

An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks

Cong Liu and Jie Wu
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
{cliu8@, jie@cse}.fau.edu

ABSTRACT

Due to uncertainty in nodal mobility, DTN routing usually employs multi-copy forwarding schemes. To avoid the cost associated with flooding, much effort has been focused on probabilistic forwarding, which aims to reduce the cost of forwarding while retaining a high performance rate by forwarding messages only to nodes that have high delivery probabilities. This paper aims to provide an optimal forwarding protocol which maximizes the expected delivery rate while satisfying a certain constant on the number of forwardings per message. In our proposed optimal probabilistic forwarding (OPF) protocol, we use an optimal probabilistic forwarding metric derived by modeling each forwarding as an optimal stopping rule problem. We also present several extensions to allow OPF to use only partial routing information and work with other probabilistic forwarding schemes such as ticket-based forwarding. We implement OPF and several other protocols and perform trace-driven simulations. Simulation results show that the delivery rate of OPF is only 5% lower than epidemic, and 20% greater than the state-of-the-art delegation forwarding while generating 5% more copies and 5% longer delay.

Categories and Subject Descriptors

C.2.2 [Computer Systems Organization]: COMPUTER-COMMUNICATION NETWORKS—*Store and forward networks*

General Terms

Algorithms, Measurement, Performance

Keywords

Delay Tolerant Networks, Optimal Stopping Rule, Routing.

1. INTRODUCTION

A delay tolerant network (DTN) [1] is a sparse mobile network where a contemporary source-destination path may not

exist between a pair of source-destination nodes and messages are routed in a store-carry-forward routing paradigm. Due to uncertainty in nodal mobility, DTN routing algorithms usually spawn and keep multiple copies of the same message in different nodes. The message is delivered if one of these nodes encounters the destination.

The most expensive routing protocol, epidemic [2], forwards copies of a message to any possible node and guarantees a maximized delivery rate. Effectively flooding the network with every message, epidemic is impractical because of poor scalability in large networks. Recently, much effort has been focused on probabilistic forwarding (or opportunistic forwarding), which tries to reduce the number of copies of each message while retaining a high routing performance, i.e., a high delivery rate¹. Since only a small fraction of the nodes can get copies of a message, it is desired that these copies are forwarded by the nodes which have a higher delivery probability than the other nodes.

In this paper, we propose *optimal probabilistic forwarding* (OPF) that integrates an optimal delivery probability metric and an optimal forwarding rule. By optimality, we mean that OPF maximizes the delivery probability based on a particular knowledge about the network. The optimality of OPF relies on the assumptions that (1) nodal mobility exhibits long-term regularity such that mean inter-meeting times between nodes can be estimated from history, and ideally (2) each node knows the mean inter-meeting times of all pairs of nodes in the network. Our optimal delivery probability metric differs from the existing delivery probability metrics in two important ways.

- It is a *comprehensive* metric which reflexes not only the direct delivery probability of a message copy on a node but also its indirect delivery probability when the node can forward the message to other intermediate nodes.
- It is also a *dynamic* metric. In a hop-count-limited forwarding scheme, our optimal delivery probability metric is a function of two important states of a message copy: remaining hop-count and residual lifetime.

Our objective in OPF is that, given a certain constraint on the maximum number of forwardings per message, OPF maximizes the delivery rate of each message. The basic idea is to model each forwarding as an optimal stopping rule problem. In this forwarding model, a forwarding time is

¹In DTNs, we do not consider delay to be an important performance metric as long as the messages are delivered before they expire.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiHoc '09, May 18–21, 2009, New Orleans, Louisiana, USA.
Copyright 2009 ACM 978-1-60558-531-4/09/05 ...\$5.00.

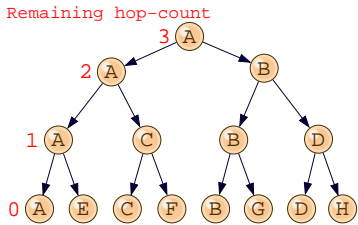


Figure 1: An illustration of hop-count-limited probabilistic forwarding

deliberately chosen for each forwarding in order to maximize the joint expected delivery probability of the copies in the sender and receiver nodes at the time of the forwarding.

Based on the proposed OPF protocol developed for a hop-count-limited forwarding scheme, we propose two non-trivial extensions for the ticket-based forwarding scheme and the broadcast forwarding scheme. In the broadcast forwarding scheme, we consider the situation that multiple receivers exist in a single forwarding and tickets are redistributed among all the new copies. We also extend OPF to work under partial routing information and propose several promising extensions for future work.

We perform simulations using the National Singapore University (NUS) student trace [3] and the UMassDieselNet [4] trace to evaluate the routing performance of OPF against several DTN routing protocols, including epidemic [2], spray-and-wait [5], and a state-of-the-art delegation forwarding [6], in terms of delivery rate, cost, and delay. Simulation results show that, in terms of delivery rate, OPF is only 5% worse than epidemic and 20% better than delegation forwarding [6]. On the other hand, OPF generates only 5% more forwardings and 5% larger delay than delegation.

This paper is organized as follows. Section 2 introduces preliminaries on probabilistic forwarding, presents an overview of OPF, displays the compared protocols in our simulation, and reviews the optimal stopping rule problem. Section 3 proposes our probability delivery metric and optimal forwarding rule in OPF. Section 4 discusses several useful extensions of OPF. Section 5 summarizes the related works. Section 6 presents our simulation methods and results. Finally, the paper is concluded in Section 7.

2. PRELIMINARIES AND OVERVIEW

2.1 Hop-count-limited forwarding

We will first propose OPF based on a hop-count-limited probabilistic forwarding scheme. In a hop-count-limited probabilistic forwarding, each message maintains a value, called *remaining hop-count*, which indicates the maximum amount of hops that the message can be forwarded. When a message whose remaining hop-count K is forwarded from one node to another, the remaining hop-count of both copies in the two nodes becomes $K - 1$. That is, if the initial hop-count of a message is a constant H , then the maximum number of copies of the message is 2^H , including the one delivered to the destination. In Figure 1, a message is created with $H = 3$ in node A , and a tree of message forwarding history is shown.

An advantage of the hop-count-limited probabilistic forwarding scheme over many other probabilistic forwarding

schemes is that, given a constant H , it has a constant per message cost², which is necessary to achieve ultimate scalability: given a constant per node message rate, the per node forwarding overhead is kept constant as the network size increases.

A ticket-based forwarding scheme is essentially a more general version of a hop-count-limited forwarding. An example is the spray-and-wait [5] protocol, in which each message is associated with a number of logical tickets, and when forwarding copies, the tickets are redistributed between the two copies. Assuming a random nodal mobility in the network, the authors wisely adopted the half-half ticket splitting strategy. We will extend OPF in a ticket-based forwarding scheme in Section 4.

2.2 Motivation and overview

In most probabilistic forwarding protocols, each node is associated with a forwarding quality/probability metric for each destination, which is usually a direct (1-hop) forwarding quality between the node and the destination, such as encounter frequency [7], or time elapsed since last encounter [8, 9, 10, 11, 12]. When node i meets node j , whether node i forwards a message to node j depends on whether the direct forwarding quality of i is better than j . We found two drawbacks in such forwarding strategies.

