A Utility-based Multi-copy Forwarding Algorithm in Delay Tolerant Networks

Cong Liu†, Yan Pan†, Jun Zhang†, Jiacheng Zhuo†, Mingjun Xiao‡, Ai Chen¶, and Jie Wu§
†Sun Yat-sen University, ‡University of Science and Technology of China,
¶Shenzhen Institutes of Advanced Technology, CAS, §Temple University
{liucong3, panyan5, issai}@mail.sysu.edu.cn, xiaomj@ustc.edu.cn, ai.chen@siat.ac.cn, jiewu@temple.edu

***Abstract*—In *delay tolerant networks* (DTNs) with uncertainty in node mobility, message forwarding usually employs a multi-copy, opportunistic forwarding scheme to improve delivery probability. Minimizing the delivery latency (delay) and minimizing the number of forwardings (message copies) have been two conflicting goals. On the other hand, utility-based routing has been proposed in multi-hop wireless networks, where a utility is defined as a compound metric that provides a trade-off between delay and forwarding cost. In this paper, we propose the first multi-copy utility-based opportunistic forwarding algorithm in DTNs. The proposed *multi-copy opportunistic utility-based forwarding* (MOUF) algorithm is able to improve the overall utility representing a user preferred trade-off between delay and forwarding cost so as to satisfy different requirements. Theoretical analysis and extensive simulations are conducted to verify the improved performance of the proposed algorithm.**[1]

***Index Terms*—Delay Tolerant Networks, Opportunistic Forwarding, Simulation, Utility.**

I. INTRODUCTION

A *Delay Tolerant Network* (DTN) [1], [2] is a sparse wireless mobile network, where a contemporary connected path may not exist between a pair of source-destination nodes. Thus, traditional connection-based routing algorithms may fail. Messages in a DTN need to be forwarded in a mobility-assisted way. Applications in DTNs include mobile social software [3], pocket switch networks [4], vehicle and pedestrian networks [5], low duty cycle sensor networks [6], deep-space satellite networks, underwater acoustic buoy networks, and many dedicate networks for developing regions.

The challenges in DTN forwarding algorithms is mostly due to the uncertainty in node mobility and connectivity in many DTNs. With these uncertainties, allowing only a single forwarder for each message at a time may not be reliable enough to provide a small delivery latency (delay) or a high delivery rate. Therefore, the multi-copy, opportunistic schemes are usually adopted, in which multiple copies of each message are opportunistically pawned and simultaneously kept by multiple nodes. The message is delivered if one of these nodes encounters the destination. The broadcast-based Epidemic algorithm [7] forwards a copy of each message to every node and guarantees a minimum delay. However, the tremendous forwarding cost induced by Epidemic makes it impractical in DTNs with a large network size, limited node battery power, small bandwidth, and intermittent connectivity. To reduce the major cost associated with forwarding and to provide a good delivery performance, much effort has been focused on opportunistic forwarding [5], [8].

While enhancing the energy efficiency in terms of either delivery performance or forwarding cost, a clear objective cannot be defined that quantifies a particular user preference between delay and cost. Delay and forwarding cost are often the two conflicting objectives in existing forwarding algorithms: some of these algorithms, such as [9], seek to minimize forwarding cost with a bounded minimum delay. On the other hand, other algorithms, such as [10], manage to minimize delay with the constraint of a fixed forwarding cost. Clearly, these simple-objective-based algorithms cannot support compound objectives that make trade-offs between delay and forwarding cost.

Utility-based routing has been proposed in multi-hop wireless networks [11], [12], where nodes are allowed to charge for message forwarding to provide an incentive for the cooperation among the nodes. In these networks, a source node needs to consider only only the benefit provided by a successful delivery, but also the forwarding costs that are charged by the forwarding nodes. To calculate the overall gain of the forwarding of a message in the network, the concept of utility [11] is proposed, which is a compound metric incorporating benefit and cost. By maximizing utility, a balance between the benefit of delivery and the forwarding cost is provided.

However, extending utility-based forwarding algorithms to DTNs is not straightforward. The challenge lies in the fact that the utility in [11], [12] is defined for single-copy multi-hop routing algorithms, where the delivery benefit is defined as the initial benefit multiplied by the joint successful probability of forwarding in a multi-hop path. And the forwarding cost is defined as the sum of the charges by each node on this path. Clearly, this definition cannot be applied to a forwarding algorithm, in which forwarding multiple copies of a message are allowed to be forwarded along multiple paths. In this case, the copy on a particular path does not know the successful probability or the forwarding costs of other copies on different paths. As a result, the overall utility of all copies of the

---

[1] This work was funded in part by National Science Foundation of China (grant No. 61370021, 61003045, 61003296, 61201245, 61003241), Natural Science Foundation of Guangdong Province, China (grant No. S2013010011905), Shenzhen Overseas High-level Talents Innovation and Entrepreneurship Funds under Grant No. KQC201109050097A, NSF grants ECCS 1231461, ECCS 1128209, CNS 1138963, CNS 1065444, and CCF 1028167.

message cannot be calculated. Although a single-copy utility-based forwarding algorithm is proposed in our previous work [13], a multi-copy forwarding algorithm is more suitable for DTNs with uncertain node mobility.

