networks (DTNs) [1], where a complete path from a source to a destination does not always exist or such a path is highly unstable after it has been discovered. To enable data delivery in mobile networks with intermittent connectivity, a new routing approach called *store-carry-forward* (SCF) routing [2] has been used. In SCF routing, a node may need to buffer a message for a certain amount of time and wait for an appropriate next-hop to appear in its communication range. To make good routing decisions, a node needs to gather certain information to decide whether a neighbor node is an "appropriate" next-hop; if not, how long it should "hold on" to a message until a good candidate shows up in range.

A number of SCF routing protocols have been proposed [2]. When future node movement is known ahead of time, depending on the amount of knowledge about network topology characteristics and traffic demands, several routing algorithms can be selected [3]. On the other hand, when the node movement schedule is not known beforehand, forwarding decisions can be made with the help of past records of meeting with other nodes [4, 5] or certain knowledge about node movement patterns [6, 7]. In networks where node mobility is not completely random, such knowledge can be used to predict the future movement and improve the message delivery performance.

In this paper, we first explore in Section 2 different types of contact statistics that have been used to help make routing decisions. Specifically, the delivery probability and the expected delay are evaluated through examples and analysis in Section 3. We find that the delivery probability cannot accurately characterize the delay property of a contact. On the other hand, it is shown in Section 4 that the expected delay lacks the ability to model the dependency among contacts, which is shown through simulations to exist in sparse networks with non-random node movement. A new metric called the *expected dependent delay*, which exploits the spatial dependency between neighboring contacts, is proposed. Unlike the expected delay, the estimation of the expected dependent delay does not assume that message arrivals at all intermediate nodes are uniform. Furthermore, the expected delay of a path can be precisely formulated by a function of the expected delay and the expected dependent delay of the contacts along the path. In highly partitioned mobile networks where two consecutive hops on a path rarely occur at the same time, the path delay can be further approximated by a simple form. The performance of routing algorithms using different statistical metrics in single-copy routing is evaluated by simulations in Section 5. Simulations verify that our method gives a better estimation of the path delay, where the saving in message delay can be as large as 14%-22% even in a network with weak contact dependency. The

MobiOpp'07, June 11, 2007, San Juan, Puerto Rico, USA.

improvement in routing performance is also robust against the effects of the limited buffer size and the increase of traffic load. Some related work is covered in Section 6. Finally, conclusions and future work are given in Section 7.

2. CONTACT INVARIANTS

In networks with intermittent connectivity, the links that connecting communication entities (i.e., nodes) can go up and down over time, due to node mobility, failures, or other reasons. A *contact* occurs when a link is up for some time and is marked by its start and end times, capacity, latency, endpoints, and direction [8].

In most DTN applications, node movement is not totally random. For example, the nodes can be PDAs that are carried by people [9]. Node movement is affected by human decisions and socialization behaviors. In the UMass DieselNet project [10], mobile nodes are buses that run on schedule and follow pre-determined routes. When the node movement is non-random, some statistical metrics of the contacts can be retrieved to characterize the regularity of the connectivity pattern. Such metrics are referred to as "*invariants*" in this paper. Based on the statistical invariants, future contacts can be predicted to some extent according to past contact records. In this section, different types of contact invariants that have been used by existing SCF routing protocols are reviewed.

2.1 Delivery Probability

Delivery probability $P(a,b) \in [0,1]$ is a probability metric describing the likelihood that node a will be able to deliver a message to node b. In the literature, there are generally two groups of methods to estimate the delivery probability: contact-frequency based estimation or contact-availability based estimation.

Using the *contact-frequency based estimation*, the delivery probability is increased whenever two nodes meet each other either directly [4] or indirectly via other intermediate nodes [4, 10]. Furthermore, [5] uses a Kalman Filter to adaptively predict the delivery probability based on the context information such as the rate of changes of connectivity. In all the above cases, the delivery probability is mainly affected by the contact frequency[1], i.e., how often two nodes meet each other. When selecting a next-hop node to forward a message, a node would prefer the neighbor that meets the destination more often.

To include the effect of contact duration, another approach defines the delivery probability $P(a,b)$ as follows [11, 12]:

$$P(a,b) = \frac{\sum(\text{time when link } ab \text{ is up})}{\text{length of time window}}. \quad (1)$$

The above formula calculates the percentage of time that link ab is up in the time window, i.e., the contact availability. Therefore, this approach is referred to as the *contact-availability based estimation* in this paper.

2.2 Expected Delay

The *expected delay* was first used in [3] as an option to compute the cost of a path when the exact contact schedule is unknown. Such routing algorithm is called the mean expected delay (MED), where the path with the minimum accumulated expected delay of contacts is selected to be the shortest path. Assuming that all message arrival times are equally likely, the expected delay of a contact between nodes a and b shown in Fig. 1 is estimated as follows [13]:

$$W(a,b) = \frac{\sum_{i=1}^{m} d_i^2}{2T}, \quad (2)$$

where T is the duration of the time window, m is the total number of disconnected periods between a and b within T, and d_i is the duration of the ith disconnection period.

[1]The PROPHET protocol [4] also reduces the delivery probability with a decaying factor if two nodes have not met each other for a long time.

