2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems

A Localized Efficient Forwarding Algorithm in Large-scale Delay Tolerant Networks

Yuxing He†, Cong Liu†, Yan Pan†, Jun Zhang†, Jie Wu‡, Yaxiong Zhao♯, Shuhui Yang⋄, and Mingming Lu§
†Sun Yat-sen University, ‡Temple University, ♯Amazon Inc., ⋄Purdue University Calumet, §Central South University

***Abstract*—This paper proposes an efficient opportunistic forwarding algorithm in *Delay Tolerant Networks* (DTNs) using only local information: inter-meeting times collected locally. It tries to minimize delay with a controlled energy consumption through placing a limitation on the number of total copies per message. The proposed forwarding algorithm makes forwarding decisions based only on local information, which means that no information is needed to be exchanged among the nodes, except for the data to be transferred. The removal of information propagation is particularly important in large-scale DTNs with limited communication opportunities like vehicular communication networks. On the contrary, most existing algorithms either forward messages randomly without facilitating any information, or require the exchange of certain information to make wise forwarding decision. Extensive real trace-driven simulations are conducted, and the proposed algorithm significantly outperforms all of the compared localized algorithms in every simulation.**

***Index Terms*—Large-scale Delay Tolerant Networks, Optimal Stopping Rule, Local Forwarding Decision, Trace-driven Simulation.**

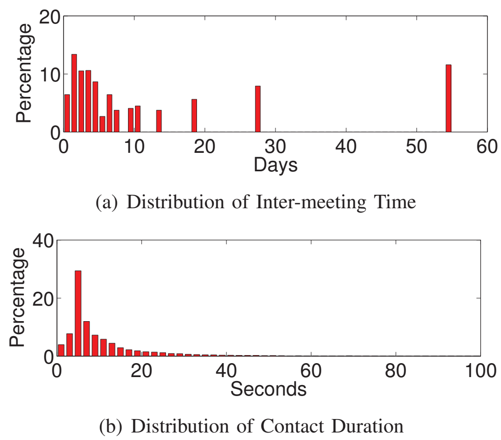


Fig. 1. Statistics of the UMassDieselNet trace show very limited forwarding opportunities among the nodes.

I. Introduction

Self-organized vehicular communication networks are under the general model of *Delay Tolerant Networks* (DTNs) [1]. A DTN is characterized by its short communication range, sparsity, high mobility, and short communication duration. Due to the short communication range and sparsity, a connected path between a source and a destination may not exist at any point in time, and messages need to be forwarded in a store-carry-forward paradigm, where messages are compared to mail and forwarding nodes are compared to postmen. Unlike the traditional routing in wireless network [2], DTN forwarding algorithms usually spawn and keep multiple copies of the same message in different nodes due to the uncertainty in node mobility and communication opportunities among nodes. The message is delivered if one of these nodes encounters the destination.

DTNs are typically limited in forwarding opportunities in terms of the number of contacts (connections established between a pair of nodes) and the duration of each contact. As shown in Figure 1, most of the average inter-meeting times between a pair of vehicular nodes are between 1 day and 10 days, and most contact durations are less than 20 seconds. Many opportunistic forwarding algorithms are proposed to make efficient use of the limited forwarding opportunities. Most of these algorithms, including our previous proposals [3], require the propagation of some kind of forwarding-information, e.g., the inter-meeting times between each pair of nodes, which unfortunately has to compete with the forwarding of data messages for the limited forwarding opportunities. Since the bandwidth required by the propagation of fowarding-information is usually proportional to the size of the network and the level of mobility, these algorithms are unsuitable for large-scale DTNs such as a self-organized vehicular network.

This paper proposes a forwarding algorithm that makes forwarding decisions based only on information collected locally. We define a local algorithm as one that does not require the exchange of forwarding-information among the nodes when no message needs to be forwarded. There are many previous work on message forwarding in DTNs with local information, including Spray-and-wait, which forwards blindly, and Delegation forwarding, which forwards copies only to the best nodes ever seen (delegates). Instead of using heuristics as in the previous work, the proposed algorithm uses local information to approximate the optimal forwarding strategy that we previously proposed in [3], which makes forwarding decisions based on global information, i.e., inter-meeting times between each pair of nodes. The value of the proposed algorithm is that it is the first opportunistic forwarding algorithm trying to optimize routing performance in DTNs that uses local information. Such a simple and localized forwarding algorithm is necessary in large-scale DTNs with limited communication opportunities.