In this paper, we propose the first *multi-copy opportunistic utility-based forwarding* (MOUF) algorithm in DTNs. As one of the implementation options, we incorporate delay and forwarding cost into our definition of utility. The objective of MOUF is to maximize this utility. This objective cannot be achieved by minimizing either delay or forwarding cost alone. Due to the uncertainty in node mobility, there is no deterministic solution. Also, a solution that requires a massive propagation of timely control information and a large computation overhead is not desired in DTNs. Therefore, MOUF uses a simple heuristics to maximize the expected utility, which incorporates the expected delay and the expected forwarding cost across different forwarding paths.

Extensive trace-driven simulations are performed using the UMassDieselNet trace [5] and the Cambridge Haggle trace set [14]. In the evaluation, MOUF shows significant improvement in terms of utility when compared with non-utility-based forwarding algorithms. The contributions of this paper are summarized in the following:

- While most existing multi-copy DTN forwarding algorithms use simple objectives, such as delivery rate, we are the first to introduce a utility-based opportunistic forwarding scheme.
- We further give a utility definition for DTNs, of which existing simple objectives are special cases. Then, we propose the MOUF algorithm to improve such utility.
- We conduct an analytical study on the improved performance of MOUF in terms of expected utility.
- We further evaluate the improved performance of MOUF by performing extensive simulations on real-world mobility traces.

The rest of this paper is organized as follows: Section II reviews the preliminaries on utility-based routing and shows our network model and motivations. The proposed forwarding algorithm is presented in Section III. Sections IV and V study expected utility of MOUF analytically and show our simulation methods and results, respectively. Related work on DTN forwarding and utility-based routing is reviewed in Section VI. Finally, the paper is concluded with future work in Section VII.

II. PRELIMINARIES, MODEL, AND MOTIVATIONS

This section provides some preliminaries about utility-based single-copy routing in wireless networks, which is followed by our DTN network model and the motivation for our utility-based multi-copy forwarding algorithm.

A. Utility-based single-copy routing

In utility-based routing [11], nodes are allowed to charge for message forwarding, as long as they have an incentive to cooperate with each other in message forwarding. In this case, a source node needs to consider not only the benefit as the result of a successful delivery, but also the cumulative forwarding cost charged by the forwarding nodes.

For a source s intending to send a message to a destination d via a single-hop path, the transmission cost and reliability from s to d are denoted by c and p, respectively. If a transmission is successful, s will obtain a benefit b and incur a cost c, and its utility is defined by $b - c$. Otherwise, if the transmission fails, the utility of s is defined by $0 - c$. With a successful probability p, the expected utility is

$$u = p \cdot b - c. \tag{1}$$

In a multi-hop path $\{n_i\}$, with $0 \leq i \leq m$, the source $s = n_0$, and the destination $d = n_m$, let the cost and the reliability from n_i to n_{i+1} be c_i and p_i, respectively. The expected utility can be derived by using Equation 1 in a backward-fashion. For instance, in a 2-hop path, the utility of the second hop is $u_1 = p_1 \cdot b - c_1$, and the utility of the first hop is $u = u_0 = p_0 \cdot u_1 - c_0 = p_0 \cdot (p_1 \cdot b - c_1) - c_0 = p_0 \cdot p_1 \cdot b - (p_0 \cdot c_1 + c_0)$. In general, the expected utility in a multi-hop path is

$$u = (\prod_{i=0}^{m-1} p_i) \cdot b - \sum_{i=0}^{m-1} (c_i \prod_{j=0}^{i-1} p_j). \tag{2}$$

In [12], Equation 2 is applied to perform an exhaustive search on all paths between source and destination to find the optimal multi-hop opportunistic path. However, Equation 2 is defined on a single-copy multi-hop path. It is not applicable to multi-copy multi-path forwarding algorithms. In [13] a time-sensitive utility is defined, however, the forwarding algorithm proposed there is a single-copy forwarding algorithm.

B. Network model

Let N be the set of all nodes and $I_{i,j}$ be the average inter-contact time between nodes i and j. In many realistic DTNs, such as *vehicular networks* [5] and *pocket switch networks* [4], the mobility of the nodes are mostly natural or human-related, which exhibits long-term regularities. In other words, some pairs of nodes consistently meet more frequently than other pairs over time. This property is frequently used to facilitate opportunistic forwarding in DTNs. We assume that each node i locally collects its average inter-contact time $I_{i,j}$'s with node j's that it has a chance to contact over a warm-up period. If nodes i and j have no contact, $I_{i,j} = \infty$. For simplicity, we further assume that $I_{i,j}$'s are time invariant, and we relax this assumption in our simulation.

Each message has a source and a destination, and it is given a time-to-live at its creation time. In DTNs with uncertain node mobility, we assume that a multi-copy and opportunistic forwarding scheme is used, and different copies of the same message are forwarded independently without any knowledge of the other copies. Once forwarded by a node, a copy is deleted only when the message expires.