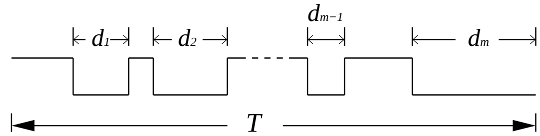

Figure 1: Contacts between two nodes.

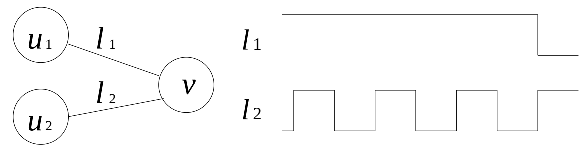

Figure 2: A counter example for the contact-frequency based estimation.

3. DELIVERY PROBABILITY VERSUS EXPECTED DELAY

Given different metrics retrieved from the contact records, which one better characterizes the invariant in contact statistics and can be used to better predict future contacts? In this section, we evaluate the delivery probability and the expected delay through examples and analysis.

3.1 Limitation of Delivery Probability

Although to some extent, the contact frequency reflects how probable two nodes meet each other, it does not tell how long each contact lasts. As a counter example, Fig. 2 shows the contact record of two links l_1 and l_2, where l_i ($i = 1, 2$) is between node u_i and destination v, and u_i is a node that is currently in contact with the node carrying a message. As seen in the figure, l_1 has been up for a long time and then turns down, whereas l_2 turns up and down more frequently. According to the rules adopted by the contact-frequency based estimation, u_2 has more chances to increase its delivery probability to v than u_1. But intuitively, u_1 should be a better candidate than u_2 to forward a message to destination v.

On the other hand, when using Eqn. (1) to estimate the delivery probability, the contact frequency does not play any role, which would result in some discrepancy. For example, suppose that the contact records of two links l_1 and l_2 are illustrated in Fig. 3, where l_i ($i = 1, 2$) is the link between u_i and destination v, as in the previous example. The contact periods of l_1 and l_2 are τ and 2τ, respectively. According to Eqn. (1), the delivery probability of two links are both 0.5. If based on the delivery probability, every message has an equal chance to be forwarded to u_1 and u_2. However, the expected delays of links l_1 and l_2 are $W_1 = \frac{\tau}{4}$ and $W_2 = \frac{\tau}{2}$,

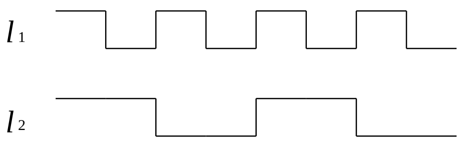

Figure 3: A counter example for the contact-availability based estimation.

respectively. Therefore, all messages should be transmitted to node u_1 to achieve a shorter delay.

3.2 Comparison between Two Invariants

A quantitative comparison between the delivery probability and the expected delay is given below. Suppose that the delivery probability (p) between nodes a and b in Fig. 1 is estimated based on the contact availability using Eqn. (1). Then,

$$p = P(a, b) = 1 - \frac{\sum_{i=1}^{m} d_i}{T}. \quad (3)$$

Note that since $d_i \geq 0$,

$$\sum_{i=1}^{m} d_i^2 \leq \left(\sum_{i=1}^{m} d_i\right)^2 = (1-p)^2 T^2. \quad (4)$$

The equality holds when $m = 1$, i.e., only one contact opportunity exists during the observation window T. Moreover,

$$\sum_{i=1}^{m} d_i^2 \geq \frac{1}{m}\left(\sum_{i=1}^{m} d_i\right)^2 = \frac{(1-p)^2}{m} T^2. \quad (5)$$

The above inequality is derived by the Cauchy-Schwarz inequality. The equality holds if and only if $d_i = \frac{(1-p)T}{m}$, for $i = 1, 2, \cdots, m$. In other words, when the distribution of link down time intervals is uniform, $\sum_{i=1}^{m} d_i^2$ obtains the minimum value. Combining Inequalities (4) and (5) and using Eqn. (2),

$$\frac{(1-p)^2}{2m} T \leq W(a, b) \leq \frac{(1-p)^2}{2} T. \quad (6)$$

Therefore, when two links have the same delivery probability based on the contact-availability estimation, their expected delays can still differ by m times. If m is large, such difference is significant, which indicates that the delivery probability based on contact availability is not an accurate metric. This motivates us to evaluate contacts from the delay aspect, instead of from the delivery-probability aspect.

4. EXPECTED DEPENDENT DELAY

According to the algorithm in [3, 13], the expected delay of a path is calculated as the sum of the expected delay of all contacts on the path. The assumption behind the calculation is that the delays of contacts on the path are independent. Does this assumption hold for highly partitioned mobile networks where node movement is not random? If not, how far is the above calculation from the actual path delay?

4.1 A Motivating Example

A motivating example is illustrated in Fig. 4. Suppose that the communication range of three nodes a, b, and c is R. a and c are two fixed nodes that are located with a

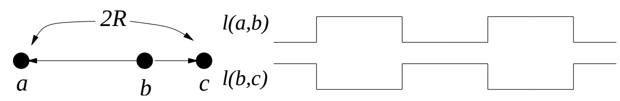

Figure 4: A motivating example of three nodes with dependent contacts.