The proposed forwarding algorithm is named *Localized Optimal Opportunistic Forwarding* (LOOF), which will be presented in two steps. We assume that the inter-meeting

978-1-4799-6036-1/14 $31.00 © 2014 IEEE
DOI 10.1109/MASS.2014.132

time is the only available forwarding-information, which is the most widely used forwarding-information in DTNs and is easy to calculate locally. In the first step of our presentation, we temporarily assume that global information, the inter-meeting times between each pair of nodes, is known to all nodes. With this global information, we derive an optimal forwarding algorithm by applying the optimal stopping rule. The only constraint in this algorithm is the limited number of forwardings per message, and the objective of it is to minimize the expected delay of the message. Then, in the second step of our presentation, we relax the assumption of global information and approximate the optimal forwarding algorithm using local information. Simulation results show that the proposed LOOF algorithm has an approximate performance to its well-informed version in most evaluation cases, and it outperforms other localized algorithms significantly in all evaluation cases. Our main contributions are summarized in the following.

- We propose the first localized forwarding algorithm for DTNs that minimizes the expected delays of the messages with a limited number of forwardings per message.
- We perform extensive trace-driven simulations to show the superior performance of the proposed algorithm over several representative localized forwarding algorithms.

II. Preliminaries and Overview

Our algorithms are developed from a depth-limited binary forwarding scheme, which controls the number of forwardings by limiting the traveling depth of each copy.

A. A Depth-limited Binary Forwarding Scheme

In a depth-limited binary forwarding algorithm, each message maintains a value, called *remaining hop-count*, which represents the maximum number of hops that the message can still be forwarded. When a message with a remaining hop-count k is forwarded from one node to another, the remaining hop-count of both copies in the two nodes becomes $k-1$. When $k=0$, the message cannot be forwarded to any node except the destination. That is, with the initial hop-count of a message being H, the maximum number of forwardings for the message is 2^H, including the one to the destination.

An advantage of this forwarding scheme is that it has a constant per message forwarding cost (assuming that the forwarding cost is the major cost in the whole communication process), which is necessary to achieve ultimate scalability: with a constant per node message rate, the per node forwarding overhead is kept constant as the network size increases.

B. Motivation and Overview

The proposed localized optimal opportunistic forwarding (LOOF) algorithm is developed as an approximation to an optimal opportunistic forwarding (OOF) algorithm that uses global information. This section provides an overview of OOF, followed by an introduction to LOOF.

To differentiate nodes of different capabilities in forwarding messages, most opportunistic forwarding algorithms develop indicators for the nodes. Representative indicators include: (1) direct (1-hop) indicators between each forwarding node and the destination, such as encounter frequency [4] and the time elapsed since the last encounter [5], [6], [7], [8], or (2) indicators along the expected forwarding path, such as expected cost [9] and expected delay [10]. When node i meets node j, node i forwards a message to node j only if the value of the indicator of node i is better than that of node j.

Two drawbacks can be found in such strategies that use a single indicator for each pair of forwarding node and destination. In the first place, a forwarding decision made by comparing the relative values of the indicators of two nodes, i and j, may not be a good one indeed: (1) the forwarding indicator value of j being better than that of i does not necessarily mean that j is a good forwarder; (2) even though the indicator value of j is good, i might encounter plenty of better nodes in the near future; (3) similarly, even though the indicator value of j is worse than that of i, j might still be better than other forwarding nodes that i encounters in the future.

Secondly, it would be better if a forwarding indicator has different values for different messages in a node, instead of having identical values for all messages in a node. For example, in the depth-limited binary forwarding scheme, an important state of each message is *remaining hop-count*, which changes each time when the message is forwarded. Remaining hop-count is an important factor: a node can be deemed as a bad 1-hop forwarding node for never connecting with the destination, but it can still be an excellent 2-hop forwarding node if it frequently contacts a node that frequently contacts the destination.