The first drawback is that a forwarding decision based on comparing the direct forwarding qualities of nodes i and j cannot guarantee a good forwarding. (1) The forwarding quality of j being better than i does not necessarily mean that j is a good forwarder. (2) Even though the quality of j is high, i might encounter better nodes in the near future. (3) Similarly, even though the quality of j is lower than i , j might still be the best forwarder that i could encounter in the future. To rectify this drawback, we use a comprehensive metric which reflexes not only the direct delivery probability of a particular message copy but also its indirect delivery probability when the node can forward the message to other intermediate nodes.

The second drawback is that the quality of a node is a constant regardless of two important states of the copy: remaining hop-count and residual lifetime. Remaining hop-count is an important factor: a node can be a bad 1-hop forwarder for having a low direct delivery probability, but it can still be an excellent 2-hop forwarder if it has a frequent contacting node which has a high direct delivery probability. On the other hand, residual lifetime is important because it affects a node's direct delivery probability and also the message's chance of being forwarded to other high quality nodes.

In this paper, we define a delivery probability P_{i,d,K,T_r} for each copy in i and for destination d . This metric is comprehensive because it represents the joint probability of all descendant copies, and it is also dynamic since it is a function of the remaining hop-count K and residual time-to-live T_r . With P_{i,d,K,T_r} , our optimal forwarding rule is presented as follows. We logically regard a forwarding from a node to another node as replacing a message copy with two new copies in the sender and the receiver nodes respectively. When node i meets node j , whether i should forward the copy to j depends on whether replacing the copy in i

²We consider forwarding cost as the major cost in the whole communication process. We ignore the computation and storage costs for today's rapid development in low-voltage, high-speed computers and large-size flash memories.

with two logically new copies increases the overall delivery probability. Specifically, the copy is forwarded only if the joint probability of $P_{i,d,K-1,T_r-1}$ and $P_{j,d,K-1,T_r-1}$ (in case of forwarding) is greater than the probability P_{i,d,K,T_r-1} (in case of no forwarding). Details on our optimal forwarding rule will be discussed in Section 3.

It is challenging to calculate the accurate delivery probability P_{i,d,K,T_r} for each K and T_r (Section 3.5). In this paper, we first assume that all nodes have full routing information, which is the mean inter-meeting times between all pairs of nodes. Our method is to model the calculation of P_{i,d,K,T_r} as an optimal stopping rule problem and apply a backward induction method. We will release the assumption from full routing information to partial routing information.

2.3 Protocols in comparison

We compare OPF against several other probabilistic forwarding protocols in our simulation. While OPF has a well-defined utility to maximize in each forwarding: the overall delivery probability of the copies of the same message, the following algorithms use either heuristic forwarding rules or blind forwarding.

Epidemic [2]. A node copies a message to every node it encounters that does not have a copy already, until its copy of the message times out.

Spray-and-wait [5]. This protocol differs from epidemic in that it controls the number of copies of each message in the network. A number L of logical *tickets* are associated with each message. A node i can only copy a message to another node j it encounters if the message in i owns $L > 1$ tickets or j is the destination. The new copy in j will have $L_j = \lfloor L/2 \rfloor$ tickets, and $L_i = L - L_j$ tickets will remain with the message in i .

Quality. This is an extension of epidemic where a message copy is only forwarded from node i to node j if node j has a higher forwarding quality than node i . In our simulation, we use a mean inter-meeting time $I_{k,d}$ with the destination d as the forwarding quality of a node k . That is, node j has a higher forwarding quality than node i if $I_{j,d} < I_{i,d}$.

Delegation [6]. In delegation forwarding, each message copy maintains a forwarding threshold τ which is initialized as the quality of its source node, i.e. the mean inter-meeting time between the source and the destination. Whenever node i meets node j , node i forwards a message to node j if the forwarding quality of node j exceeds the message's threshold τ , i.e. $I_{j,d} < \tau$, and then the τ s of both copies in i and j are set to $I_{j,d}$. In the case that $I_{j,d} < \tau$ but j already has the message copy, the copy is not forwarded, but the τ of the copy in i will still be set to $I_{j,d}$.

Epidemic and spray-and-wait do not use any forwarding metric. The performance of spray-and-wait degrades the fastest as network size increases. In terms of cost, spray-and-wait and OPF maintain a constant cost per message which achieves ultimate scalability. Quality has an $O(N)$ worst-case cost and delegation has an $O(\sqrt{N})$ average-case cost. Cost being proportional to network size N may result in degraded performance in small networks and excessive cost in large networks.

2.4 Optimal stopping rule problem

In a stopping rule problem [13], you may observe a sequence X_1, X_2, \dots for as long as you wish, where X_1, X_2, \dots are random variables whose joint distribution is assumed

to be known. For each stage $t = 1, 2, \dots$ after observing X_1, X_2, \dots, X_t , you may stop and receive the known reward y_t , or you may continue and observe X_{t+1} . The optimal stopping rule is to stop at some stage t to maximize the expected reward.

A stopping rule problem has a finite horizon if there is a known upper bound T on the number of stages at which one may stop. If stopping is required after observing X_1, \dots, X_T , we say the problem has horizon T . In principle, such problems may be solved by the method of *backward induction*. Since we must stop at stage T , we first find the optimal rule at stage $T - 1$. Then, knowing the optimal reward at stage $T - 1$, we find the optimal rule at stage $T - 2$, and so on back to the initial stage (stage 0). Let $V_t^{(T)}$ ($1 \leq t \leq T$) represent the maximum expected reward one can obtain starting from stage t . We define $V_T^{(T)} = y_T$ and then inductively for $t = T - 1$, backward to $t = 0$,

$$V_t^{(T)} = \max \left\{ y_t, E(V_{t+1}^{(T)}) \right\}.$$

The meaning of the above equation is that, at stage t , we compare the reward for stopping, namely y_t , with the reward $E(V_{t+1}^{(T)})$ that we expect to be able to get by continuing and using the optimal rule for stages $t + 1$ through T . The optimal reward is therefore the maximum of these two quantities, and it is optimal to stop at the earliest t when $y_t \geq E(V_{t+1}^{(T)})$.

Here, we use a house-selling scenario as a simple example for the finite horizontal optimal stopping rule problem. Suppose you have a house to sell within T days. An offer comes in each day and X_t denotes the amount of the offer received on day t . X_1, \dots, X_T are independent and identically-distributed (i.i.d.), and are uniform over 0 to M . You may stop at any day t and receive $y_t = X_t$. You don't know the offers before they come in and you cannot recall a past offer. You need to find a stopping rule that maximizes the expected sales value.

Let us derive the optimal stopping rule using the backward induction method. Since we must sell the house by day T , the expected value $E(V_T^{(T)}) = E(y_T) = E(X_T) = \frac{M}{2}$. Inductively, at day t , $E(V_t^{(T)})$

$$\begin{aligned} &= E(\max \{ y_t, E(V_{t+1}^{(T)}) \}) = \int_0^M \max \{ x, E(V_{t+1}^{(T)}) \} dF(x) \\ &= \int_{E(V_{t+1}^{(T)})}^M x d\frac{x}{M} + \int_0^{E(V_{t+1}^{(T)})} E(V_{t+1}^{(T)}) d\frac{x}{M} = \frac{M^2 + (E(V_{t+1}^{(T)}))^2}{2M}, \end{aligned}$$

where $F(x) = \frac{x}{M}$ is the cumulative distribution function of y_t , a uniform distribution over 0 to M . We can calculate $E(V_t^{(T)})$ inductively for $t = T - 1$ down to 1. The optimal stopping rule is to sell the house on day t if $X_t \geq E(V_{t+1}^{(T)})$. In other words, the optimal stopping rule uses the expected reward in stage $t + 1$ as the decision threshold for stage t .