Table 1: Parameters used in the simulated network

Number of hubs	$M = 10$
Radius of each hub area	$R_h = 100$
Number of nodes	$n = 50$
Communication range of nodes	$R_n = 150$
Number of hubs in a node's hublist	$N_m = 5$
Minimum speed outside hubs	$V_{min} = 10$
Maximum speed outside hubs	$V_{max} = 50$
Maximum time inside a hub	$T_{max} = 50$

distance of $2R$, i.e., a and c can never communicate directly. Node b moves back and forth between node a and c with a fixed speed. $l(a, b)$ and $l(b, c)$ are contacts between a and b, b and c, respectively. Whenever $l(a, b)$ is up, $l(b, c)$ must be down, and vice versa. In other words, the status of one link would tell the status of the other, i.e., $l(a, b)$ and $l(b, c)$ are dependent.

If Eqn. (2) is used to calculate the expected delay, then $W(a, b) = W(b, c) = \frac{\tau}{4}$, where τ is the time that b takes to move from a to c and is also the duration of both contacts. If the expected delay of path abc is calculated by summing up the expected delays of $l(a, b)$ and $l(b, c)$, then $W(abc) = W(a, b) + W(b, c) = \frac{\tau}{2}$. When we take the dependency of these two links into consideration, however, the expected delay of path abc should be $W(abc) = 0.5 \times W(abc|l(a, b) \text{ is up}) + 0.5 \times W(abc|l(a, b) \text{ is down}) = 0.5 \times \frac{\tau}{2} + 0.5 \times (\frac{\tau}{2} + \tau) = \tau$, which is twice of $\frac{\tau}{2}$. From this simple example, we can see that the dependency between contacts cannot be ignored when evaluating the path delay, especially when the node movement follows some pattern and contacts between different node pairs are dependent.

4.2 Dependency between Neighboring Contacts

To verify our perception, we evaluate the correlation coefficients of link pairs in a simulated network model. This network consists of n nodes in a 1000×1000 area. M hubs are randomly located in the square, and each hub covers a circular shape with a range of R_h. These circular hub areas do not overlap with each other. The hubs represent popular places that the nodes will stop by. The communication range of all nodes is R_n. Among the n nodes, M are stable at the center of each hub; the rest are variable in location. The movement of the mobile nodes follows a certain social behavior. Specifically, each moving node randomly selects N_m hubs as its waypoints and forms a hublist. Initially, every active node is randomly located in the entire square area and moves toward the first waypoint on its hublist with a random speed in $[V_{min}, V_{max}]$. Once a node reaches the target hub, it chooses a slower speed in $(0, V_{min}]$ and moves inside the hub area for a random time between $(0, T_{max}]$. After the timer expires, the node moves to the next waypoint on its hublist. The simulation parameters are given in Table 1. With this setting, at any time instant, there

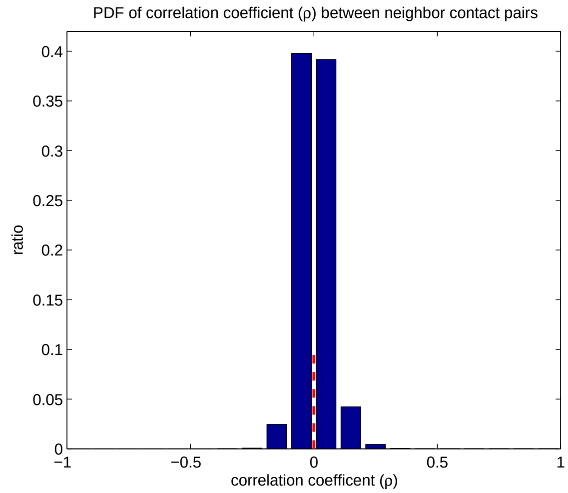

Figure 5: PDF of correlation coefficient (ρ) between neighbor contact pairs.

exist only a few contacts, and the entire network is highly partitioned.

We compute the correlation coefficient (ρ) of every two neighboring contacts. Suppose that the status of a contact is either 1 (up) or 0 (down), i.e., the contact status is a Bernoulli random variable, the correlation coefficient ($\rho_{1,2}$) of two neighboring contacts l_1 and l_2 (l_1 and l_2 have a common endpoint) is calculated as

$$\rho_{1,2} = \frac{\text{cov}(l_1, l_2)}{\sigma_1 \sigma_2} = \frac{\mu_{1,2} - p_1 p_2}{\sqrt{(p_1 - p_1^2) \cdot (p_2 - p_2^2)}}, \quad (7)$$

where $\mu_{1,2} = P\{l_1 = l_2 = 1\}$ is the probability that both l_1 and l_2 are up at the same time, p_i ($i = 1, 2$) is the delivery probability of contact l_i and is calculated using Eqn. (1). The correlation coefficient $\rho_{1,2} \in [-1, 1]$ measures the dependency of the statuses of two links l_1 and l_2. When $\rho_{1,2} = 0$, l_1 and l_2 are uncorrelated. Generally, $\rho_{1,2} = 0$ does not lead to independency between l_1 and l_2; however, for Bernoulli random variables like the contact status in our case, $\rho_{1,2} = 0$ also means that l_1 and l_2 are independent.