In our optimal forwarding algorithms, we use new indicators to rectify the above drawbacks. Firstly, to determine whether node i should forward a message copy to node j, our indicators do not compare the relative forwarding capability between nodes i and j, but they compare the relative forwarding capability of j with those of all of the other nodes that could contact node i, which rectifies the first drawback. Secondly, to rectify the second drawback, our indicators are not simple indicators for particular forwarding nodes, but are indicators for messages in each of their stages.

In this paper, we assume that the inter-meeting time is the only available forwarding-information, which is the most widely used forwarding-information in DTNs and is easy to calculate locally. In the optimal opportunistic forwarding algorithm with global information, we define an indicator, called expected delay $D_{i,d,k}$, for each message copy with remaining hop-count k, being stored in forwarding node i, and heading for destination d. With $D_{i,d,k}$, our optimal forwarding algorithm logically regards a message being forwarded from a node i to another node j as replacing the message copy with a new copy in each of the two nodes. The decision, regarding whether node i should forward the copy to j, is made by comparing the expected delay of the original copy in node i (before the forwarding) with the joint expected delay of the two new copies (after the forwarding). The calculation of the proposed expected delay is a result of the application of the optimal stopping rule, which will be discussed in Section III.

The proposed localized optimal opportunistic forwarding algorithm is developed by relaxing the assumption that global inter-meeting times between all pairs of nodes are known. When relaxing this assumption, some variables used in the calculation of the expected delays are not available locally and need to be approximated. Although limited by local information, the proposed algorithm still possesses the metric of optimal opportunistic forwarding in that: (1) it tries to evaluate the capability of a forwarding node for a given message among all possible forwarding nodes instead of simply comparing it with the current node, and (2) it uses the approximated expected delays, which reflect not only the delivery capacities of the forwarding nodes, but also the statuses of the messages being forwarded. Having these advantages is the reason that the proposed algorithm has a superior performance over other localized algorithms.

III. Optimal Opportunistic Forwarding Algorithm with Global Information

LOOF is an approximation to the optimal opportunistic forwarding (OOF) algorithm that uses global information. In this section, we will apply the optimal stopping rule to derive the expected delays of the messages and the optimal forwarding rule in OOF, similar to previous work [3].

A. Assumptions

The following is a description of our problem. Each message has a destination and is given a time-to-live (TTL) at its creation time. Different copies of the same message have no knowledge of the status of the other copies. A copy is deleted from a forwarding node only when it expires. We consider communication opportunity as the major bottleneck and assume an infinite buffer.

We assume that node mobility exhibits long-term regularities; such a feature is common in natural or human-related mobile networks. In OOF, we assume that each node knows the global information: the mean inter-meeting times $I_{i,j}$ between all pairs of nodes $\{i, j\}$ ($I_{i,j} = \infty$ if nodes i and j do not have any contact). On the other hand, in LOOF, we assume that each node i has all of the mean inter-meeting times, $I_{i,j}$s, of itself and its direct contacts, but not the inter-meeting times between other pairs of nodes, i.e., $I_{k,j}$s for $k \neq i$. In the calculation of the expected delays (not in the simulation), we use the assumption of exponential meeting probability of the nodes to simplify our calculation.

B. Expected Delay

We present the OOF algorithm starting from its forwarding indicator, *expected delay* ($D_{i,d,k}$), which is parameterized by k, the remaining hop-count. An expected delay $D_{i,d,k}$ denotes the expected time it takes to deliver a message with a remaining hop-count of k. The expected delay we define considers the joint expected delay of the 2^k copies of a message being forwarded by the 2^k forwarders in any possible forwarding tree. The 1-hop (directly) expected delivery delay $D_{i,d,0}$ of a message in node i is simply $I_{i,d}/2$, where $I_{i,j}$ is the mean inter-meeting time between nodes i and j. In the following, we derive the calculation of $D_{i,d,k}$ for the cases where the remaining hop-count $k > 1$.