3. OPTIMAL PROBABILISTIC FORWARDING (OPF)

3.1 Assumption

Each message has a random source and destination and is given a time-to-live at its creation time. Expired copies of messages will be deleted immediately. Different copies of

the same message are forwarded independently without any knowledge of the status of the other copies.

Like other probabilistic forwarding protocols, it is ideal that nodal mobility exhibits long-term regularities such that some nodes consistently meet more frequently than others over time. Specifically, the distribution of the mean inter-meeting times between nodes is slope, and the mean inter-meeting time between two nodes in the past will be close to that in the future with high probability. Therefore, OPF is expected to be efficient in most natural or human-related mobile networks where nodes have large clustering coefficients (nodes have preferred contacts which together form communities) and small degrees of separation (paths between nodes can have small hop-counts) [14].

First, we assume that each node knows the full routing information of the mean inter-meeting times $I_{i,j}$ between all pairs of nodes $\{i, j\}$. This can be accomplished by dissemination or via global periodical updates [15], when this routing information is long-term stable routing information that requires no timely update. In this case, the amortized overhead can be arbitrarily small depending on the update frequency. The assumption on full routing information will be relaxed in Section 4. Simulation results in Section 6 show that the performance of OPF degrades gracefully with partial information.

3.2 Discrete residual time-to-live

To model our optimal forwarding problem as an optimal stopping rule problem, we need to use a discrete residual time-to-live T_r for each message copy, with time-slot size U . T_r is a measurement in clock time. Let T_{max} be the maximum possible time-to-live of any message, the range of T_r is between 0 and T_{max}/U . Our delivery probability metric is a function of T_r , and it is calculated using an inductive method. The amount of computation for our delivery probability metric is inversely proportional to the length of U , but its accuracy decreases as U increases. In the rest of the paper, we use T_r to denote a residual lifetime or a particular time-slot at T_r interchangeably without causing confusion.

In each time-slot T_r , a node can either meet or not meet another node. A node has the probability to meet several other nodes during the same time-slot, and we simply assume that all meetings start at the beginning of some time-slot. This assumption holds when U is smaller than any meeting duration, and we truncate all meeting durations so that the starting time of them are aligned in the beginning of their respective time-slots. The meeting probability of two nodes in any time-slot of length U is estimated under the assumption of exponential inter-meeting time [5, 10] by

$$M_{i,j} = 1 - \exp\left(-\frac{U}{I_{i,j}}\right).$$

Note that the calculation of $M_{i,j}$ itself does not rely on the assumption of exponential inter-meeting times. Using a particular estimation that is more realistic for a network in question should result in better routing performance.

3.3 1-hop delivery probability

The 1-hop delivery probability of a message copy is the probability that the hosting node meets the destination directly within its time-to-live. It is only a function of residual time-to-live. We estimate the 1-hop delivery probability, as-

Table 1: Forwarding options.

T_r	P_{i,d,K,T_r}	
$T_r - 1$	Not Forward	Forward (becomes $K - 1$)
	P_{i,d,K,T_r-1}	$P_{i,d,K-1,T_r-1}, P_{j,d,K-1,T_r-1}$

suming again an exponential inter-meeting time, by

$$P_{i,d,0,T_r} = 1 - \exp\left(-\frac{T_r \times U}{I_{i,d}}\right),$$

where 0 means that the remaining hop-count is 0, $T_r \times U$ is the residual time-to-live of the message (T_r is the number of residual time-slots of length U before the message expires), and $I_{i,d}$ is the mean inter-meeting time between node i and the destination d . Again, the calculation of $P_{i,d,0,T_r}$ does not rely on the assumption of exponential inter-meeting time.

3.4 K -hop delivery probability and forwarding rule

Our optimal delivery probability and optimal forwarding rule are inter-dependent. The optimal delivery probability of a copy in node i , heading for destination d , with a remaining hop-count K ($K > 0$), and with a residual time-to-live T_r , is denoted by P_{i,d,K,T_r} .

First, we present our optimal forwarding rule. When a copy, whose remaining hop-count is K , is in node i and node i meets node j at time-slot T_r , the decision on whether to forward depends on whether replacing the copy in i with two new copies in i and j respectively will increase the overall delivery probability. As shown in Table 1, if the message is not forwarded in time-slot T_r , then in the next time-slot, we have the same copy (with the same remaining hop-count K) in i and its delivery probability becomes P_{i,d,K,T_r-1} . On the other hand, if the message is forwarded in time-slot T_r , then in the next time-slot, we have two new copies with remaining hop-count $K - 1$ in i and j respectively whose delivery probabilities are $P_{i,d,K-1,T_r-1}$ and $P_{j,d,K-1,T_r-1}$ respectively. To maximize the delivery probability, the optimal forwarding rule is to forward the message if

$$1 - (1 - P_{i,d,K-1,T_r-1}) \times (1 - P_{j,d,K-1,T_r-1}) \geq P_{i,d,K,T_r-1}.$$

For simplicity, in the above discussion, we assumed that in a sparse DTN, two consecutive forwardings (i.e. from i to j and then from j to another node) cannot happen in the same time-slot. Also, we only considered uni-cast forwarding. When connected with several nodes at the same time-slot, we forward the copy to the node j which has the largest $P_{j,d,K-1,T_r-1}$. Since wireless communication is broadcast in nature, a single forwarding can create copies in several nodes. A more general forwarding scheme, broadcast forwarding, will be discussed as an extension in the next section.

Our optimal delivery probability $P_{i,d,K-1,T_r}$ depends on the optimal forwarding rule and the meeting probabilities $M_{i,j}$ of i and each node j in time-slot T_r whose delivery probability $P_{j,d,K-1,T_r-1}$ satisfies the forwarding criteria. We will calculate P_{i,d,K,T_r} in the next subsection by modeling forwarding as an optimal stopping rule problem.

3.5 OPF as an optimal stopping rule problem

We can model each forwarding as an optimal stopping problem as follows. A node i has a message copy with re-

maining hop-count K , which can be forwarded once. At the time of forwarding, the copy is logically regarded as being replaced by two new copies, both of which have a $K - 1$ remaining hop-count. Candidate copy receivers j come in at each time-slot T_r (which also denotes the residual time-to-live of the message) with probability $M_{i,j}$. Upon meeting with j , i can either forward the copy to j or not. Since we assume no consecutive forwardings occur in the same time-slot, we calculate the resulting overall delivery probability in the next time-slot $T_r - 1$. If the copy is forwarded, the delivery probability of the copy in node j will be $P_{j,d,K-1,T_r-1}$, and that of the copy in node i will become $P_{i,d,K-1,T_r-1}$. On the other hand, if the copy is not forwarded to any node at time-slot T_r , the delivery probability of the same copy in node i will become P_{i,d,K,T_r-1} .