Twenty different mobile network models are generated using different seeds. After the simulation runs for 10000 time units for all 20 network models, the PDF of the correlation coefficient averaged over 20 runs is shown in Fig. 5. The dashed line in the center depicts the percentage of neighbor contact pairs whose correlation coefficient is zero, i.e., independent of each other. The results show that under the given mobile network scenario, only 9.48% of neighbor contact pairs are independent, and a majority (86.17%) of correlation coefficients span over the range of $[-0.3, 0.4]$, excluding 0. Therefore, the dependency between neighbor contact pairs cannot be ignored. If a path has multiple hops, the expected delay of the path may not necessarily be the sum of the expected delay of each hop on the path. On the other hand, Fig. 5 also tells that close to 90% of the correlation coefficients are in $[-0.1, 0.1]$, including 0. Thus, the dependency between neighboring contacts in this simulated network model is not strong.

We also check μ for neighbor contact pairs. In our generated network model, the probability that two neighboring links are up at the same time is extremely low. Specifically, above 99% of the pairs have a small probability (< 0.05) to be up at the same time.

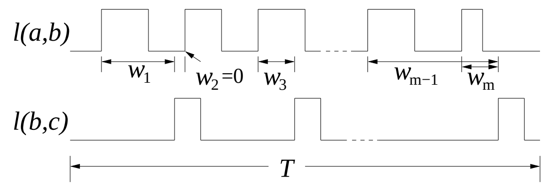

Figure 6: Two neighboring contacts.

4.3 Expected Dependent Delay

To characterize the dependency between neighboring contacts, we define the *expected dependent delay* as the expected delay of a contact depending on a previous hop. Specifically, the following equation is used to estimate the expected dependent delay of contact bc depending on node a, where both a and c are b's neighbors:

$$W_{|a}(b, c) = \frac{\sum_{i=1}^{m} w_i}{m}, \quad (8)$$

where w_i is the time waiting for contact bc from the time that contact ab goes up, and m is the number of time that contact ab goes up in the time window, as shown in Fig. 6. If contact bc is up when ab rises up the ith time, $w_i = 0$. Eqn (8) takes the dependency of contacts into consideration. If links ab and bc have a negative correlation coefficient, then $P\{l(b,c) = 0 | l(a,b) = 1\} > P\{l(b,c) = 0\}$, and the expected delay of contact bc since contact ab is up is longer than the "independent" expected delay, i.e., $W_{|a}(b,c) > W(b,c)$. This can also be confirmed by the example in Fig. 4, where the expected dependent delay of contact bc on a is $W_{|a}(b,c) = \tau$, which is longer than the expected delay $W(b,c) = \frac{\tau}{4}$. On the other hand, if the correlation coefficient of ab and bc is positive, then $W_{|a}(b,c) < W(b,c)$. In the case that the movement of a, b, c is independent, contact ab can hardly tell any information about contact bc, then $W_{|a}(b,c) = W(b,c)$, and Eqn. (2) would be a good estimation of the delay.

4.4 Expected Delay of a Path

In this section, we attempt to arrive at the closed-form expression of the expected delay of a path, given the expected delay and the expected dependent delay of contacts on the path.

Suppose that $\mathcal{P}_n = (u_0, u_1, u_2, ..., u_n)$ is a path consisting of n hops ($n \geq 2$), where u_i is a node along the path. l_i is the contact between node u_{i-1} and u_i, $i = 1, ..., n$. We use $W(l_i)$ to denote expected delay $W(u_{i-1}, u_i)$ and $W_{|u_{i-2}}(l_i)$ for expected dependent delay $W_{|u_{i-2}}(u_{i-1}, u_i)$. $W(\mathcal{P}_n)$ is the expected delay of path $\mathcal{P}_n$. The following equation is the closed-form expression of the expected delay of $\mathcal{P}_n$:

$$\begin{aligned} W(\mathcal{P}_n) &= W(\mathcal{P}_{n-1}) + \left(1 - \tfrac{\mu_{n-2,n-1}}{p_{n-2}}\right) \times W_{|u_{n-2}}(l_n) \\ &+ \left(\tfrac{\mu_{n-2,n-1}}{p_{n-2}} \cdot \tfrac{p_{n-1} - \mu_{n-1,n}}{p_{n-1}}\right) \times \tfrac{W(l_n)}{1 - p_n}, \end{aligned} \quad (9)$$

where $W(\mathcal{P}_1) = W(l_1)$, p_0 and l_0 are introduced for convenience, and $p_0 = 1, \mu_{0,1} = p_1$.

PROOF: When $n = 2$, let $W^0(\mathcal{P}_2)$ and $W^1(\mathcal{P}_2)$ be the expected delay of $\mathcal{P}_2$, given that the first link l_1 is down or up when a message arrives. Using the formula of $W(l_i|l_i = 0) = \frac{W(l_i)}{1-p_i}$ and $P\{l_2 = 0|l_1 = 1\} = \frac{p_1-\mu_{1,2}}{p_1}$,

$$\begin{array}{l} W^0(\mathcal{P}_2) = \frac{W(l_1)}{1-p_1} + W_{|u_0}(l_2) \\ W^1(\mathcal{P}_2) = \frac{p_1-\mu_{1,2}}{p_1} \times \frac{W(l_2)}{1-p_2} \end{array}. \quad (10)$$

The expected delay of a 2-hop path $\mathcal{P}_2$ can be calculated as:

$$\begin{array}{l} W(\mathcal{P}_2) = (1-p_1) \times W^0(\mathcal{P}_2) + p_1 \times W^1(\mathcal{P}_2) \\ = W(l_1) + (1-p_1) \times W_{|u_0}(l_2) + (p_1 - \mu_{1,2}) \times \frac{W(l_2)}{1-p_2}. \end{array} \quad (11)$$

Since $W(l_1) = W(\mathcal{P}_1)$ and $\frac{\mu_{0,1}}{p_0} = p_1$, the above result matches Eqn. (9) when $n = 2$.