Algorithm 1 Calculation of $D_{i,d,k}$ for $k > 1$

```
W_{i,N} := I_{i,d}/2
D_{i,d,k} := ∞
∑ := 0
Q := a priority queue of j in increasing order of D_{j,d,k-1}
while (j := dequeue(Q) and D_{i,d,k} > W_{i,N} × (1 + ∑)) {
    D_{i,d,k} := W_{i,N} × (1 + ∑)
    W_{i,N} := 1 / (1/W_{i,N} + 2/I_{i,j})
    ∑ := ∑ + 2 / (I_{i,j} × (1/D_{i,d,k-1} + 1/D_{j,d,k-1}))
}
```

The expected delay and optimal forwarding rule in OOF cannot be presented seperatively: (1) the optimal forwarding rule makes forwarding decisions based on the expected delay, and (2) the expected delay results from the fact that the optimal forwarding rule is applied.

We will present our optimal forwarding rule first. When a copy, whose remaining hop-count is k, is in node i, and node i meets node j, the decision on whether to forward depends on if replacing the copy in i with two new copies in i and j, respectively, will result in an increased joint expected delay. If the message is not forwarded, the copy's expected delay remains unchanged. On the other hand, if the message is forwarded, we have two new copies with a remaining hop-count of $k - 1$ in both node i and node j, whose expected delays are $D_{i,d,k-1}$ and $D_{j,d,k-1}$, respectively. To maximize the expected delay, we use the optimal forwarding rule, which forwards the message only if the joint expected delay of the two copies (in the case of forwarding) is greater than the that of the single copy (in the case of no forwarding).

We assume that the meeting probability of the nodes follows the exponential distribution to simplify our calculation: if two copies of a message have expected delays, D_1 and D_2, respectively, then their joint expected delay equals $\frac{1}{\frac{1}{D_1}+\frac{1}{D_2}}$. This is because, the probabilities of delivery for the copies are $p_1 = 1 - e^{-\frac{t}{D_1}}$ and $p_2 = 1 - e^{-\frac{t}{D_2}}$, respectively, within a period of time t. Therefore, the probability of delivery for one of the copies within t is $1-(1-p_1)\times(1-p_2) = 1-e^{-t(\frac{1}{D_1}+\frac{1}{D_2})}$. With the above assumption, the optimal forwarding rule can be described as: node i forwards, to node j, a message copy with remaining hop-count k and destination d if and only if:

$$D_{i,d,k} > \frac{1}{\frac{1}{D_{i,d,k-1}} + \frac{1}{D_{j,d,k-1}}}. \tag{1}$$

C. Forwarding as an Optimal Stopping Rule

To minimize the statistic delay of each message, we model each forwarding as an optimal stopping rule problem as follows. We will define a simple objective of minimizing the expected delay in each single forwarding, instead of choosing a more difficult objective like minimizing the expected delay in the whole forwarding process. Consider a single forwarding of a message in node i with remaining hop-count k. 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. Upon meeting with node j, i can either forward the copy to j, or not. In the following, we will derive the optimal forwarding rule, and as a result of which, the expected delay of each message is minimized.

The expected delay $D_{i,d,k}$ of a message partially depends on each node j whose expected delay $D_{j,d,k-1}$ satisfies the forwarding criteria in Equation 1. Suppose that under the optimal forwarding rule, N is a set of nodes that satisfy the forwarding criteria. The destination d of the message always belongs to N. Let $W_{i,N}$ be the average waiting time for i to encounter the first node in N, and let $p_{i,j}$ be the probability that j is the first node to encounter i among all nodes in N ($\sum_{j\in N} p_{i,j} = 1$).

Assuming that inter-meeting times are not correlated, we have $W_{i,N} = \frac{1}{\sum_{j\in N}\frac{2}{I_{i,j}}}$, where $I_{i,j}$ is the average inter-meeting time between nodes i and j, and $I_{i,j}/2$ is therefore the average encountering time between the two nodes. Here, the inter-meeting time is the expected waiting for the next contact starting from the end of the previous contact, while the average encountering time is the expected waiting for the next contact average on any starting time. Since the meeting probabilities $p_{i,j}$s for different nodes j are proportional to the meeting probability between nodes i and j, which in turn are reversely proportional to the inter-meeting times between nodes i and j (i.e., $\frac{p_{i,j}}{p_{i,k}} = \frac{\frac{1}{I_{i,j}}}{\frac{1}{I_{i,k}}}$), we have $p_{i,j} = \frac{p_{i,j}}{\sum_{k\in N} p_{i,k}} = \frac{\frac{1}{I_{i,j}}}{\sum_{k\in N}\frac{1}{I_{i,k}}} = \frac{2W_{i,N}}{I_{i,j}}$.

