Minimizing Average Spraying Cost for Routing in Delay Tolerant Networks

Eyuphan Bulut, Zijian Wang and Boleslaw K. Szymanski
Department of Computer Science and Center for Pervasive Computing and Networking
Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180, USA
{bulute, wangz, szymansk}@cs.rpi.edu

***Abstract*—In this paper, we study cost efficient multi-copy spraying algorithm for routing in Delay Tolerant Networks (DTN) in which source-to-destination path does not exist most of the time. We present a novel idea and the corresponding algorithm for achieving the average minimum cost of packet transmission while maintaining the desired delivery rate by the given deadline. The number of message copies in the network depends on the urgency of meeting the delivery deadline for that message. We find the efficient copying strategy analytically and validate the analytical results with simulations. The results demonstrate that our time dependent spraying algorithm achieves lower cost of message copying than the original spraying algorithm while maintaining the desired delivery rate by the deadline.**

I. Introduction

Delay Tolerant Networks (DTNs) are wireless networks in which at any given time instance the probability that there is an end-to-end path from a source to destination is low. There are many examples of such networks in real life including wildlife tracking sensor networks [1], military networks [2] and vehicular ad hoc networks [4]. Since the standard routing algorithms assume that the network is connected most of the time, they fail in routing packets in DTNs.

Routing algorithms for DTNs need to carefully consider the disconnectivity of the network. Hence, in recent years, new algorithms using buffering and contact time schedules have been proposed. Since most of the nodes in a DTN are mobile, the connectivity of the network is maintained only when nodes come into the transmission ranges of each other. If a node has a message copy but it is not connected to another node, it stores the message until an appropriate communication opportunity arises. The important considerations in such a design are (i) the number of copies that are distributed to the network for each message, and (ii) the selection of nodes to which the message is replicated.

In this paper, we study how to distribute the copies of a message among the potential relay nodes in such a way that the predefined percentage of all messages meets the given delivery deadline with the minimum number of copies used. Unlike the previous algorithms, we propose a time dependent copying scheme which basically considers the time remaining to the given delivery deadline.

The idea of our scheme is as follows. We first spray some number of copies smaller than the necessary to guarantee that the predefined percentage of all messages meets the given delivery deadline of the message to the destination. If the delivery does not happen for some time, then we spray some additional copies of the message to increase the probability of its delivery. Consequently, we will benefit from early delivery with fewer number of copies, if it happens, so the average number of copies used by each message is minimized.

The remaining of the paper is organized as follows. In Section II we present the previous work done on this topic and discuss some basic mobility assisted routing concepts. We also comment about the differences between our algorithm and the others. In Section III we describe our algorithm in detail and provide its analysis for its different variants. In Section IV, we present evaluation of the performance of the proposed scheme using simulations and demonstrate the achieved improvements. We also compare the results of our analysis with the simulation results. Finally, we offer conclusion and outline the future work in Section V.

II. Related Work

Routing algorithms for delay tolerant networks are generally classified as either replication based or coding based [13]. In replication based algorithms, multiple or a single copy of the message is generated and distributed to other nodes (often referred to as relays) in the network. Then, any of these nodes, independently of others, try to deliver the message copy to the destination. In coding based algorithms, a message is converted into a large set of code blocks such that any sufficiently large subset of these blocks can be used to reconstruct the original message. As a result, a constant overhead is maintained and the network is made more robust against the packet drops when the congestion arises. However, these algorithms introduce an overhead of an extra work needed for coding, forwarding and reconstructing code blocks.

Epidemic Routing [3] is an approach used by the replication based routing algorithms. Basically, in each contact between any two nodes, the nodes exchange their data so that they both have the same copies. As a result, the fastest spread of copies is achieved yielding the optimum delivery time. However, the main problem of this approach is the overhead incurred in bandwidth usage, buffer space required and energy consumed by the greedy copying and storing of messages. Hence, this approach is inappropriate for resource constrained networks. To address this weakness of epidemic routing, the algorithms with controlled replication or spraying have been proposed [5], [6], [7], [14]. In these algorithms, only a small number of copies are distributed to other nodes and each copy is delivered to the destination independently of others. Of course, such approach limits the aforementioned overhead and uses the resources efficiently.

The replication based schemes with controlled replication differ from each other in terms of assumptions about the network. Some of them assume that the trajectories of the mobile devices are known while some others assume that the times and durations of contacts between nodes are known. There are also some algorithms which assume zero knowledge about the network. The algorithms which fall in this last category seem to be the most relevant to applications because in most of the examples of delay tolerant networks from real life, neither the contact times nor the trajectories are known for certain. Consider the difficulty of acquiring such information in a wild life tracking application where the nodes are attached to animals that move unpredictably.