When $n \geq 3$,

$$\begin{array}{l} W(\mathcal{P}_n) = W(\mathcal{P}_{n-1}) + \\ P\{l_{n-1} = 0|l_{n-2} = 1\} \times W_{|u_{n-2}}(l_n) + \\ P\{l_{n-1} = 1|l_{n-2} = 1\} \times P\{l_n = 0|l_{n-1} = 1\} \times \frac{W(l_n)}{1-p_n}, \end{array} \quad (12)$$

where $P\{l_{n-1} = 0|l_{n-2} = 1\} = 1 - \frac{\mu_{n-2,n-1}}{p_{n-2}}$ reflects the probability that a message stops at node u_{n-2} and waits for contact l_{n-1} to go up to reach node u_{n-1}; $P\{l_{n-1} = 1|l_{n-2} = 1\} \cdot P\{l_n = 0|l_{n-1} = 1\} = \frac{\mu_{n-2,n-1}}{p_{n-2}} \cdot \frac{p_{n-1}-\mu_{n-1,n}}{p_{n-1}}$ is the probability that contacts l_{n-2} and l_{n-1} are both up but l_n is down, i.e., a message doesn't stop at node u_{n-2} and the message arrival at node u_{n-1} is uniform in l_{n-1}'s up period. Plugging the probabilities back into Eqn. (12) leads to Eqn. (9). ∎

We can also verify the result for the 2-hop path in the example in Fig. 4, where $\mu_{l(a,b),l(b,c)} = 0, p_1 = p_2 = 0.5, W(a,b) = W(b,c) = \frac{\tau}{4}$, and $W_{|a}(b,c) = \tau$. The path delay is calculated using Eqn. (11) as $W(abc) = \frac{\tau}{4} + (1-0.5) \times \tau + \frac{0.5}{1-0.5} \times \frac{\tau}{4} = \tau$, which matches the analysis in Section 4.1.

In highly partitioned mobile networks, it is very rare that two consecutive hops on a path occur at the same time, which is also the reason that the SCF routing is proposed. Therefore, when $n \geq 3$, $\mu_{n-2,n-1}$ is close to zero, the third term in Eqn. (9) is very small and can be ignored, and the second term can be approximated by $W_{|u_{n-2}}(l_n)$, which results in the following formula:

$$W(\mathcal{P}_n) = W(\mathcal{P}_{n-1}) + W_{|u_{n-2}}(l_n). \quad (13)$$

The delay of a 2-hop path can be approximated by

$$W(\mathcal{P}_2) = W(l_1) + (1-p_1) \times W_{|u_0}(l_2) + \frac{p_1}{1-p_2} \times W(l_2). \quad (14)$$

When p_1 and p_2 are small, $W(\mathcal{P}_2) \approx W(l_1) + W_{|u_0}(l_2)$, which also satisfies Eqn. (13).

As a special case, if all nodes move independent of each other, $\mu_{i,i+1} = p_i p_{i+1}$ and $W_{|u_{i-2}}(l_i) = W(l_i)$. Using Eqn. (9), the expected delay of a path is reduced to the sum of the expected delay of all hops along the path, i.e., $W(\mathcal{P}_n) = \sum_{i=1}^{n} W(l_i)$.

4.5 Discussion

If the expected path delay is calculated as in [3, 13], the dependency of neighboring contacts is not taken into consideration. Furthermore, the calculation of the expected delay following Eqn. (2) assumes that the message arrivals at all intermediate nodes are uniform. In sparse mobile networks where the contact opportunity is infrequent, however, messages usually get "stuck" in intermediate nodes and wait for the next-hop to appear. Once an appropriate next-hop appears in range, all messages for this neighbor are forwarded immediately. Therefore, for intermediate nodes (other than the source), message arrivals would be highly non-uniform, more specifically, clustered at the time when the contact turns up. Hence, simply adding up the expected delay of all links as the expected path delay is not a good estimation.

Our estimation of the expected path delay considers the dependency of contacts on the previous hop. Moreover, using Eqn. (8), the dependent delay starts to accumulate ever since the previous contact is available. For intermediate nodes where message arrivals are clustered in the beginning of a contact, we believe that such estimation is more accurate.

5. PERFORMANCE EVALUATION

In this section, we evaluate different contact invariants and their effects on routing performance. Mobile network models are generated using the same parameters as in Section 4.2. As observed previously, the probability that two neighboring contacts are both up is very low. Specifically, over 99% of the pairs have a probability smaller than 0.05 to be up at the same time. Therefore, the approximation in Eqn. (13) is well qualified.

In each simulation run, a mobile network model is generated using a new seed. In the first 10000 time units, nodes gather contact statistics required for each routing algorithm. After this initial statistics gathering period, messages are generated following the Poisson arrival. Each message randomly selects two fixed nodes at different hubs as its source and destination. A shortest path is calculated using the Dijkstra's algorithm with different methods of computing the link cost. Specifically, the following algorithms are implemented:

1. Maximum delivery probability (MDP) [11]: The cost for a path is $\sum 1/P(u_i, u_{i+1})$, where $P(u_i, u_{i+1})$ is the delivery probability of each contact on the path and is estimated based on the contact availability using Eqn. (1).