We assume that nodes charge differently for message forwarding. Particularly, we set $c_i = \bar{c} \cdot \frac{f_i}{f}$ as the charge of node i for receiving, storing, and subsequently forwarding copies of the message to other nodes, where $\bar{c}$ is the average forwarding

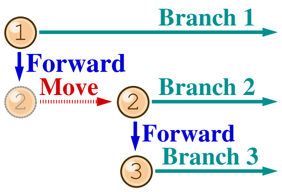


Fig. 1. A multi-copy forwarding algorithm creates multiple copies of a message and forwards them in different branches of the forwarding tree. The source (node 1) forwards a copy to node 2, which creates a copy forwarded in a second branch. Then, node 2 forwards a copy to node 3 and creates a third branch. The copies on the first branch might not have the complete information about the copies on the second or the third branch. Therefore, each copy might not be able to calculate the exact utility of the message based on its local view.

cost, $f_i = \sum_{j \in N, j \neq i} \frac{1}{I_{i,j}}$ is the overall contact frequency between node i and the other nodes, and $\bar{f} = \frac{1}{|N|} \sum_{i \in N} f_i$ is the average contact frequency in the network. This definition of charges c_i's can potentially improve load-balancing and provide better delivery quality for important and emergent messages.

C. Motivation

DTN forwarding algorithms using simple metrics, such as delivery rate, delay and forwarding cost, cannot provide user-preferred trade-offs between these metrics. For example, an algorithm that maximize delivery rate might create excessive forwarding cost; an algorithm that bounds forwarding cost inappropriately might results in low delivery rate.

Existing utility definitions [11], [12] are designed for single-copy multi-hop routing algorithms, which is not directly applicable to the multi-copy forwarding schemes. Specifically, the definition of utility, as showed in Equation 2, requires the forwarding cost c_i for all nodes that forward the message. However, as showed in Figure 1, copies forwarded on different branches of the forwarding tree might not have information about each other. Therefore, the exact calculation of such utility is infeasible in DTNs.

III. MULTI-COPY OPPORTUNISTIC UTILITY-BASED FORWARDING ALGORITHM

In this section, we will describe the proposed *multi-copy opportunistic utility-based forwarding* (MOUF) algorithm. MOUF employs simple heuristics, which effectively shows a significant performance gain in terms of utility when compared with non-utility-based forwarding algorithms. In the following, we will first describe our choice of utility before we go into the details of our main algorithm.

A. Utility definition

Since delay and cost is two important metrics for DTN forwarding protocols, we define the utility u of a successfully delivered message with multiple copies as

$$u = b - \Delta \cdot t - c, \tag{3}$$

where b is the initial benefit of the message, t is the delay of the message, Δ is the *benefit decay* that penalizes delay, $c = \sum c_i$ is the total forwarding charges c_is by every message forwarder, i. Different benefit decays provide different user-preferred trace-offs between delay t and forwarding cost c. For instance, an message that is emergent or important should have a large benefit decay Δ, which will result in more forwardings in order to reduce delay.

The successful delivery of a message within a time-to-live is important in DTNs, and we let $b = \Delta \cdot T$, where T is the time-to-live of the message. For a message that fails to be delivered within the time-to-live T, i.e., $b - \Delta \cdot t < 0$, we define its utility as

$$u = -c. \tag{4}$$

Our utility definition is just one among the many possible definitions of utility. Nevertheless, different user preferences can be embodied in this definition. For instance, for messages with the same time-to-live, a large initial benefit b and a large benefit decay Δ can be assigned to an emergent message, so that it can be forwarded by more forwarders including those charge more for forwarding.

Simple metrics in existing forwarding algorithms, such as delivery rate, delay and forwarding cost can be regarded as special cases of our utility. For example, an algorithm that only maximizes delivery rate, or equivalently minimizes delay, can use very large b and Δ, such that the forwarding cost c does not take effect; an algorithm that only bounds the number of fowardings to k can set $\Delta = 0$ and $b = k$ and use unit forwarding cost: $c_i = 1$.

B. A simplified forwarding scheme

As showed in Figure 1, copies forwarded on different branches of the forwarding tree might not have information about each other. Therefore, the exact total forwarding charges $c = \sum c_i$ by all forwarders is generally not available for all forwarder.

To correctly calculate the total forwarding charges, MOUF adopts a simple approach as follows: We define the *key-forwarder* k of a message as the forwarder that has the smallest inter-contact time $I_{k,d}$ with the destination d among all forwarders of the message. We require that all forwarders, except for the key-forwarder, can only forward the message to the destination, and the key-forwarder can forward copies to the destination and other nodes. If a message is forwarded to a new forwarder that has the smallest inter-contact time with the destination, the new forwarder becomes the key-forwarder and the original key-forwarder becomes an ordinary forwarder. Note that, there is exactly one key-forwarder for each message at any time.

With this simplified forwarding approach, all forwarders are in a single branch, and therefore the current key-forwarder knows the exact total forwarding charges c. Also, only the key-forwarder needs the exact total forwarding charges to calculate the utility so as to make the forwarding decision, since the ordinary forwarders are only allowed to forward the message to the destination.

One might worry about the performance of this simplified approach. Fortunately, a simplified two-hop forwarding

Algorithm 1 MOUF (for ordinary forwarders)

```
$j \leftarrow$ the current node
$i \leftarrow$ a node that $j$ encounters
$m \leftarrow$ each message in $j$ of which
        $j$ is an ordinary forwarder
if (the destination of $m$ is $i$)
    forward $m$ to $i$
```

scheme [15], where the key-forwarder is restricted to be the source node, is showed to have comparable forwarding performance in simulations using a wide range of real mobility traces.