The algorithms which assume zero knowledge about the network include [9], MaxProb [12], SCAR [11] and Spray and Wait [8]. In each of these algorithms limited number of copies are used to deliver a message. Yet, the process of choosing the nodes for placing new replications is different in each of them. In [9] and MaxProb each node carries its delivery probability which is updated in each contact with other nodes. If a node with a message copy meets another node that does not have the copy, it replicates the message to the contact node only if that node's delivery probability is higher than its own. A similar idea is used in SCAR. Each node maintains a utility function which defines the carrier quality in terms of reaching the destination. Then, each node tries to deliver its data in bundles to a number of neighboring nodes which have the highest carrier quality.

In [8] Spyropoulos et al. propose two different algorithms called Source Spray and Wait, and Binary Spray and Wait, respectively. While in the former, only the source is capable of spraying copies to other nodes, in the latter all nodes having the copy of the message are also allowed to do so. In Binary Spray and Wait, when a node copies a message to another node, it also gives the right of copying the half of its remaining copy count to that node. This results in distributed and faster spraying compared to the source spraying, but once the spraying is done, the expected delivery delay is the same. The authors provide the expected delay of message delivery in these two algorithms in [14].

Although there are many algorithms utilizing the controlled flooding approach, the idea of copying depending on the urgency of meeting the delivery deadline for a message has not been used by any of them. To the best of our knowledge this idea is new and it helps to decrease the average number of copies generated in the network. We will describe the details of this idea in the next section.

While designing a routing algorithm for mobile network, an important issue that must be considered is the model of mobility of nodes in the network. Random direction, random walk and random waypoint mobility models are the most popular ones among those used by the previous routing algorithms in this field. Among these models, random direction model is considered more realistic than the others.

In a network with mobile nodes moving according to a mobility model, two concepts are introduced; expected hitting time (ET) and expected meeting time (EM). While ET of a node is defined as the expected time interval of being in contact (in the transmission range) with a stable node (most often the sink), EM of a node is defined as the expected time interval of being in contact with a nearby mobile node. These two parameters are specific to each mobility model and can be derived when the network parameters are known [10]. Then, the hitting time and the meeting time of a sample node in such a network are assumed to be exponentially distributed with mean ET and EM, respectively.

III. Time Dependent Spraying

In this section, we start with listing the assumptions of our model and then we provide the details of our routing scheme and its analysis.

We assume that there are M nodes randomly walking on a $\sqrt{N}$ x $\sqrt{N}$ 2D torus according to the random direction mobility model. Each node has a transmission range R and all nodes are identical. The buffer space in a node is assumed to be infinite (not crucial since we use fewer copies), and the communication between nodes is assumed to be perfectly separable, that is, any communicating pair of nodes do not interfere with any other simultaneous communication. To be consistent with previous research, by L we denote the number of copies distributed to the network.

Given the mobility model, the expected time of delivery in Spray and Wait algorithm is equal to [14]:

$$\sum_{i=1}^{L-1} \frac{EM}{M-i} + \frac{M-L}{M-1} EW$$

This formula assumes that in the first $L-1$ contacts, the source node does not meet with the sink node and thus a wait phase is needed (probability of this happening is $\frac{M-L}{M-1}$). Here, EW is the expected duration of wait phase which is actually exponentially distributed with mean $\frac{EM}{L}$. Note that, when $M >> L$ (which we enforce to be satisfied by limiting permissible values of L), duration of spraying phase is much shorter than the duration of waiting phase, so that we can assume that the expected time of delivery in Spray and Wait algorithm is exponentially distributed with mean $\frac{EM}{L}$.

Figure 1 shows the cumulative distribution function of the expected delay of Spray and Wait algorithm for different L values. Clearly, when L increases, mean value $(1/\lambda)$ decreases and the expected delay shrinks.

In order to meet the given deadline with desired delivery rate, the simplest way is to find the minimum L that achieves this goal and spray that many message copies to the network at the beginning. However, as the main contribution of our work shows, it is possible to control the number of copies sprayed to other nodes by the urgency of meeting the predefined delivery delay. More precisely, the algorithm starts with spraying the

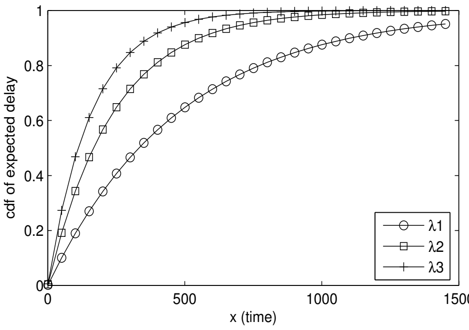

Fig. 1. The cumulative distribution function of probability of meeting the expected delay in Spray and Wait algorithm for different λ values, where $\lambda_1 > \lambda_2 > \lambda_3$