2. Minimum expected delay (MED) [3]: The path cost is the sum of the expected delay $W(u_i, u_{i+1})$ of all contacts on the path, and $W(u_i, u_{i+1})$ is estimated using Eqn. (2).

3. Minimum expected dependent delay (MEDD): The path cost is the sum of the expected delay of the first hop and the expected dependent delay $W_{|u_{i-1}}(u_i, u_{i+1})$ of the remaining contacts on the previous hop using Eqn. (13). $W_{|u_{i-1}}(u_i, u_{i+1})$ is estimated by Eqn. (8).

For the MEDD algorithm, we modified the Dijkstra's algorithm to calculate the link cost, taking account of the dependency on the previous hop. Moreover, to avoid abrupt changes in the estimation, the expected dependent delay of a contact is used only if there is enough number of records available. Specifically, we use Eqn. (8) to calculate the expected dependent delay only if $m \geq 8$. Otherwise, the expected delay calculated by Eqn. (2) is used instead.

To exclude the influence of other factors, we make the following assumptions in our simulation:

- Each node has contention-free access to the wireless channel. At the time scales that we are considering,

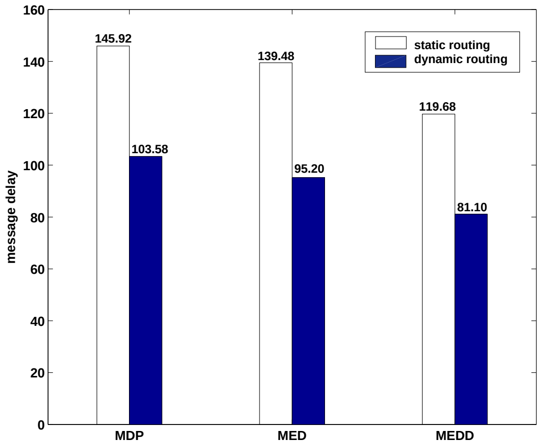

Figure 7: Comparison of delay performance with infinite buffer size.

MAC level retransmissions and delays can be safely ignored.

- Contact capacity is unlimited so that the transmission delay can be ignored.
- The node communication range is not too far so that the propagation delay can be ignored.
- Single-copy routing is considered; each message follows the shortest path calculated by the Dijkstra's algorithm.

5.1 Scenario I: Infinite Buffer Size

In the first scenario, we assume that every node has an infinite buffer size so that messages are not discarded because of buffer overflow. In each simulation run, 100 messages are inserted into the network, and messages follow Poisson arrival process with an average interval of 10 time units. A simulation run is stopped when all messages are successfully delivered to their destinations. We measured the average message delay for 100 independent runs.

We consider two difference cases. In the first case (referred to as "*static routing*"), the statistics gathered in the first 10000 time units are used to calculate the link costs. After this first information gathering phase, these statistics are not updated. Hence, the shortest path between a source-destination pair does not change, and a message always follows the shortest path calculated at the source. The second case is called "*dynamic routing*", where nodes keep gathering new information after the initial 10000 time units. A message will follow an updated shortest path as it passes a new node on the way. The "per-contact routing" concept in [13] is also adopted, where the routing table is recomputed each time when a contact arrives and the links to the neighbors that are currently in contact are "short-circuited" (the link cost is set to zero).

The comparison of the average message delay using different algorithms under these two cases is given in Fig. 7. The results show that MEDD based on the expected dependent delay achieves the best performance in terms of the message delay. Compared with MDP and MED, the delay improvement in static routing case is as large as 17.98% and 14.08%, respectively. Moreover, when dynamic routing is adopted, the message delay performance can be further improved to 21.70% saving than MDP and 19.89% saving than MED. This improvement is the result of the knowledge of the dependent delay, which reflects the dependency of the node movement. It should be noted that in our simulation model, the dependency of neighboring contacts is not strong. As mentioned in Section 4.2, about 86.17% of the neighbor contact pairs have a nonzero correlation coefficient in $[-0.3, 0.4]$, but most of the values fall in $[-0.1, 0.1]$. In such a network model with weak contact dependency, the performance improvement of MEDD over MDP and MED is yet apparent. If the dependency is higher, we expect MEDD to have even greater improvement than other algorithms.

5.2 Scenario II: Limited Buffer Size

In the second simulation scenario, we evaluate the effect of the limited buffer size on routing performance of MEDD. We first set the message interval to 10 and change the buffer limit from 1 message to 10 messages. If a new message is forwarded to a node and finds the buffer full, this message is dropped. That is, the buffer management policy is first-in-first-out (FIFO). Similar to the previous setting, nodes gather contact statistics in the first 10000 time units. After that, messages are generated following the Poisson arrival until 15000 time units. We measured the average delay of successfully delivered messages and the successful message delivery ratio, where the only failure recorded is message dropping as the result of buffer overflow. The results are averages of 10 independent runs with different seeds.