In the case that a node meeting several other nodes in the same time-slot, forwarding the copy to the node with the highest delivery probability is the optimal strategy to maximize delivery probability. Given the sorted delivery probability $P_{j,d,K-1,T_r-1}, P_{k,d,K-1,T_r-1}, \dots$ of the nodes j, k, \dots that i will probably meet in time-slot T_r with meeting probability $M_{i,j}, M_{i,k}, \dots$ respectively, the maximum probability that the copy will be forwarded to one of nodes j, k, \dots in time-slot T_r and then be delivered is

$$P(\text{delivered} | \text{forwarded at } T_r) \times P(\text{forwarded at } T_r) = P(\text{delivered} \cap \text{forwarded at } T_r) = M_{i,j} \times P_{i,d,K-1,T_r-1} + (1 - M_{i,j}) \times M_{i,k} \times P_{k,d,K-1,T_r-1} + \dots$$

Therefore, the expected optimal delivery probability P_{i,d,K,T_r} equals the sum of (1) the probability that the copy will be forwarded in time-slot T_r and then be delivered, and (2) the delivery probability P_{i,d,K,T_r-1} when the message is not forwarded in time-slot T_r multiple by the probability $M'_{i,N}$ that node i does not meet any node in T_r that satisfies the forwarding criteria, where $M'_{i,N} = 1 - M_{i,j} - (1 - M_{i,j}) \times M_{i,k} - \dots$. Algorithm 1 shows the calculation of P_{i,d,K,T_r} using the backward induction method. In line 7, the while loop stop when queue Q is empty.

Algorithm 1 Calculation of P_{i,d,K,T_r}

```

1:  $P_{i,d,K,T_r} := 0$ 
2:  $M'_{i,N} := 1$ 
3: for each (node  $j, j \neq i \cap j \neq d$ ) {
4:    $P_{i,j} = 1 - (1 - P_{i,d,K-1,T_r-1}) \times (1 - P_{j,d,K-1,T_r-1})$ 
5: }
6:  $Q :=$  a priority queue of  $j$  in decreasing order of  $P_{i,j}$ 
7: while ( $j := \text{dequeue}(Q)$  and  $P_{i,j} > P_{i,d,K,T_r-1}$ ) {
8:    $P_{i,d,K,T_r} := P_{i,d,K,T_r} + M'_{i,N} \times M_{i,j} \times P_{i,j}$ 
9:    $M'_{i,N} := M'_{i,N} - M'_{i,N} \times M_{i,j}$ 
10: }
11:  $P_{i,d,K,T_r} := P_{i,d,K,T_r} + M'_{i,N} \times P_{i,d,K,T_r-1}$ 

```

4. EXTENSIONS

4.1 Routing with partial information

To calculate P_{i,d,K,T_r} for any K and T_r , each node collects the mean inter-meeting times of every pair of nodes. When the nodal mobility in the network exhibits long-term regularities, the mean inter-meeting times are a long-term stable routing information which can be infrequently updated and therefore generate a low amortized overhead. In practice, they can be generated from historical connectivity informa-

tion (as in the UMassDieselNet trace [9, 4]) or from prior knowledge on the contact pattern of the nodes (as in the NUS student contact trace [3]).

The mean inter-meeting times can also be incrementally exchanged among the nodes. When node i meets node j , node i sends to node j its mean inter-meeting time with the other nodes (1-hop routing information), or it can also send the mean inter-meeting times received from the other nodes (k -hop routing information). Alternatively, node i can send only some preferred information to j , e.g., only the mean inter-meeting times between frequent meeting nodes are sent.

OPF can be simply extended to work under partial information (including k -hop and preferred mean inter-meeting times), i.e., when the mean inter-meeting times between all pairs of nodes are not available to every node. To allow OPF to work with partial information, for each pair of nodes i and j whose mean inter-meeting time is unknown, we simply set their time-slot based meeting probability $M_{i,j}$ and their 1-hop meeting probabilities $P_{i,j,0,T_r}$ (for all T_r) to 0s. When no routing information is available, it is easy to see that OPF degrades to spray-and-wait which spawns copies to the first node seen.

4.2 Ticket-based forwarding

We have proposed an optimal probabilistic forwarding based on the hop-count-limited forwarding. Using similar techniques, we can design a slightly more complicated ticket-based optimal probabilistic forwarding. In this ticket-based scheme, each message copy is associated with a number of L logical tickets [5] which will be redistributed between the two replacing copies in a message forwarding. L upper-bounds the total number of forwardings of each message. For ticket-based optimal forwarding, we define a delivery probability P_{i,d,L,T_r} , which differs from the hop-count-limited version by replacing K with L .

When a node i meets node j and is deciding whether to forward a copy with $L > 1$ tickets, it first lists all possible ticket redistribution situations. Lets say that after a forwarding, the copy in i has $L_i > 0$ tickets and the copy in j has $L_j > 0$ tickets. It requires that $L_i + L_j = L$. Also, L_i and L_j must be selected so as to get the maximum joint delivery probability of the two new copies, $P_{i,j}^{max} = \max 1 - (1 - P_{i,d,L_i,T_r-1}) \times (1 - P_{i,d,L_j,T_r-1})$. In the event that $P_{i,j}^{max} \leq P_{i,d,L,T_r-1}$, the message will not be forwarded.

Algorithm 2 Calculation of P_{i,d,L,T_r}

```

1:  $P_{i,d,L,T_r} := 0$ 
2:  $M'_{i,N} := 1$ 
3: for each (node  $j, j \neq i \cap j \neq d$ ) {
4:   calculate  $P_{i,j}^{max}$  using all  $L_i$  and  $L_j$ 
5: }
6:  $Q :=$  a priority queue of  $j$  in decreasing order of  $P_{i,j}^{max}$ 
7: while ( $j := \text{dequeue}(Q)$  and  $P_{i,j}^{max} > P_{i,d,L,T_r-1}$ ) {
8:    $P_{i,d,L,T_r} := P_{i,d,L,T_r} + M'_{i,N} \times M_{i,j} \times P_{i,j}^{max}$ 
9:    $M'_{i,N} := M'_{i,N} - M'_{i,N} \times M_{i,j}$ 
10: }
11:  $P_{i,d,L,T_r} := P_{i,d,L,T_r} + M'_{i,N} \times P_{i,d,L,T_r-1}$ 

```

With this optimal forwarding rule, we apply backward induction to calculate each P_{i,d,L,T_r} as listed in Algorithm 2. The direct delivery probability $P_{i,d,0,T_r}$ is identical to that in the hop-count-limited scheme. The ticket-based optimal

forwarding is expected to have a better routing performance than the hop-count-limited counterpart since the later can be regarded as a special case of the former. In this paper, we did not implement this ticket-based OPF.

4.3 Broadcast forwarding

Before this point, we assumed uni-cast forwarding, in which only one copy is created as a result of one forwarding. Now, we consider a more general ticket-based scheme, broadcast forwarding, in which a single forwarding can spawn copies in multiple nodes and redistributes tickets among those copies.

Suppose that in time-slot T_r , node i has a copy with L tickets and it meets several other nodes $S = \{j, k, \dots\}$. Node i then finds the best combination of ticket redistribution, say $\{L_i, L_j > 0, L_k > 0, \dots\}$, which satisfies $L_i + L_j + L_k + \dots = L$ and that the maximum joint probability $P_{i,S}^{max}$ of all copies is achieved when forwarded. In the event that $P_{i,S}^{max} \leq P_{i,d,L,T_r-1}$, the message will not be forwarded.