The first node that the current node i encounters can be the destination d or another node $j \in N\backslash\{d\}$. The probability of the first case is $p_{i,d}$, and the expected delay in this case is $W_{i,N}$, which is the average time that node i encounters the first node in N. On the other hand, the probability that node i encounters some node $j \in N\backslash\{d\}$ is $p_{i,j}$, in which case the expected delay is $W_{i,N} + \frac{1}{\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}}}$, where $\frac{1}{\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}}}$ is the joint expected delay of the two copies in nodes i and j after the two nodes encounter each other. To sum up, $D_{i,d,K}$ can be derived as follows: $D_{i,d,K} =$

$$
\begin{aligned}
& p_{i,d}\times W_{i,N} + \sum_{j\in N\backslash\{d\}} p_{i,j}\times (W_{i,N} + \frac{1}{\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}}}) \\
&= W_{i,N} + \sum_{j\in N\backslash\{d\}} p_{i,j}\times \frac{1}{\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}}} \\
&= W_{i,N} + \sum_{j\in N\backslash\{d\}} \frac{2W_{i,N}}{I_{i,j}}\times \frac{1}{\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}}} \\
&= W_{i,N}\times (1 + \sum_{j\in N\backslash\{d\}} \frac{2}{I_{i,j}\times(\frac{1}{D_{i,d,k-1}}+\frac{1}{D_{j,d,k-1}})}) \quad (2)
\end{aligned}
$$

From the above equation, $D_{i,d,k}$ can be calculated with (a) $I_{i,j}$s for all nodes j, (b) $D_{j,d,k-1}$s for all nodes j (which implicitly requires $I_{j,n}$s between all pairs of nodes j and n), and (c) the set of nodes N that satisfies the forwarding criteria in Equation 1. Firstly, the $I_{j,n}$s between all pairs of nodes j and n are known as we assumed. Secondly, the expected delays $D_{j,d,k-1}$s for all nodes j are known if we calculate all the expected delays in the increasing order of k. That is, we firstly calculate $D_{j,n,1}$s for all pairs of nodes j and n (with $D_{j,n,0} = \frac{I_{j,n}}{2}$), which is followed by $D_{j,n,2}$s, and so on.

The set of nodes N can be determined by the following two principles. (1) Since the purpose of forwarding an additional copy is to decrease the expected delay, the inclusion of a node j into N should always make $D_{i,d,k}$ smaller. (2) Regarding its inclusion into N, a node j with a smaller expected delay $D_{j,d,k-1}$ is always preferred over another node with a larger expected delay. Therefore, N can be constructed by adding each node j into N in the increasing order of $D_{j,d,K-1}$, until the value of $D_{i,d,K}$ stop decreasing. Algorithm 1 shows the calculation of $D_{i,d,k}$ with a given k. This algorithm is invoked H (the maximum possible hop-count) rounds, starting with $k = 1$, and ending with $k = H$, and in each round the algorithm is again invoked for all pairs of nodes i and d.

IV. Localized Optimal Opportunistic Forwarding Algorithm

The localized optimal opportunistic forwarding algorithm, LOOF, has the same objective as the previously presented optimal opportunistic forwarding algorithm, OOF, that uses global information. With only local information, the former approximates the latter by making its best estimations from its local information about the unknown parameters used by the latter.