The message delay and success ratio results of MDP, MED, and MEDD for both static routing and dynamic routing are depicted in Fig. 8. When the buffer limit is at least 5, the increase of buffer limits does not have any effect on the message delivery performance, i.e., no message is dropped because of buffer overflow. In terms of delay, MEDD performs consistently the best under both static and dynamic routing. When the delivery ratio is considered, MEDD always has a larger ratio than MDP and MED in static routing. Although MEDD has a slightly smaller success ratio in dynamic routing when the buffer limit is small (≤ 2), the delay improvement of MEDD is consistent in all cases (at least 18%-25% shorter than MDP and at least 13%-18% shorter than MED). Therefore, the overall message delivery performance of MEDD can be guaranteed, especially when a sufficiently large buffer size is provided (≥ 5 in the simulated network model).

Next, we fix the buffer limit to 3 and change the message arrival interval. The delay and ratio performance of the three algorithms is given in Fig. 9. The figures show that the good performance of MEDD both in message delay and delivery ratio is maintained even as the traffic load grows. The average saving in delay for all cases is 24.45% than MDP and 20.10% for MED.

The simulation results verify that MEDD improves the delay performance than MDP and MED. MEDD is also robust against the effects of the limited buffer size and the increase of traffic load.

6. RELATED WORK

In ad hoc networks with opportunistic connectivity, to achieve end-to-end message delivery between disconnected parts of the network, nodes often need to carry messages

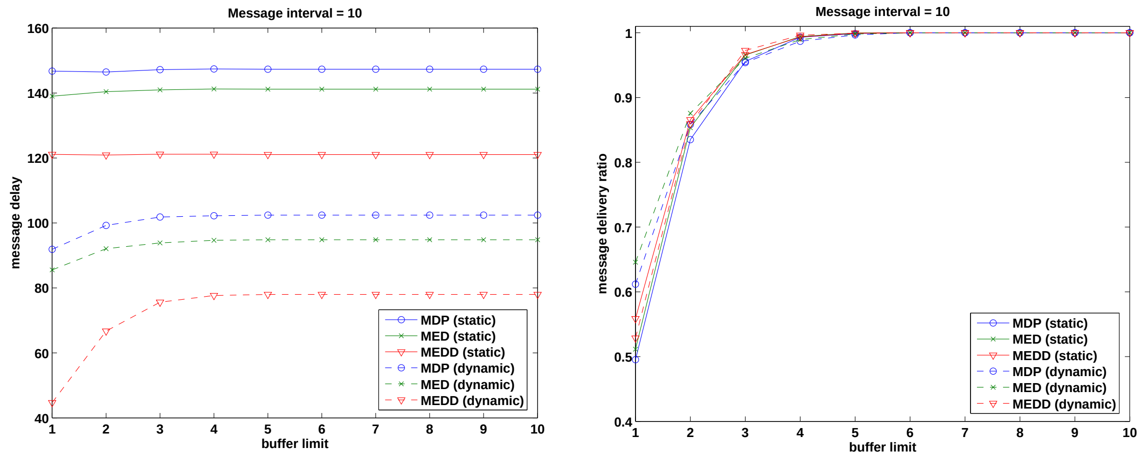

Figure 8: Delay and ratio performance under different buffer limits with fixed message interval = 10.

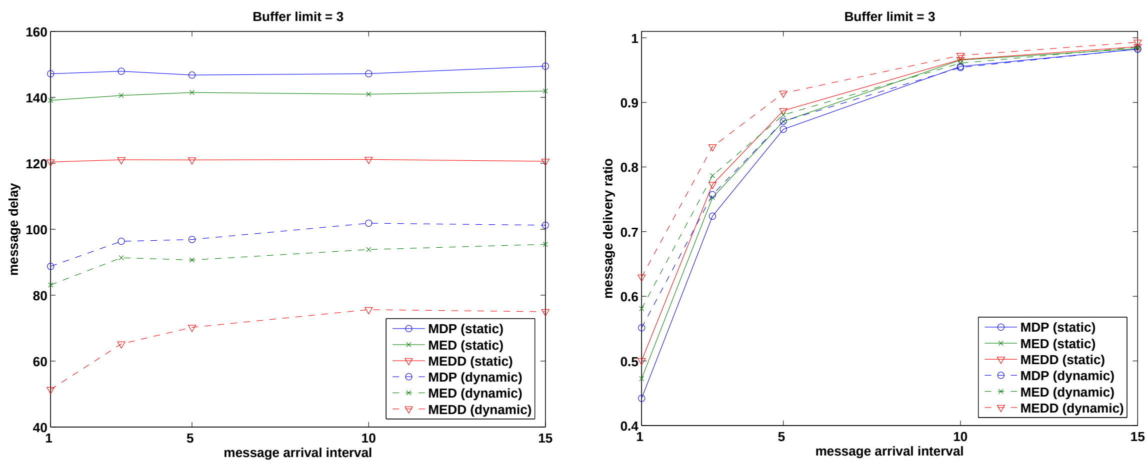

Figure 9: Delay and ratio performance under different message intervals with fixed buffer limit = 3.

around while moving. Therefore, the knowledge about user mobility and the opportunities for user devices to interact when they are close to each other becomes very important for forwarding decisions.

Some work has focused on analyzing mobility characteristics on data traces from universities or conference environments [9, 14, 15, 16]. These traces suggest there are some popular locations that nodes tends to visit often and spend most of the time. Moreover, node movement is not independent and identically distributed. One possible reason is that mobile devices are usually carried by humans, so the movement of these devices is affected by human decisions and socialization behavior, which are often dependent on each other. A few mobility models [4, 17, 18, 19] are proposed to capture the above-mentioned real-world movement characteristics. The simulation model used in this paper is in essence very similar to the one in [19] where nodes switches between states of moving inside local hotspots and roaming outside.