C. Utility maximization

The objective of MOUF is to maximize the expected utility. To realize this, we use a simple heuristic, which requires an increasing utility in each forwarding. With the utility definition for successfully delivered messages as given in Equation 3, the expected utility is

$$\hat{u} = b - \Delta \cdot \hat{t} - c, \tag{5}$$

where the $\hat{t}$ is the local estimation of the key-forwarder about the overall delay of the copies of the message on all of its forwarders, and

$$\hat{t} = t' + \hat{d}, \tag{6}$$

where t' is the time elapsed since the creation of the message and $\hat{d}$ is the estimated delay starting from the current time. To ease the design of our algorithm, we assume that inter-contact time is oblivious and independent. With this assumption, we estimate $\hat{d} = \frac{1}{\sum \frac{1}{I_{i,d}}}$ using the combination of the inter-contact times $I_{i,d}$'s of all forwarders i's for the copies of the message.

When a message is created in its source node s with a given b, Δ, and destination d, the initial values are $t' = 0$, $\hat{d} = I_{s,d}$, and $c = 0$, where $I_{s,d}$ is the average inter-contact time between the source and the destination.

Whenever the key-forwarder k of message m is connected with another node i, a decision is made about whether to forward a copy of message m to node i. The two options of k are given below:

- If a copy is forwarded to node i, the expected utility of the message becomes $\hat{u}^{new} = b - \Delta \cdot (t' + \frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}) - (c + c_i)$.
- Otherwise, the expected utility of the message remains $\hat{u} = b - \Delta \cdot (t' + \hat{d}) - c$.

In the first option, $I_{i,d}$ is the average inter-contact time between node i and destination d, c_i is the forwarding charge of node i, and $\frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}$ is the joint expected delay.

To have an increasing utility in each forwarding, a copy is only forwarded when $\hat{u}^{new} > \hat{u}$. Once forwarded, both copies will have their $\hat{d}$'s updated to $\frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}$.

Finally, with the utility definition for failed messages in Equation 4, no further message forwarding is necessary when $b - \Delta \cdot \hat{t} < 0$. This is because any forwarding charge added to $\hat{c}$ will only decrease the expected utility $\hat{u} = -c$.

Algorithm 2 MOUF (for the key-forwarder)

```
$k \leftarrow$ the current node
$i \leftarrow$ a node that $k$ encounters
$m \leftarrow$ each message in $k$ of which
        $k$ is the key-forwarder
if (the destination of $m$ is $i$)
    forward $m$ to $i$
else
    $\hat{u} = b - \Delta \cdot (t' + \hat{d}) - c$
    $\hat{u}^{new} = b - \Delta \cdot (t' + \frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}) - (c + c_i)$
    if ($\hat{u}^{new} > \hat{u}$)    // a positive utility gain
        $\hat{d} \leftarrow \frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}$
        $c \leftarrow c + c_i$
        forward a clone of $m$ to node $i$
        if ($I_{i,d} < I_{k,d}$)
            node $i$ become the key-forwarder of $m$
```

D. The forwarding algorithm

The complete MOUF algorithm is listed in Algorithms 1 and 2. The algorithm for an ordinary forwarder is simply to forward a message when it encounters the destination of the message.

For the key-forwarder k of a message m, the current expected utility $\hat{u}$ and the expected utility in the case that the message is forwarded $\hat{u}^{new}$ are first calculated. Specifically, $\hat{u} = b - \Delta \cdot (t' + \hat{d}) - c$, where b is the initial benefit of m, t' is the time elapsed since m was created, $\hat{d}$ is the estimated joint delay of all forwarders of m starting from the current time, and c is the cumulative charges for m; $\hat{u}^{new} = b - \Delta \cdot (t' + \frac{1}{\frac{1}{\hat{d}} + \frac{1}{I_{i,d}}}) - (c + c_i)$, where $I_{i,d}$ is the inter-contact time between node i and destination d of m, and c_i is the charge of node i.

If forwarding m to i results in a positive utility gain $\hat{u}^{new} - \hat{u}$, node k will forward the message to i, and the value of $\hat{d}$ and c of m will be updated accordingly. If a copy of m is forwarded to node i, and node i has a smaller inter-contact time with destination d than the current key-forwarder k, then node i becomes the new key-forwarder of m.

IV. ANALYSIS

In this section, we conduct an analytical study to compare the proposed algorithm, MOUF, with several representative forwarding algorithms. We will first review the compared forwarding algorithms. Then, we derive the expected utility achieved by these algorithms.

A. The compared forwarding algorithms

We compare the following forwarding algorithms in both our analysis is this section and the simulations in the next section, among which *Spray-and-Wait* and *Delegation* are non-utility-based opportunistic forwarding algorithms, and *Delegation* has the state-of-the-art forwarding performance in terms of delivery rate.

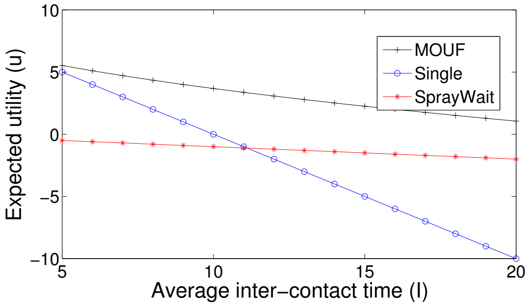


Fig. 2. The analytical results of the expected utility.