A. Approximation with Local Information

For simplicity, we use the same notations with caps to denote the estimated parameters in LOOF. In Algorithm 1, the unknown parameters for LOOF are $D_{j,d,k-1}$s for all nodes j, which can hardly be estimated with local information. Also note that in Algorithm 1, only the sum of the smallest expected delays $D_{j,d,k-1}$s is used, and the rest of them are not important. We assume that each destination, which can be d, i, j, or other nodes, is in frequent contact with only a small percentage of all nodes, and the expected delay for every node toward each destination follows a similar distribution. Under this assumption, we use the sum of the smallest expected delays $D_{j,i,k-1}$s to estimate the sum of the smallest expected delays of $D_{j,d,k-1}$s. Also, we assume that $D_{j,i,k-1} \approx D_{i,j,k-1}$, and thus we can use the sum of the smallest $D_{i,j,k-1}$s to estimate that of $D_{j,d,k-1}$s.

In Equation 2, the probabilities $p_{i,j}$s (represented by $I_{i,j}$s in the last two steps of the equation) are no longer valid for $D_{j,d,k-1}$s when they are estimated by $D_{i,j,k-1}$s. We simply replace any $I_{i,j}$ with an $\hat{I} = \lambda \times I^{avg}$, where I^{avg} is the average of all $I_{i,j}$s, and $\lambda \gg 1$ is a pessimistic parameter. We use λ because that the nodes j with the smallest $D_{i,j,k-1}$

Algorithm 2 Calculation of $\hat{D}_{i,d,k}$ for $k > 1$

1: $W_{i,N} := I_{i,d}/2$
2: $\hat{D}_{i,d,k} := \infty$
3: $\sum := 0$
4: $Q :=$ a priority queue of j in increasing order of $\hat{D}_{i,j,k-1}$
5: **while** (j :=dequeue(Q) **and** $\hat{D}_{i,d,k} > W_{i,N} \times (1+\sum)$) {
6: $\quad \hat{D}_{i,d,k} := W_{i,N} \times (1+\sum)$
7: $\quad W_{i,N} := \frac{1}{\frac{1}{W_{i,N}}+\frac{2}{\hat{I}}}$
8: $\quad \sum := \sum + \frac{2}{\hat{I}\times(\frac{1}{\hat{D}_{i,d,k-1}}+\frac{1}{\hat{D}_{i,j,k-1}})}$
9: }

are direct contacts of node i, while they may not be direct contacts of d, and the over-estimation of $D_{j,d,k-1}$s will harm the forwarding performance for being too optimistic about wasting the limited forwarding opportunity.

With the above approximation, the expected delays $\hat{D}_{i,d,k}$s can be calculated locally as listed in Algorithm 2. The only differences between Algorithm 2 and Algorithm 1 are that any $D_{j,d,k-1}$ is replaced by $\hat{D}_{i,j,k-1}$ in line 8 and $I_{i,j}$s are replaced by $\hat{I}$ in lines 7 and 8. In this algorithm, the expected delays $\hat{D}_{i,j,k-1}$s for all nodes j are known if we calculate all expected delays in the increasing order of k. The storage complexity of the calculation of Algorithm 2 in each node is $O(NH)$, where N is the number of nodes, and H is the maximum hop-count of all messages.

The forwarding algorithm in LOOF is described as follows. The algorithm executes when node i with a message m, whose destination is d, encounters node j. Node i first requests $\hat{D}_{j,d,k-1}$ from node j. It forwards a copy of m to node j if $\hat{D}_{i,d,k} > \frac{1}{\frac{1}{\hat{D}_{i,d,k-1}}+\frac{1}{\hat{D}_{j,d,k-1}}}$, where $\hat{D}_{i,d,k}$ and $\hat{D}_{i,d,k-1}$ are locally available to node i.

B. Remarks

Although we use only local information, LOOF still possesses the merits of OOF in that: (1) it evaluates the capability of a forwarding node for a given message among all possible forwarding nodes, instead of performing simple comparison between the forwarding node and the current node, and (2) the estimated expected delays that it uses reflect not only the delivery capacities of the forwarding nodes, but also the statuses of the messages being forwarded. Having these advantages is the reason that the proposed algorithm has a superior performance over the compared localized forwarding algorithm, as shown in our simulation results.

A hybrid algorithm with partial information $I_{j,n}$ can be devised by replacing the known $I_{j,n}$s with $\hat{I}$ in Algorithm 2 to improve performance in practice.