We denote the ticket-based optimal delivery probability for broadcast forwarding as P_{i,d,L,T_r}^m , and calculate it using backward induction as listed in Algorithm 3. In this algorithm, we let $P_{i,d,0,T_r}^m = P_{i,d,0,T_r}$. For a combination $S = \{j, k, \dots\}$ of nodes, we use $M_{i,S} := M_{i,j} \times M_{i,k} \times \dots$ as the probability that i meets all nodes in S simultaneously (assuming that the meeting probability of i with the other nodes are independent).

Algorithm 3 Calculation of P_{i,d,L,T_r}^m

```

1:  $P_{i,d,L,T_r}^m := 0$ 
2:  $M'_{i,N} := 1$ 
3: for each ( $S = \{j, k, \dots\} \subseteq N - \{i, d\}$ ) {
4:   calculate  $P_{i,S}^{max}$  using all combinations  $\{L_i, L_j, L_k, \dots\}$ 
5: }
6:  $Q :=$  a priority queue of  $S$  in decreasing order of  $P_{i,S}^{max}$ 
7: while ( $S := \text{dequeue}(Q)$  and  $P_{i,S}^{max} > P_{i,d,L,T_r-1}$ ) {
8:    $M_{i,S} := M_{i,j} \times M_{i,k} \times \dots$ 
9:    $P_{i,d,L,T_r}^m := P_{i,d,L,T_r}^m + M'_{i,N} \times M_{i,S} \times P_{i,S}^{max}$ 
10:   $M'_{i,N} := M'_{i,N} - M'_{i,N} \times M_{i,S}$ 
11: }
12:  $P_{i,d,L,T_r}^m := P_{i,d,L,T_r}^m + M'_{i,N} \times P_{i,d,L,T_r-1}$ 

```

4.4 Closed-form expression

We have solved the optimal forwarding problem assuming that each node has the mean inter-meeting between all pairs of nodes instead of assuming a stationary distribution function of the inter-meeting times between the nodes. Our delivery probability metric is a function of remaining hop-count K (or tickets L) and the residual time-to-live T_r of the message. In the future, we will continue our work from a more theoretical aspect in which we will derive a closed-form expression for delivery probability P_{i,d,K,T_r} as a function of K (or L) and T_r , assuming a stationary distribution function for the mean inter-meeting time is available.

4.5 Dynamic routing parameters

In our hop-count-limited or ticket-based forwarding algorithms, we assume that the initial hop-count and the initial number of tickets are constant parameters in the algorithm. In the future, we can make them variable parameters in the forwarding algorithm. For example, we can collect delivery rate in the network and the ratio between busy time over

idle time in terms of communication during meetings between nodes, and derive utility functions regarding factors such as idle time ratio, delivery rate, and initial hop-count.

5. RELATED WORKS

Delay Tolerant Network Research Group (DTNRG) [1] has designed a complete architecture to support various protocols in DTNs. In [16], Jain, Fall, and Patra presented a comprehensive investigation on the DTN routing problem with different levels of prior knowledge about the network. Specifically, Dijkstra's algorithm (with future connectivity information) or the linear programming approach (with information of future connectivity and traffic demands) is used to obtain an optimal path between a source and a destination. In [17], Merugu, Ammar, and Zegura proposed a DTN routing algorithm that is similar in spirit to Dijkstra's algorithm in [16]. In [18], Liu and Wu presented hierarchical routing in DTNs with deterministic repetitive mobility to improve scalability.

Epidemic routing [2] is the first flooding-based routing algorithm. Gossip [19] forwards with probability p . Probabilistic routings, such as [7], forward based on some delivery probability metric. Different delivery probability metrics are proposed including encounter frequency [7], time elapsed since last encounter [8, 9, 10, 11, 12], social similarity [20, 21], location similarity [22], time-varying expected delay [15], timely-contact probability [23] and geometric distance [24].

Trace data available for the research community [25] includes the UMassDieselNet trace, the NUS student contact trace, the Huggle project, and the MIT Reality Mining. [32].

6. SIMULATION

We evaluate our protocol, OPF, against other routing algorithms using the NUS student contact trace and the UMassDieselNet trace. The routing protocols implemented to compare with OPF were listed in Section 2.3. All of the protocols that we implement aim to compare different delivery probability metrics, and all other optimizations that have orthogonal effects on the performance of the protocols are not implemented. The orthogonality means that these optimizations can be added to all of our implemented algorithms and they are expected to provide an equal level of improvement in their routing performance. Such optimizations may include buffer management [9], global estimation of message delivery probability [10] and social centrality of the nodes [20], and geometric information [24].

We do not assume any acknowledgment mechanism, and therefore even though some copies of a message may have been delivered, the other copies of the same message may still be forwarded in the network.

6.1 Simulation methods and settings

NUS student contact trace. Accurate information of human contact patterns are available in scenarios such as university campuses. As shown by the National University of Singapore (NUS) student contact trace model [3], when the class schedules and student enrollment for each class on a campus are known, accurate information about contact patterns between students over large time scales can be obtained without a long-term contact data collection. The schedules of the 4,885 classes and enrollment of 22,341 stu-

Table 2: Settings for NUS student trace.

parameter name	default	range
number of students	300	1~500
attendance rate (P_{attend})	0.8	0.1~0.9
clustering factor (C)	0.4	0.1~0.9
message time-to-live (T_r)	77 hours	11~77 hours
tickets in spray-and-wait (L)	10	
initial hop-count (K)	3	
length of time-slot (U)	1 hour	
simulation time	77 hours	

dents for each of the classes for each week of 77 class hours are publicly available on [25]. Their contact model is simplified in several ways. (1) Two students are in contact with each other only if they are in the same classroom at the same time. (2) Sessions start on the hour and end on the hour, which means that hour is the unit of time for the contact duration. (3) Only the contacts that take place during the 11 class hours per day are used. Non-class hours are removed to compress time. The trace synthesized in this model exhibits the same set of characteristics to those observed in the real world.

Similar to [15], we select a number of students N ($100 \leq N \leq 500$) in each experiment due to the memory constraint in the simulation environment. Contacts related to the non-selected students are ignored. We generate non-deterministic traces by taking absentees into consideration. Each student attends a class with an attendance probability P_{attend} . A clustering factor C is needed in the selection of N students. That is because, if students are selected randomly, the network becomes too sparse for messages to be delivered. On the other hand, when students are selected by maximizing their similarities (the number of common classes they are enrolled in), the network becomes almost fully connected. To prevent the above extremes and maintain the small-world property in the size-reduced student networks, we use the following process. We select the first student randomly. To select the k^{th} student, we divide the $k - 1^{th}$ selected students into two groups S_1 and S_2 , and select the k^{th} student s as the one with the highest score $\sum_{s_1 \in S_1} sim(s, s_1) - \sum_{s_2 \in S_2} sim(s, s_2)$ among the students that are not yet selected, where the similarity function sim is defined as the number of common class sessions enrolled by two students. The clustering factor is defined as $C = |S_1| / (|S_1| + |S_2|)$ which determines the degree of connectivity in the network.