Spray-and-Wait [8] controls the number of copies per message in the network to be smaller than L. Spray-and-Wait is an oblivious forwarding strategy. Any non-oblivious opportunistic forwarding algorithm that optimizes a particular forwarding metric should outperform Spray-and-Wait in terms of that metric.

Delegation forwarding [16] can harness a wide range of forwarding metrics. We use the inter-contact time of each node with the destination of the message as an estimation of the expected delay, and use it as the forwarding metrics for *Delegation* in our simulation. A threshold is maintained in each message, which equals to the smallest among the forwarders' inter-contact times with the destination. A message will only be forwarded to a node whose inter-contact time with the destination is smaller than this threshold.

Single is a single-copy utility-based opportunistic forwarding algorithm. It uses the same utility definition $u = b-\Delta\cdot t-c$, as in Equation 5. *Single* forwards the message if the utility can be increased. Being a single-copy forwarding algorithm, *Single* deletes the copy in the forwarder immediately after each forwarding.

MOUF is our algorithm as presented in Section III.

B. Comparison on expected utility

In this sub-section, within a simplified network environment, we analyze the proposed algorithm MOUF and show that it significantly reduces expected utility compared to non-utility-based algorithms. The assumptions about the network made in this section is listed as follows: The network has a large number N of nodes. All nodes charge the same forwarding cost $\bar{c}$. All messages have the same initial benefit b, the same time-to-live T, and the same benefit decay Δ. Finally, inter-contact times are independent and identically distributed with mean $\bar{I}$.

Let k be the number of copies of a message in the network. According to Equation 3, the utility of the message can be given by

$$\hat{u} = b - \Delta \cdot \hat{t} - c = b - \Delta \cdot \frac{\bar{I}}{k} - k \cdot \bar{c}. \tag{7}$$

In the equation above, we assume that the time to spread the message to k forwarders is neglectable, since the network has a large number N ($N \gg k$) of nodes. The expected time for one of these k forwarders to encounter the destination $\hat{t} = \frac{\bar{I}}{k}$ due to the assumption that inter-contact times are independent and identically distributed with mean $\bar{I}$. The total forwarding charges $c = k \cdot \bar{c}$ since all forwarders charge the same $\bar{c}$.

MOUF's expected utility. As its objective, MOUF maximizes the expect utility. Therefore, it will implicitly select an optimal number of forwarders k^{opt}, which can be derived as follows. Let the gradient of $f(k) = b - \Delta \cdot \frac{\bar{I}}{k} - k \cdot \bar{c}$ with respect to k be 0. We have: $\frac{\Delta \cdot \bar{I}}{k^2} - \bar{c} = 0$, and thus

$$k^{opt} = \sqrt{\frac{\Delta \cdot \bar{I}}{\bar{c}}}. \tag{8}$$

Combining Equations 7 and 8, we have

$$\hat{u}^{MOUF} = b - 2 \cdot \sqrt{\Delta \cdot \bar{I} \cdot \bar{c}}. \tag{9}$$

Single's expected utility. Single is a single-copy utility-based forwarding algorithm. The only difference between Single and MOUF is that there is no ordinary forwarder in Single and the only copy of each message is in the custody of its key-forwarder. Therefore, $\hat{t} = \bar{I}$, and we have

$$\hat{u}^{Single} = b - \Delta \cdot \bar{I} - k \cdot \bar{c}. \tag{10}$$

The objective of Single is also to maximize the expect utility. It is obviously that the utility is maximized when setting $k^{opt} = 0$, and thus

$$\hat{u}^{Single} = b - \Delta \cdot \bar{I}. \tag{11}$$

Since under the simplified assumption in this section, every forwarder has the same expected delay, the optimal forwarding strategy of a single-copy utility-based forwarding algorithm is not to forward any copy for any message so that the total forwarding charge is 0.

Spray-and-Wait's expected utility. Spray-and-Wait forwards copies to the first $k = L$ nodes encountered by existing forwarders. Therefore,

$$\hat{u}^{SprayWait} = b - \Delta \cdot \frac{\bar{I}}{L} - L \cdot \bar{c}. \tag{12}$$

In Spray-and-Wait, if there are excessive copies, such that $L \geq \frac{b}{\bar{c}}$, the expected utility will be negative. In other words, if b is selected regardless of L, the expected utility of Spray-and-Wait can be arbitrarily small.

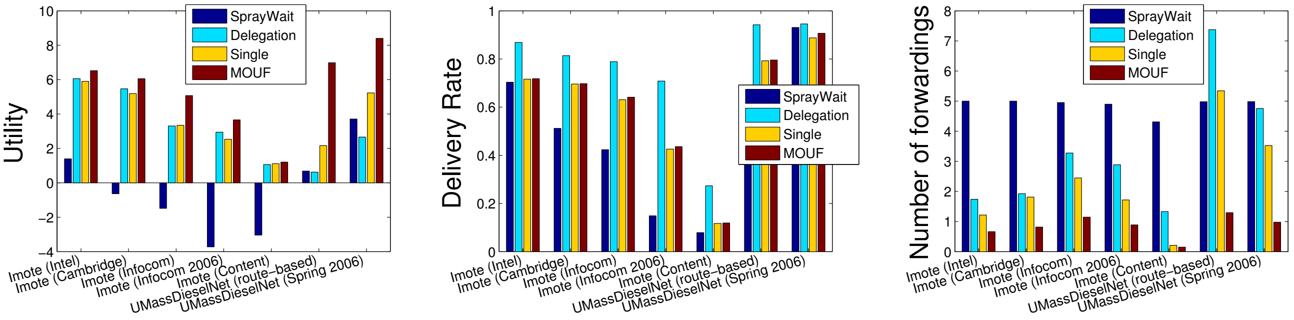
Example plot. To illustrate the analytical expected utilities of the above algorithms, we plot the analytical results using $\Delta = \bar{c} = 1$, $b = 10$, $L = 6$ and $\bar{I}$ ranges between 5 and 20. As showed in Figure 2, we can see that MOUF constantly has higher expected utility than the compared algorithms. *Single* has a low expected utility when $\bar{I}$ is large, since in this case a multi-copy scheme is required to decrease the average delay.