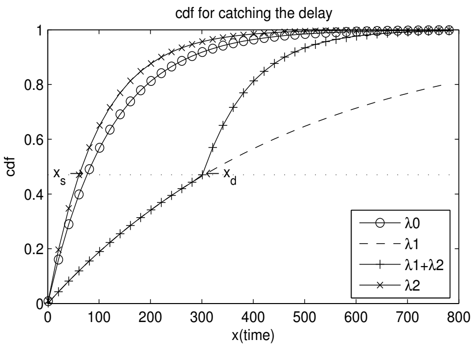

Fig. 2. The cumulative distribution function of delivery time of a message with spraying different number of copies in two different periods.

message copies to fewer nodes than the minimum L needed and then waits to see if the message is delivered for a certain period of time. When delivery does not happen, the algorithm increases the number of copies sprayed and again waits for delivery. This process repeats until the message is delivered or the time limit for delivery is reached. Hence, as the time remaining to the delivery deadline decreases and delivery has not happened[1], the number of nodes carrying the message copy increases. To the best of our knowledge, this idea has not been used by any of the previously published algorithms for DTN routing.

Consider the Figure 2. It summarizes what our algorithm wants to achieve. In this specific version of the algorithm, we allow two different spraying phases. The first one is done at the beginning and the second one is done at time x_d. The main objective of the algorithm is to attempt delivery with small number of copies and use the large number of copies only when this attempt is unsuccessful. With proper setting, the average number of copies sprayed in the network till delivery will be lower than in case of spraying all messages at the beginning.

To analyze the performance of our algorithm analytically, we need to derive two formulas, one for the average number of copies used by the algorithm, and the second one for the cumulative distribution of the probability of meeting the delivery deadline with mixed number of copies (and therefore mixed λ values). The goal is to achieve the same percentage of the messages delivered in the given delivery deadline using fewer copies on average than the standard Spray and Wait algorithm uses.

In our scheme, the term *period* refers to the time duration from the beginning of one spraying phase to the beginning of the next spraying phase. There may be multiple spray phases and the corresponding periods between them, each of different length. We start with the analysis of the two period case to find out the optimal period length and the corresponding copy counts for each period.

1) Partitioning into Two Periods: If there are two periods until the delivery deadline of a message, the arising questions are how the time should be divided into the two periods and how many copies should be allowed in each. In other words, what should be the value of x_d in Figure 2 to minimize the average number of copies used by the algorithm?

Let's assume that the standard Spray and Wait algorithm uses L copies (including the copy in the source node) of a message to achieve the probability $p_d \approx 1$ of delivery of the message by the deadline t_d. Let's further assume that the Two Period Delayed Spraying algorithm sprays L_1 copies to the network at the beginning of execution and additional $L_2 - L_1$ copies at time x_d, the beginning of the second period. Then, the cumulative distribution function of the probability of delivering the message at or below time x is:

$$cdf(x) = \begin{cases} 1 - e^{-\alpha L_1 x} & \text{if } x \leq x_d \\ 1 - e^{-\alpha L_2 (x - x_s)} & \text{if } x > x_d \end{cases}$$

where, $\alpha = 1/EM$ is an inverse of the expected meeting time of the nodes and x_s is the time interval by which the second exponential function of the above formula is delayed compared to the first one and defined by equality of the both functions at point x_d, hence:

$$\begin{aligned} 1 - e^{-\alpha L_1 x_d} &= 1 - e^{-\alpha L_2 (x_d - x_s)} \\ x_s &= x_d \frac{L_2 - L_1}{L_2} \end{aligned}$$

The expected delivery ratio when L copies are used in the standard Spray and Wait algorithm is by definition $p_d = 1 - e^{-\alpha L t_d} \approx 1$. Our objective is to match the same delivery rate by decreasing the average number of copies below L, the number of copies used in the Spray and Wait algorithm. Hence, at the expected time of delivery, t_d, the following inequality must be satisfied:

$$1 - e^{-\alpha L_2 (t_d - x_s)} \geq 1 - e^{-\alpha L t_d}$$
$$L_2(t_d - x_d + x_d L_1 / L_2) \geq L t_d$$

We can use this inequality to bound x_d as $x_d \leq t_d \frac{L_2 - L}{L_2 - L_1}$. It is clear that to minimize the average copy count in the two period case with the given L_1 and L_2 values, x_d should be as

[1]We explain how the delivery of messages are acknowledged to other nodes at the end of Section III.