The default settings in our simulation, as shown in Table 2, are $P_{attend} = 0.8$, $C = 0.5$, 300 students, 10 tickets per message in spray-and-wait, and 3 initial hop-counts in OPF. In different simulations, we vary one of the four variable parameters as shown in the table. For each setting, 30 simulations are run. In the beginning of the simulation, every node sends 20 messages to 20 randomly selected destination nodes. The total simulation time in each experiment is one week, or 77 class hours. The initial time-to-live of all messages is 77 hours. A message is considered to be undelivered if none of its copies are sent to its destination before the end of the simulation. Since the uniform meeting interval of the nodes is one hour, we can safely use an hour as the time unit for the residual time-to-live T_r . The mean inter-meeting time between any pair of students is calculated from

Table 3: Settings for UMassDieselNet trace.

parameter name	default	range
tickets in spray-and-wait (L)	10	
initial hop-count (K)		1~5
message time-to-live (T_r)	10 hours	
length of time-slot (U)	1 minute	
simulation time	1 day	day 1~55

the number of their common classes in each week and the attendance rate.

UMassDieselNet trace. In the UMassDieselNet [4, 9] bus system consisting of 40 buses, the bus-to-bus contacts (the durations of which are relatively short) are logged. Our experiments are performed on traces collected over 55 days during the spring 2006 semester with weekends, spring break, and holidays removed due to reduced schedules. The bus system serves approximately ten routes. There are multiple shifts serving each of these routes. Shifts are further divided into morning (AM), midday (MID), afternoon (PM), and evening (EVE) sub-shifts. Drivers choose buses at random to run the AM sub-shifts. At the end of the AM sub-shift, the bus is often handed over to another driver to operate the next sub-shift on the same route or on another route. Unfortunately, the all-bus-pairs contacts provided in the original traces show no discernible contact pattern among the nodes. We performed the data process in [15] to generate the contacts at a sub-shift level which exhibit periodic behavior. This process translates 55 days of the bus-to-bus contacts into contacts between sub-shifts.

The default settings of the UMassDieselNet trace simulation are shown in Table 3. We use 10 tickets per message in spray-and-wait and 1~5 initial hop-counts in OPF. We use the 55 days of traces to run respective simulations. In each simulation, every node (sub-shift) sends a message for a random destination node every five minutes. Since most contacts in the UMassDieselNet trace are between hours 6 and 20, messages are sent only during hours 6 to 12 and we set a uniform initial time-to-live of all messages which is 10 hours. We use one minute as the time unit for residual time-to-live T_r . The mean inter-meeting time between all pairs of shifts is generated from the 55 days of sub-shift based contacts.

6.2 Results with full routing information

NUS student contact trace. The delivery rates of the forwarding algorithms are compared in Figures 2(a), 2(d), 3(a), and 3(d) with different numbers of students, attendance rate, message time-to-live, and clustering factor. The results show that OPF delivers only about 5% fewer messages than epidemic and around 20% more than delegation in our default settings. Compared with delegation, OPF has a more steady performance in different network conditions. For example, OPF delivers 50% more messages than delegation when the clustering factor is 0.5.

The delays of the message delivered by all algorithms are compared in Figures 2(b), 2(e), 3(b), and 3(e). On average, OPF has a 5% additional delay than the other forwarding algorithms. This amount of delay is the side-effect of a deliberate forwarding scheme which sends copies to the best relay nodes instead of the earliest contacting nodes.

While OPF maintains high delivery rates, its also has the

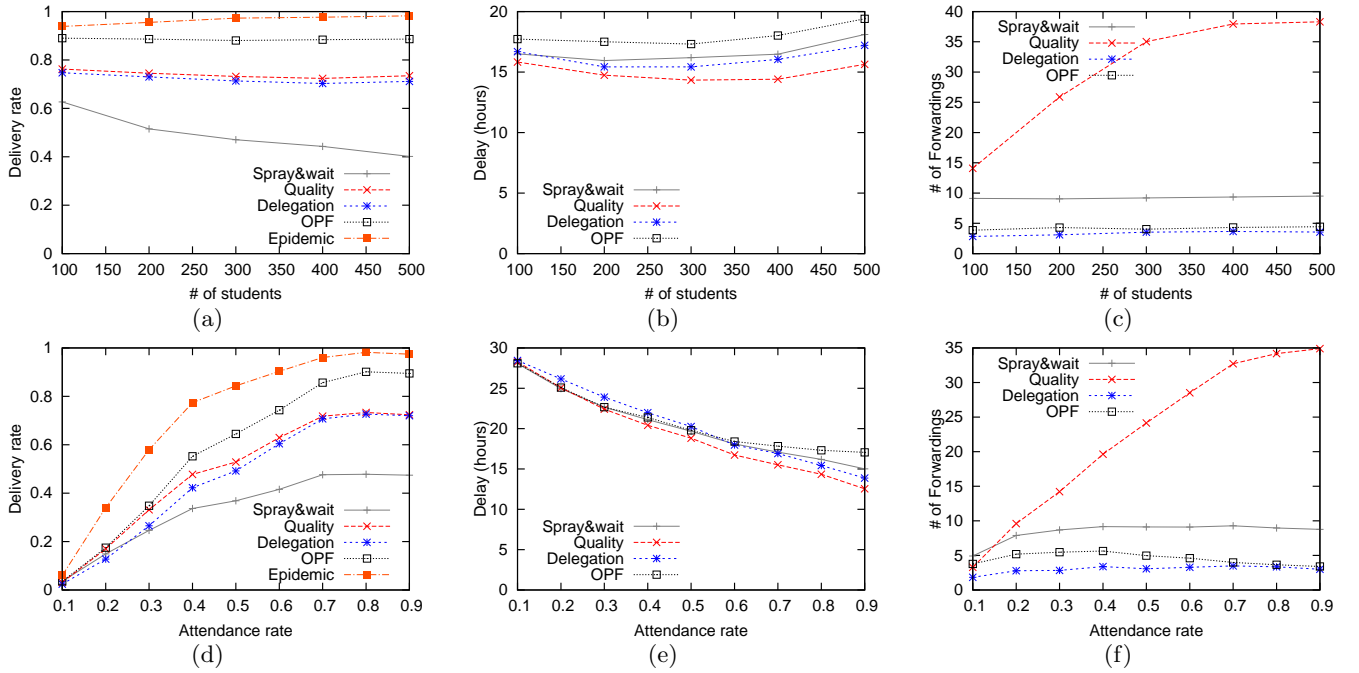


Figure 2: Delivery rate, delay, and cost verse number of students and attendance rate in NUS trace.

lowest possible cost in terms of number of forwardings, which is shown in Figures 2(c), 2(f), 3(c), and 3(f). OPF has a much lower cost than quality and spray-and-wait. Although delegation has lower costs than OPF in some situations, its corresponding delivery rates degrade much faster than OPF.

UMassDieselNet trace. The delivery rate, delay, and cost of different routing algorithms are compared in Figures 5(a), 5(b), and 5(c) respectively. All probabilistic forwarding algorithms have lower delivery rates compared with epidemic. Compared with spray-and-wait, OPF, initialized with two remaining hop-count ($H = 2$), has a higher delivery rates and a lower cost. In terms of all metrics, OPF ($H = 2$) is similar to quality and delegation in this trace.