V. SIMULATION

In this section, we conduct extensive simulations to evaluate the proposed algorithm, MOUF, compared with several representative forwarding algorithms as described in the Section IV. We will first present the real traces that we use to drive our simulation, followed by our simulation settings. Lastly, we will show and discuss our simulation results.

TABLE I
THE STATISTICS OF THE TRACE DATA.

#	Trace	Contact	Length (d.h:m.s)	Forwarding nodes	Dest nodes
1	Imote (Intel)	2766	4.3:48.32	9	128
2	Imote (Cambridge)	6732	6.1:36.3	12	223
3	Imote (Infocom)	28216	3.4:38.29	41	264
4	Imote(Infocom2006)	227657	3.21:43.39	98	4519
5	Imote(Content)	41587	23.19:50.18	54	11418
6	UMassDieselNet (route-based)	43874	54.23:27.14	92	92
7	UMassDieselNet (Spring 2006)	43874	54.23:27.14	31	32



(a) Utility (b) Delivery Rate (c) Number of forwardings

Fig. 3. In this set of simulations, messages are assigned with random initial benefits and benefit decays, with both initial benefit rate and benefit decay range from 2 to 20. Results for four simulation metrics in all of the seven traces are showed.

A. The real trace sets used in simulations

We conduct simulations in a total of seven real traces. These traces are two UMassDieselNet traces [5] and five traces in the Cambridge Haggle trace set [14]. Table I provides the information about the number of contacts, the duration, the forwarding nodes, and the destination nodes of the traces.

UMassDieselNet trace In the UMassDieselNet bus system consisting of 40 buses [5], 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 multiple routes. There are 92 shifts serving these routes. Because the buses are often handed over to another driver to operate the next sub-shift on a different route, the all-bus-pair contacts in the original trace are almost random. We also translate 55 days of bus-to-bus contacts into contacts between sub-shifts. The resulting trace is showed as *UMassDieselNet (route-based)* in our simulation results, while the original trace is showed as *UMassDieselNet (Spring 2006)*.

Cambridge Haggle (Imote) trace set The Cambridge Haggle trace [14] data includes a total of five traces of Bluetooth device connections by people carrying mobile devices (iMotes) for a number of days. These traces are collected by different groups of people in office environments, conference environments, and city environments, respectively. Bluetooth contacts are classified into two groups: iMotes' sightings of other iMotes are classified as *internal contacts*, while sightings of other types of Bluetooth devices are called *external contacts*. Since there is no record of contact between non-iMotes, we only use the iMotes as *routing nodes*. Other nodes, or *external nodes*, can only be assigned as destinations. In our simulation results, these traces are showed as *Imote (Intel)*, *Imote (Cambridge)*, *Imote (Infocom)*, *Imote (Infocom2006)*, and *Imote (Content)*.

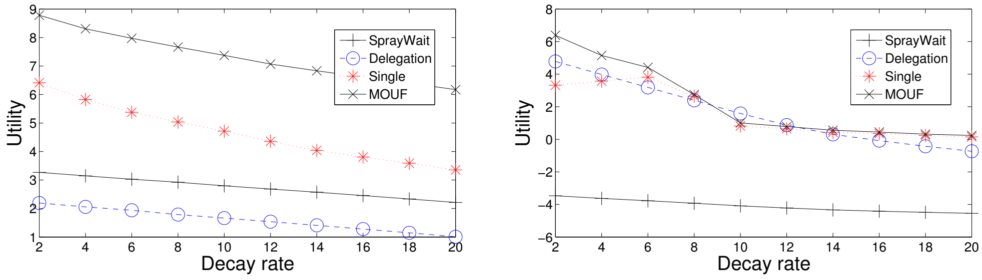
B. Simulation settings

Simulation variables In every simulation, we set the forwarding charge of each node $c_i = \bar{c} \cdot \frac{f_i}{\bar{f}}$ as discussed in Section II-B, where f_i is the overall contact frequency of node i, $\bar{f}$ is the average of f_i over all nodes in the network, and $\bar{c}$ is the average forwarding cost. We set $\bar{c} = 1$ in all our simulations.

The simulation variables are the initial benefit rate $\frac{b}{\bar{c}}$ and the benefit decay rate $\frac{\Delta \cdot T_{total}}{\bar{c}}$, where T_{total} is the simulation length in each simulation, as listed in Table I. In different simulations, both the initial benefit rate and the benefit decay rate range from 2 to 20 with an interval of 2.