V. Evaluation

We evaluate the proposed algorithm, LOOF, against other forwarding algorithms by using four Cambridge Haggle traces [11] and the UMassDieselNet trace [9]. The forwarding algorithms that we implement to compare to LOOF are the optimal opportunistic forwarding algorithm with global information, or OOF, which is described in Section III, and those that are listed in Section V-A. Since all of the algorithms that we implement aim to compare different forwarding indicators, other optimizations that have orthogonal effects on the performance of these algorithms are not implemented. These optimizations can be added to all of our implemented algorithms, and they are expected to improve the forwarding performance of all of them. They may include buffer management [9], estimation of global message delivery probability [6] and social centrality of the nodes [12], the use of position information [13], [14], as well as acknowledgment mechanisms [9], [6]. The initial value of hop-count H is chosen to be 3 in our evaluation, which makes all of the compared algorithms have similar numbers of forwardings.

A. Protocols in Comparison

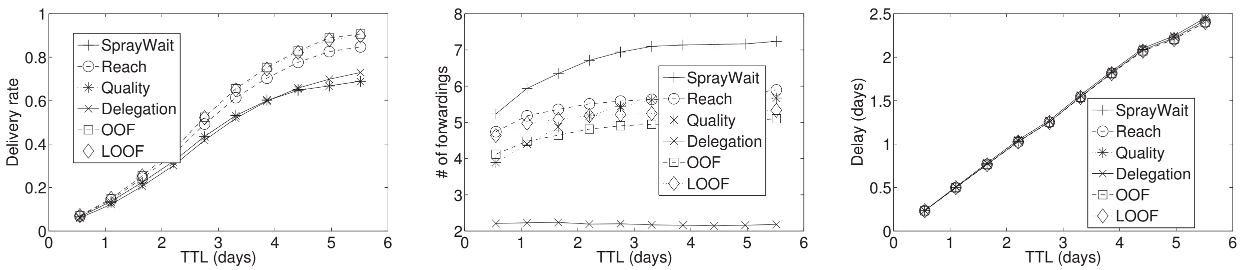
We compare the proposed optimal opportunistic algorithms against OOF and several opportunistic forwarding algorithms. While the proposed algorithms have a well-defined objective (i.e., to minimize the expected delay in each forwarding), other algorithms use either heuristic forwarding rules or blind forwarding. Many recently proposed forwarding algorithms are evaluated in the traces we used, which we will not compare with since they are proposed based on different knowledge of the network. The compared algorithms include: *Spray-and-wait* [15], *Quality forwarding* [16], *Delegation forwarding*, *Reach forwarding* is a simple algorithm we devise to compare with LOOP. It sends a copy of the message to every node encountered that has a chance to encounter (i.e., can reach) the destination.

B. Results and Discussions

In the following, we will present the results of our evaluation. Due to space limitation, only the simulation results in two traces, the Cambridge trace and the UMassDieselNet trace, are shown in Figures 2 and 3. Each row of figures shows the results belonging to the same trace. The forwarding algorithms are compared in terms of delivery rate, number of forwardings (cost), and delay in columns (a), (b), and (c), respectively. The only simulation variable is the TTL of the messages.

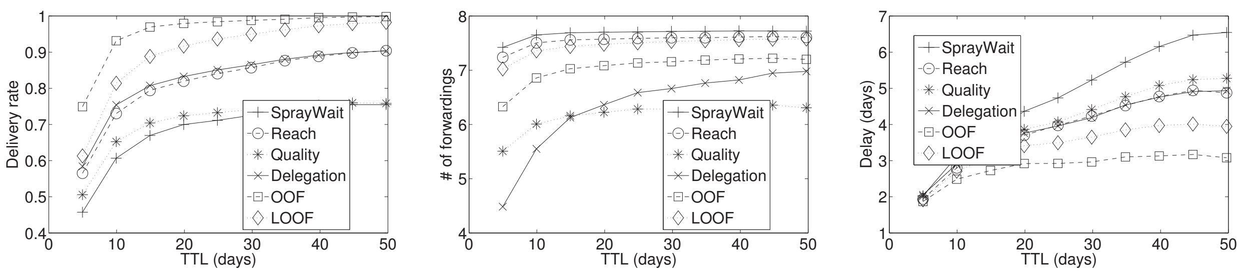
The most important metric in our simulation is the delivery rate. In three of the traces: Cambridge (Figures 2(a)), Infocom2006, and Infocom, LOOF has similar delivery rate to that of OOF. In the trace Content, LOOF delivers 30% more messages than the second-best algorithms, Delegation and Reach, and delivers twice the number of messages than Spray-and-wait and Quality. In the trace UMassDieselNet (Figure 3(a)), LOOF delivers 15% more messages than the second-best algorithms, Delegation and Reach, and delivers 60% more messages than Spray-and-wait and Quality.