The performances of OPF with initial hop-counts H ranging from 1 to 5 are compared in Figures 5(d), 5(e), and 5(f). The delivery rate of OPF has an almost constant increase at each increment of H . On the other hand, the cost increases much faster and is within the corresponding bounds of 2^H . Note that when $H = 5$, the delivery rate of OPF is very close to epidemic with less than a half of its cost; when $H = 4$, the delivery rate of OPF is 90% of that of epidemic with 30% of its cost.

6.3 Results with partial routing information

In this subsection, we will evaluate the routing performance of OPF with incomplete routing information by comparing it to the routing performance of OPF with full routing information in the NUS student contact trace using its default settings. The routing information used by OPF is the inter-meeting times between the nodes. With full information, the mean inter-meeting times between all pairs of nodes are known to every node in the network. We simulate partial routing information by setting the mean inter-meeting times between a certain percentage of nodes as unknown. As presented in Section 4.1, when the mean inter-meeting times

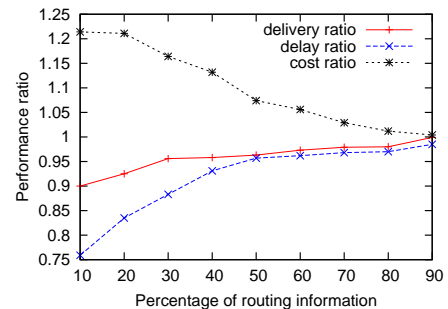


Figure 4: Performance ratio of OPF with incomplete routing information.

between a pair of nodes i and d is unknown, their meeting probability $M_{i,d} = 0$ in any time-slot U , and their direct (1-hop) meeting probability $P_{i,d,0,T_r} = 0$ for any T_r .

In our simulation, we perform different sets of simulations with different percentages of available routing information ranging from 10% to 90%, in increments of 10%. For example, when the available routing information is 10%, we remove 90% of the mean inter-meeting times from the nodes.

As shown in Figure 4, as the percentage of routing information decreases, the delivery ratio of OPF degrades very slowly and the cost increases moderately. This is because the delivery probabilities P_{i,d,K,T_r} will be under-estimated as a result of the decreasing of (1) the meeting probabilities $M_{i,j}$ of node i with other forwarders j and (2) the 1-hop delivery probabilities of all other nodes $P_{j,d,0,T_r}$. With P_{i,d,K,T_r} being under-estimated, the forwarding threshold is lowered. This could explain why the cost (number of forwardings) increases and the delivery rate decreases. The delay increases as a result of the increase in forwardings.

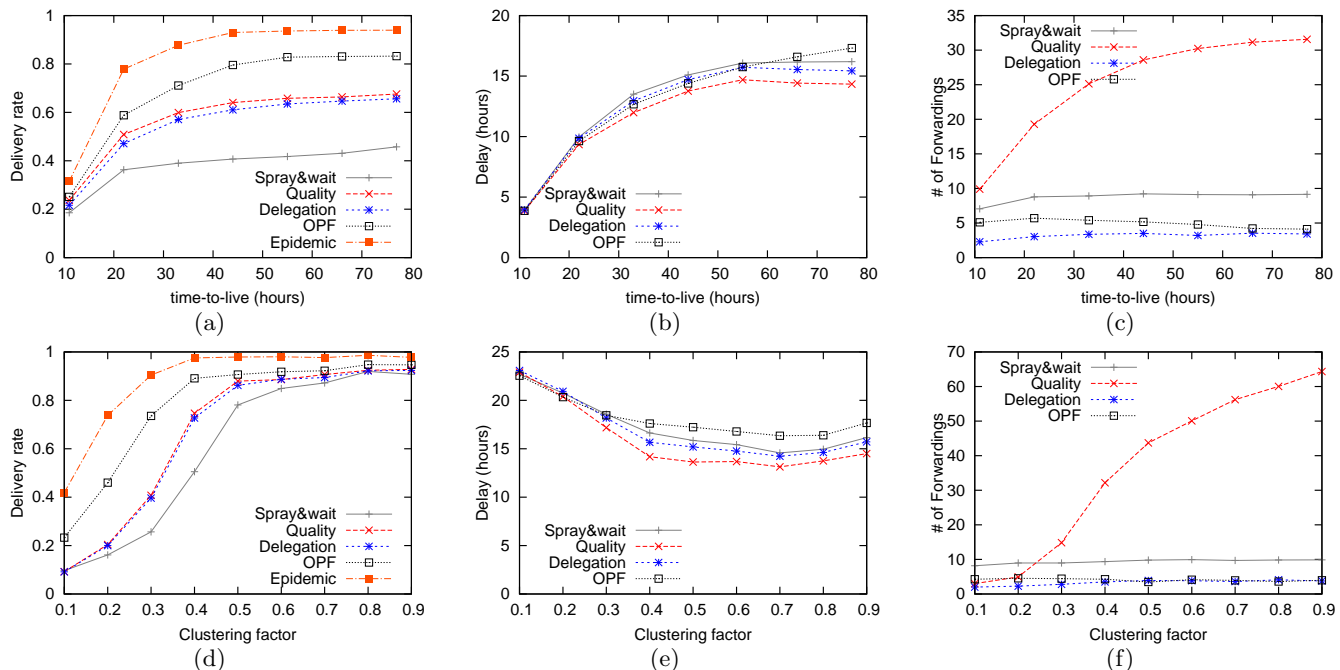


Figure 3: Delivery rate, delay, and cost verse number of time-to-live and clustering factor in NUS trace.

However, even though OPF is proposed under a full routing information assumption, the simulation results show that OPF maintains a good performance with only a small portion of routing information. As shown in Figure 4, with only 10% of routing information, OPF still delivers 90% of the message that was delivered with full routing information. The possible reason for the slow performance degradation is that the delivery probabilities are under-estimated as a whole but the ratios between the delivery probabilities are basically preserved. The fact that OPF performs well with partial routing information increases its practical value, as it does not require global information and therefore is applicable to networks with more irregular factors.

We also perform a simulation using the NUS trace where the partial routing information is simulated by restricting each node to using the mean inter-meeting times of its k -hop ($1 \leq k \leq 3$) contacting nodes. The results shown that, in our network with an average diameter of 4 hops, with 1-hop information, the delivery ratio is 93% and the cost ratio is 102%; with 2-hop information the delivery ratio is 98% and the cost ratio is 101%.

6.4 Summary of simulation

Simulation results confirmed that, compared with other algorithms, OPF has a high delivery rate and a low cost. The results under the default setting for the NUS student trace show that the delivery rate of OPF is only 5% lower than epidemic, and 20% greater than delegation forwarding while having only 5% more copies and 5% longer delay. Simulation results using partial routing information show that the routing performance of OPF degrades very slowly as routing information decreases. With only 10% of routing information, OPF delivers only 10% fewer messages and has a 10% increase in cost.

7. CONCLUSION

In this paper, we provided an optimal forwarding protocol which maximizes the expected delivery rate while satisfying the constant on the number of forwardings per message. We proposed the *optimal probabilistic forwarding* (OPF) protocol which makes optimal forwarding decisions by modeling forwarding as an optimal stopping rule problem. Specifically, OPF uses forwarding thresholds as functions of remaining hop-count and residual time-to-live. We also presented several useful extensions to allow OPF to run with partial routing information and work with other probabilistic forwarding schemes. We implemented OPF as well as several other protocols and performed trace-driven simulations. Simulation results verified the efficiency of OPF.