Simulation metrics We evaluate and compare algorithms using three metrics as follows: (1) *Utility* measures the utility of each message according to Equations 3 and 4. (2) *Delivery rate* measures the percentage of messages successfully delivered. Although it is not the optimization objective in this paper, we include the results to observe its possible relation with utility. (3) *Number of forwardings* measures the average of the actual forwarding cost per message.

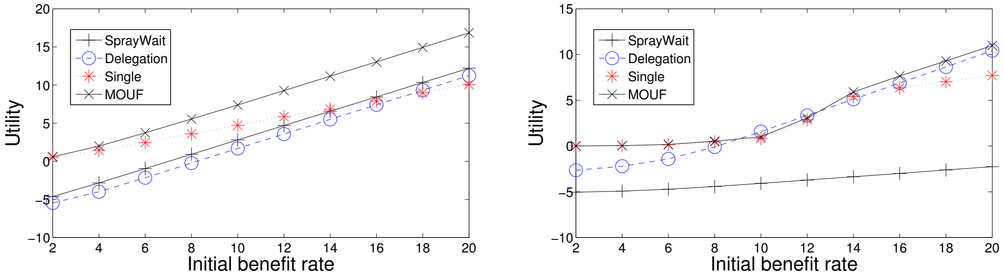
Four sets of simulations Different sets of simulations are conducted, where the simulation variables are of different ranges, and the simulation results are illustrated in different ways to facilitate the analysis of the results. Due to space limitation, only representative simulation results in each set of simulation are showed.



(a) Utility in UMassDieselNet (Spring 2006)

(b) Utility in Imote (Infocom 2006)

Fig. 4. Results on utility in UMassDieselNet (Spring 2006) and Imote (Infocom 2006). In this set of simulations, the initial benefit rate ranges from 2 to 20, while the benefit decay rate is fixed at 10.



(a) Utility in UMassDieselNet (Spring 2006)

(b) Utility in Imote (Infocom 2006)

Fig. 5. Results on utility in UMassDieselNet (Spring 2006) and Imote (Infocom 2006). In this set of simulations, the benefit decay rate ranges from 2 to 20, while the initial benefit rate is fixed at 10.

In the first set of simulations, messages are assigned with random initial benefits and benefit decays, where both initial benefit rate and benefit decay range from 2 to 20. All of the five simulation metrics are displayed. In the second set of simulations, the initial benefit rate ranges from 2 to 20, while the benefit decay rate is fixed at 10. In this set of simulations, the benefit decay rate ranges from 2 to 20, while the initial benefit rate is fixed at 10. In the fourth set of simulations, both initial benefit rate and the benefit decay rate ranges from 2 to 20, and only the performance of MOUF is displayed. For the second, the third, and the fourth sets of simulations, we only show the results on utility.

Each simulation result is the average over 30 rounds of simulations with the same settings. In each round of simulations, a total of 5,000 messages are generated randomly with respected to their sources and destinations.

C. Simulation results and discussion

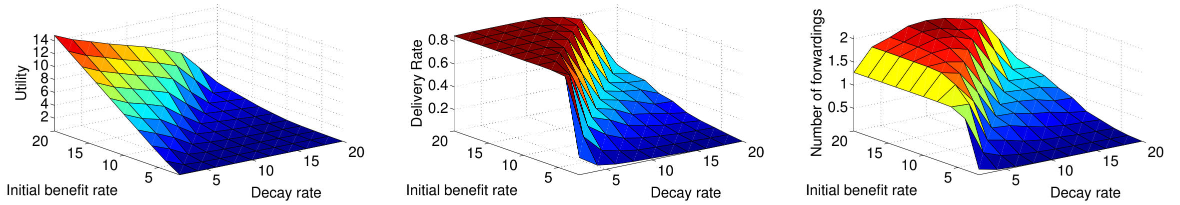
Figures 3(a), 3(b), and 3(c) show the simulation results under the three simulation metrics in all of the seven traces. Figure 3(a) shows that MOUF has the highest utility in all traces. Compared with Delegation, the higher utility of MOUF is a result of its slightly lower delivery rate and it much lower number of fowardings.

Spray-and-Wait has very low utility in traces #3 to #5, where it also has a very low delivery rate. Delegation has the lowest utility in traces #6 and #7 although it has the highest delivery rate in these traces. It is interesting to see in trace #7 that Delegation has a higher delivery rate and a lower number of forwardings than Spray-and-Wait, but Delegation has a lower utility. We conclude that most pairs of nodes in traces #6 and #7 have contact opportunities, and the early dissemination of copies in Spray-and-Wait results in a smaller delay. On the other hand, most pairs of nodes do not have contact opportunities in traces #3 to #5, and the blind forwarding in Spray-and-Wait results in a low delivery rate.

Figures 6 displays how initial benefit and benefit decay affect utility and load-balancing. As showed in Figure 6, in trace #4, which belongs to a large network, both initial benefit and benefit decay have clear effects on utility. Delivery rate drops sharply when the expected utility becomes negative, as the initial benefit decreases and the benefit decay increases. Number of forwardings is encouraged by large benefit decay and initial benefit. The weak effect of benefit decay on utility and the bad load balancing in Figures 6(e-h) suggest the existence of backbone nodes in the network of trace #7.