Another related work is the *utility-based routing* in intermittently connected ad hoc networks, where each node i maintains a utility function $U_i(j)$ for every other node in the network. $U_i(j)$ reflects the probability that node i will deliver a message to node j, and it may be based on a number of different parameters [5, 20] (e.g., encounter history, mobility, rate of change of connectivity, friendship index with other nodes, current energy level, etc.). If the current carrier meets a node with a higher utility, the message is transferred to the neighbor node. Both the utility-based routing and our approach (MEDD) are based on prediction of future activity according to available statistics. Unlike MEDD, the forwarding in utility-based routing is based on local improvement of delivery probability. Our approach, however, aims to optimize the end-to-end delay performance, where the path with the shortest expected delay is selected.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new statistical invariant called the *expected dependent delay*, which characterizes the expected delay of a contact dependent on another node. The closed-form expression of the delay on a path is given. In highly partitioned mobile networks where two consecutive hops on a path rarely occur at the same time, the path delay can be approximated by the sum of the expected delay of the first hop and the expected dependent delay of the remaining hops. The expected dependent delay metric can be calculated easily. Therefore, the proposed routing algorithm is applicable to a real network. Simulation results show that our proposed algorithm performs better than other algorithms that consider only either the delivery probability or the expected delay and results in a much smaller average message delay.

As future work, we plan to integrate our expected dependent delay estimation with other routing mechanisms, such as multi-copy routing where a message can be forwarded to multiple neighbors, and utility-based routing where forwarding is based on hop-by-hop decisions. Different performance metrics of the proposed routing strategy, e.g., message delivery reliability, will be examined. Furthermore, we will verify the dependency among contacts using some real movement traces and evaluate our path delay estimation in actual sparse mobile ad hoc networks.

8. REFERENCES

[1] Delay tolerant networking research group. *http://www.dtnrg.org*.

[2] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 1, pp. 24-37, 2006.

[3] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *Proc. of ACM SIGCOMM*, pp. 145-158, Aug.-Sept., 2004, Portland, OR, USA.

[4] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19-20, 2003.

[5] M. Musolesi, S. Hailes, and C. Mascolo, "Adaptive routing for intermittently connected mobile ad hoc networks," in *Proc. of IEEE Intl. Symp. on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*, vol. 1, pp. 183-189, June 2005, Taormina, Italy.

[6] J. Ghosh, S.J. Philip, and C. Qiao, "Sociological orbit aware location approximation and routing in MANET," *A*d Hoc Networks, vol. 5, no. 2, pp. 189-209, 2007.

[7] J. Leguay, T. Friedman, and V. Conan, "DTN routing in a mobility pattern space," in *Proc. of ACM SIGCOMM Workshop on Delay Tolerant Networking (WDTN)*, pp. 276-283, August 2005, Philadelphia, PA, USA.

[8] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proc. of ACM SIGCOMM*, pp. 27-34, August 2003, Karlsruhe, Germany.

[9] A. Chaintreau et al., "Impact of human mobility on the design of opportunistic forwarding algorithms," in *Proc. of IEEE INFOCOM*, April, 2006, Barcelona, Spain.

[10] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: routing for vehicle-based disruption-tolerant networks," in *Proc. of IEEE INFOCOM*, April, 2006, Barcelona, Spain.

[11] K. Tan, Q. Zhang, and W. Zhu, "Shortest path routing in partially connected ad hoc networks," in *Proc. of IEEE GLOBECOM*, vol. 2, pp. 1038-1042, December, 2003, San Francisco, CA, USA.

[12] B. Burns, O. Brock, and B.N. Levine, "*MV* routing and capacity building in disruption tolerant networks," in *Proc. of IEEE INFOCOM*, vol. 1, pp. 398-408, March, 2005, Miami, FL, USA.

[13] E.P.C. Jones, L. Li, and P.A.S. Ward, "Practical routing in delay-tolerant networks," in *Proc. of ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN)*, pp. 237-243, August 2005, Philadelphia, PA, USA.

[14] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *Proc. of ACM MobiCom*, pp. 187-201, September 2004, Philadelphia, PA, USA.

[15] M. McNett and G.M. Voelker, "Access and mobility of wireless PDA users," *ACM Mobile Computing and Communications Review*, vol. 9, no.2, pp. 40-55, April 2005.

[16] V. Srinivasan, M. Motani, and W.T. Ooi, "Analysis and implications of student contact patterns derived from campus schedules," in *Proc. of ACM MobiCom*, pp. 86-97, September 2006, Los Angeles, LA, USA.

[17] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Proc. of the 2nd ACM/SIGMOBILE International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality (REALMAN)*, May, 2006, Florence, Italy.

[18] M. Kim, D. Kotz, and S. Kim, "Extracting a mobility model from real user traces," in *Proc. of INFOCOM*, April 2006, Barcelona, Spain.

[19] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Performance analysis of mobility-assisted routing," in *Proc. of ACM/IEEE Mobihoc*, pp. 49-60, May 2006, Florence, Italy.

[20] T Spyropoulos, K. Psounis, and C.S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," in *Proc. of IEEE Conf. on Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 235-244, October 2004, Reston, VA, USA.