In the future, we will perform simulations on the extended version of OPF, i.e., ticket-based forwarding and broadcast forwarding. We will also do research on existing mobility models and find a closed-form expression for more accurate delivery probability metrics using the backward induction method used in this paper. Also, our algorithm can be more adaptive by using an adjustable initial hop-count.

8. REFERENCES

- [1] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss. Delay Tolerant Networking Architecture. In *Internet draft: draft-irrf-dtnrg-arch.txt*, DTN Research Group, 2006.
- [2] A. Vahdate and D. Becker. Epidemic Routing for Partially-connected Ad Hoc Networks. In *Technical Report*, Duke University, 2002.
- [3] V. Srinivasan, M. Motani, and W. T. Ooi. Analysis and Implications of Student Contact Patterns Derived from Campus Schedules. In *Proc. of ACM MobiCom*, 2006.

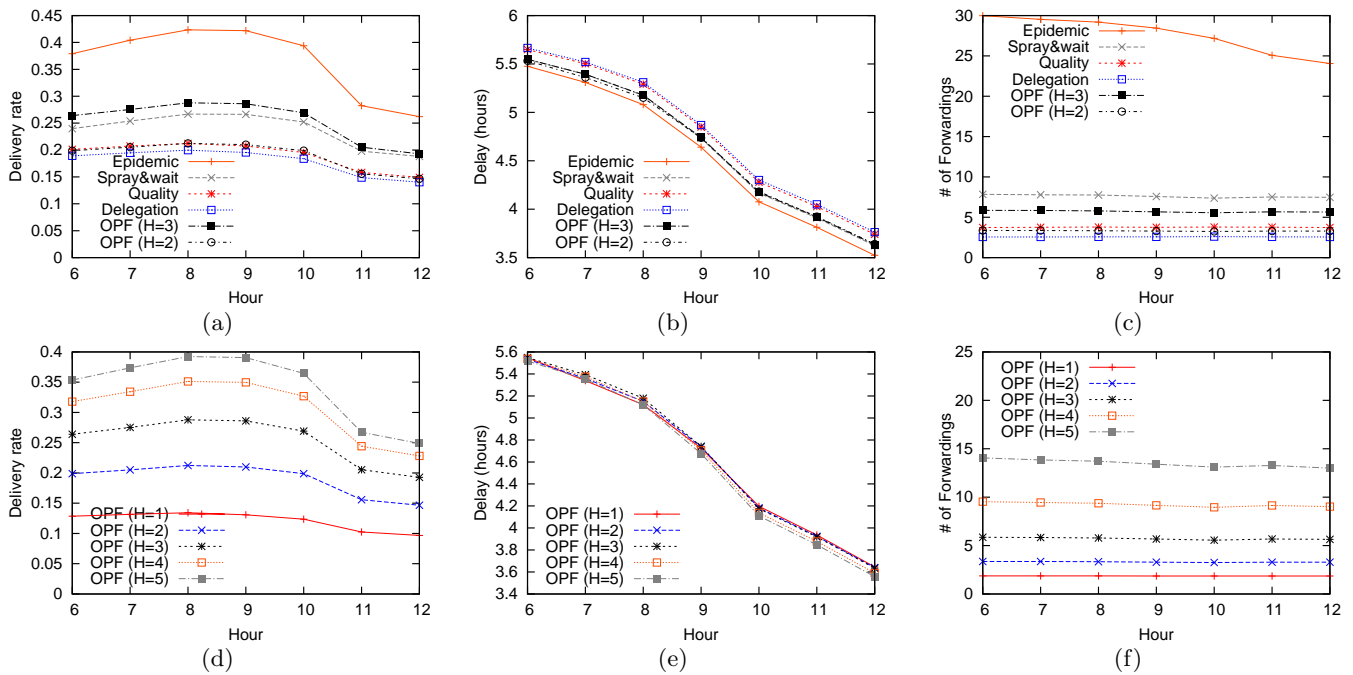


Figure 5: Comparison of routing performance in UMassDieselNet trace.

- [4] X. Zhang, J. F. Kurose, B. Levine, D. Towsley, and H. Zhang. Study of a Bus-Based Disruption Tolerant Network: Mobility Modeling and Impact on Routing. In *Proc. of ACM MobiCom*, 2007.
- [5] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks. In *Proc. of ACM WDTN*, 2005.
- [6] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot. Delegation Forwarding. In *Proc. of ACM MobiHoc*, 2008.
- [7] A. Lindgren, A. Doria, and O. Schelen. Probabilistic Routing in Intermittently Connected Networks. *Lecture Notes in Computer Science*, 3126:239–254, August 2004.
- [8] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility. In *Proc. of IEEE PerCom*, 2007.
- [9] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networking. In *Proc. of IEEE INFOCOM*, 2006.
- [10] A. Balasubramanian, B. N. Levine, and A. Venkataramani. DTN Routing as a Resource Allocation Problem. In *Proc. ACM SIGCOMM*, 2007.
- [11] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Age Matters: Efficient Route Discovery in Mobile Ad Hoc Networks Using Encounter Ages. In *Proc. of ACM MobiHoc*, 2003.
- [12] M. Grossglauser and M. Vetterli. Locating Nodes with Ease: Last Encounter Routing in Ad Hoc Networks through Mobility Diffusion. In *Proc. of IEEE INFOCOM*, 2003.
- [13] Optimal Stopping and Applications. <http://www.math.ucla.edu/~tom/Stopping/Contents.html>.
- [14] D. Watts and S. Strogatz. Collective Dynamics of Small-world Networks. *Nature* 393, 440, 1998.
- [15] C. Liu and J. Wu. Routing in a Cyclic MobiSpace. In *Proc. of ACM MobiHoc*, 2008.
- [16] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *Proc. of ACM SIGCOMM*, 2004.
- [17] S. Merugu, M. Ammar, and E. Zegura. Routing in Space and Time in Network with Predictable Mobility. In *Technical report: GIT-CC-04-07, College of Computing, Georgia Tech*, 2004.
- [18] C. Liu and J. Wu. Scalable Routing in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2007.
- [19] J. Haas, J. Y. Halpern, and L. Li. Gossip-Based Ad Hoc Routing. In *Proc. of IEEE INFOCOM*, 2002.
- [20] D. Elizabeth and H. Mads. Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs. In *Proc. of ACM MobiHoc*, 2007.
- [21] P. Hui, J. Crowcroft, and E. Yoneki. BUBBLE Rap: Social-based Forwarding in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2008.
- [22] J. Leguay, T. Friedman, and V. Conan. DTN Routing in a Mobility Pattern Space. In *Proc. of ACM WDTN*, 2005.
- [23] L. Song and D. F. Kotz. Evaluating Opportunistic Routing Protocols with Large Realistic Contact Traces. In *Proc. of ACM CHANTS*, 2007.
- [24] U. Acer, S. Kalyanaraman, and A. Abouzeid. Weak State Routing for Large Scale Dynamic Networks. In *Proc. of ACM MobiCom*, 2007.
- [25] CRAWDAD data set. Downloaded from <http://crawdad.cs.dartmouth.edu/>.