D. Summary of evaluation

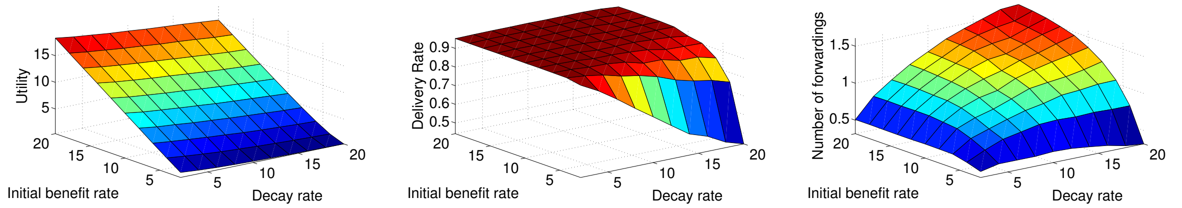
Simulation results show that the proposed algorithm MOUF has the highest utility in all simulations. While some compared algorithm optimizes a certain simple forwarding metrics, such as delivery rate, it can have bad performance in terms of some user-preferred utility. On the other hand, single-copy utility-based forwarding algorithm can have worse performance than non-utility-based multi-copy algorithms in networks with uncertain mobility and small average inter-meeting times. These results proof the necessity of the proposal of a utility-based multi-copy forwarding algorithm.



(a) Utility in Imote (Infocom 2006)

(b) Delivery Rate in Imote (Infocom 2006)

(c) Number of forwardings in Imote (Infocom 2006)



(d) Utility in UMassDieselNet (Spring 2006)

(e) Delivery Rate in UMassDieselNet (Spring 2006)

(f) Number of forwardings in UMassDieselNet (Spring 2006)

Fig. 6. Results on utility for MOUF in the Imote (Infocom 2006) trace and the UMassDieselNet (Spring 2006) trace. In these two sets of simulations, both the benefit decay rate and the initial benefit rate range from 2 to 20.

VI. RELATED WORK

Opportunistic forwarding is usually applied in networks where node mobility is nondeterministic, and where the bandwidth for message forwarding is limited so that flooding-based algorithms [7] are not affordable. Opportunistic forwarding algorithms make thrifty forwarding decisions by selecting nodes of high delivery potential according to different forwarding metrics. They include delivery probability [17], encounter frequency [17], time elapsed since last encounter [5]. The differences between the above forwarding metrics and the proposed utility metric are: (1) they are not compound metrics, and (2) they cannot improve load-balancing.

Several utility-based routing algorithms [11], [12] are defined with the objective of minimizing transmission cost in a multi-hop single-path routing algorithm in unreliable wireless networks. The most significant difference of our utility-based forwarding algorithm is that our algorithm is a multi-copy forwarding algorithm suitable to the uncertain node mobility that characterizes many DTNs.

VII. CONCLUSION

We proposed the first utility-based message forwarding algorithm under the multi-copy, opportunistic forwarding paradigm in DTNs. Evaluation results show that the proposed MOUF algorithm has a significant performance gain over the compared forwarding algorithms in terms of utility.

Future work includes (1) an information propagation method that allows each copy in MOUF to have a more accurate local estimation on the overall expected delay and forwarding cost, (2) a pricing strategy for the nodes to determine their forwarding charges to optimize other objectives, such as load-balancing, and (3) variations of MOUF under different utility definitions and different kinds of forwarding information, such as social-similarity.

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] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *Proc. of ACM SIGCOMM*, 2004.

[3] M. Motani, V. Srinivasan, and P. Nuggenhalli. PeopleNet: Engineering a wireless virtual social network. In *Proc. of ACM MobiCom*, 2005.

[4] A. Chaintreauy, P. Hu, J. Crowcroft, C. Dioty, R. Gassy, and J. Scotty. Pocket Switched Networks: Real-world mobility and its consequences for opportunistic forwarding. In *Tech. Rep. UCAM-CL-TR-617*, 2005.

[5] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networking. In *Proc. of IEEE INFOCOM*, 2006.

[6] Y. Gu and T. He. Data Forwarding in Extremely Low Duty-cycle Sensor Networks with Unreliable Communication Links. In *Proc. of ACM SenSys*, 2007.

[7] A. Vahdat and D. Becker. Epidemic Routing for Partially-connected Ad Hoc Networks. In *Technical Report, Duke University*, 2002.

[8] 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.

[9] D. Gunawardena, T. Karagiannis, A. Proutiere, E. Santos-Neto, and M. Vojnovic. Scoop: Decentralized and Opportunistic Multicasting of Information Streams. In *Proc. of ACM MobiCom*, 2012.

[10] C. Liu and J. Wu. An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2009.

[11] M. Lu and J. Wu. Social Welfare Based Routing in Ad Hoc Networks. In *Proc. of ICPP*, 2006.

[12] J. Wu, M. Lu, and F. Li. Utility-based Opportunistic Routing in Multi-hop Wireless Networks. In *Proc. of IEEE ICDCS*, 2008.

[13] M. Xiao, J. Wu, C. Liu, and L. Huang. TOUR: Time-sensitive Opportunistic Utility-based Routing in Delay Tolerant Networks. In *Proc. of IEEE INFOCOM*, 2013.

[14] 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.

[15] D. Gunawardena T. Karagiannis, A. Proutiere, E. Santos-Neto, and M. Vojnovic. Scoop: Decentralized and Opportunistic Multicasting of Information Streams. In *Proc. of ACM MobiCom*, 2011.

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

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