We measure the costs of the forwarding algorithms in terms of the average number of forwardings per message, which is calculated from the messages that are delivered by all forwarding algorithms. The number of forwardings may



(a) Delivery rate in Cambridge (b) Number of forwardings in Cambridge (c) Delay in Cambridge

Fig. 2. Simulation results in trace Cambridge



(a) Delivery rate in UMassDieselNet (b) Number of forwardings in UMassDieselNet (c) Delay in UMassDieselNet

Fig. 3. Simulation results in UMassDieselNet trace (bus-route-based)

increase or decrease as the TTL changes. The number of forwardings in all forwarding algorithms is less than 8.

In the trace UMassDieselNet (Figure 3(c)), the delay of LOOF is about 15% smaller than other localized forwarding algorithms, while in other traces: Cambridge (Figures 2(c)), Infocom2006, and Infocom, Content, delays are similar.

To sum up, the delivery rate of LOOF approximates that of the optimal opportunistic forwarding algorithm with global information (OOF) in some traces, and its delivery rate is much better than the compared localized forwarding algortihms in all traces.

VI. Conclusion

In this paper, we proposed a localized optimal opportunistic forwarding algorithm, which minimizes delay with a limited number of copies per message. The proposed forwarding algorithm makes forwarding decisions based only on information collected locally; no information, except for the real data, is required to be exchanged among the nodes, which makes this forwarding algorithm particularly suitable for large-scale DTNs. Simulation results showed that the proposed algorithm approximates the performance of the optimal forwarding algorithm with global information in several traces.

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. Boukerche, X. Cheng, and J. Linus. Energy-aware Data-centric Routing in Microsensor Networks. In *Proceedings of ACM MSWiM*, 2003.

[3] C. Liu and J. Wu. On Multicopy Opportunistic Forwarding Protocols in Nondeterministic Delay Tolerant Networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(6):1121–1128, June 2012.

[4] A. Lindgren, A. Doria, and O. Schelen. Probabilistic Routing in Intermittently Connected Networks. *Lecture Notes in Computer Science*, 3126:239–254, August 2004.

[5] 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.

[6] A. Balasubramanian, B. N. Levine, and A. Venkataramani. DTN Routing as a Resource Allocation Problem. In *Proc. ACM SIGCOMM*, 2007.

[7] 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.

[8] 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.

[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] M. Y. Uddin, H. Ahmadi, T. Abdelzaher, and R. Kravets. A Low-energy, Multi-copy Inter-contact Routing Protocol for Disaster Response Networks. In *Proc. of IEEE SECON*, 2009.

[11] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD data set cambridge/haggle (v. 2006-09-15). http://crawdad.cs.dartmouth.edu/cambridge/haggle, September 2006.

[12] E. Daly and M. Haahr. Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs. In *Proc. of ACM MobiHoc*, 2007.

[13] U. Acer, S. Kalyanaraman, and A. Abouzeid. Weak State Routing for Large Scale Dynamic Networks. In *Proc. of ACM MobiCom*, 2007.

[14] X. Cheng, A. Thaeler, G. Xue, and D. Chen. TPS: A Time-based Positioning Scheme for Outdoor Sensor Networks. In *Proceedings of IEEE INFOCOM 2004*, 2003.

[15] 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.

[16] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot. Delegation Forwarding. In *Proc. of ACM MobiHoc*, 2008.