Algorithm 1 FindOptimalsInTwoPeriods(L)

```
minCost = L
for each 0<L_1<L do
   L_{2_{Bound}} = FindBound(L_1)
   for each L<L_2<L_{2_{Bound}} do
      if c_2(L_1, L_2)<minCost then
         minCost = c_2(L_1, L_2)
         [L_{opt_1},L_{opt_2}] = [L_1,L_2]
      end if
   end for
end for
return [L_{opt_1},L_{opt_2}]
```

large as possible, hence:

$$x_d = t_d \frac{L_2 - L}{L_2 - L_1}$$

We want to minimize the average number of copies, $c_2(L_1, L_2)$ defined as:

$$\begin{aligned} c_2(L_1, L_2) &= L_1(1 - e^{-\alpha L_1 x_d}) \\ &\quad + L_2(e^{-\alpha L_1 x_d}) \\ &= L_1 + (L_2 - L_1)e^{-\alpha L_1 x_d} \end{aligned}$$

Note that if there is a failure to deliver the message in the first round, then cost becomes L_2 copies. Substituting x_d in the above, we get:

$$c_2(L_1, L_2) = L_1 + (L_2 - L_1)e^{-\alpha L_1 t_d \frac{L_2 - L}{L_2 - L_1}}$$

Taking derivative of c_2 in regard of L_2, and comparing it to zero, we obtain:

$$L_2 = L1 + \alpha L_1 t_d (L - L_1)$$

so $L_2 - L_1 = \alpha L_1 t_d (L - L_1)$ and therefore

$$c_2^*(L_1) = L_1[1 + \alpha t_d (L - L_1) e^{-\alpha L_1 t_d + 1}].$$

Taking the derivative of the above function we can obtain the complicated formula on the optimal value of L_1^* as a function of L and t_d, and then taking the floor and ceiling compute the corresponding optimal values of L_2^* and again their floors and ceilings can be used to arrive at the result.

We can also find out the optimal values of L_1 and L_2 with a simpler method. Since $L_1 < L$ (otherwise there is no saving) and possible values for L_1 are integers, we can use enumeration as explained in Algorithm 1 and obtain the optimal values with a limited cost.

Note that, for a given L_1, there is an upper bound for L_2 ($L_{2_{Bound}}$) after which the value of $c_2(L_1, L_2)$ does not provide lower value than L. The value of $L_{2_{Bound}}$ is calculated as:

$$\begin{aligned} x_d &< t_d \text{ (otherwise } L_1 = L) \\ L_1 + (L_2 - L_1)e^{-\alpha L_1 t_d} &< c_2 < L \text{ (otherwise no saving)} \\ L_2 &< L_1 + (L - L_1)e^{\alpha L_1 t_d} = L_{2_{Bound}} \end{aligned}$$

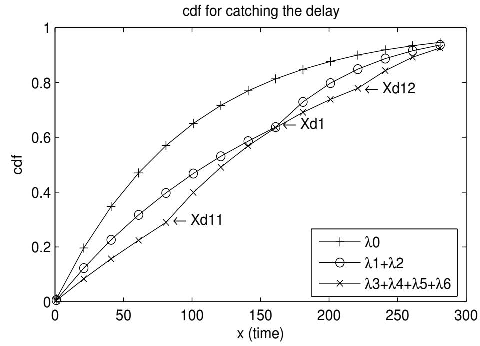

Fig. 3. The same behavior for the cumulative distribution function of meeting the deadline can be obtained using recursive partitioning algorithm. In it, more periods of spraying is achieved and the total cost of spraying is further decreased.

2) More Periods with Recursive Partitioning: In this section, we show that by applying recursive partitioning of each period, more periods can be used and lower cost of spraying can be achieved. Consider the example illustrated in Figure 3. From previous section, we know how to partition the entire period into two periods. However, it is also possible to partition each of these two periods individually and decrease the cost of spraying even further. Although this may not be the optimal partitioning, it still decreases the spraying cost.

If we want to have three periods until the delivery deadline of messages, we can either partition the first period ($\lambda 1$) or the second period ($\lambda 2$) and select the one which achieves the lowest cost. In other words, we need to select either ($\lambda 3$, $\lambda 4$, $\lambda 2$) or ($\lambda 1$, $\lambda 5$, $\lambda 6$). Furthermore, after obtaining the three period spraying, we can even run the same algorithm to find a lower cost spraying with four periods. However, we need to partition each period carefully considering the boundaries of possible L_i values.

Assume that we currently have k periods of spraying. Let L_i denote the copy count after spraying in each period and x_{di} denote the end time of i^{th} period. Then, the cumulative distribution function of the probability of delivering the message by the time x becomes:

$$cdf(x) = \begin{cases} 1 - e^{-\alpha L_1 x} & [0, x_{d1}] \\ 1 - e^{-\alpha L_2 (x - x_{s_2})} & (x_{d1}, x_{d2}] \\ \ldots & \\ 1 - e^{-\alpha L_k (x - x_{s_k})} & (x_{d(k-1)}, x] \end{cases}$$

where x_{s_i} is the delay of the i^{th} exponentional function compared to the first function and can be computed as:

$$x_{s_k} = \sum_{i=1}^{k-1} x_{di} \frac{L_{i+1} - L_i}{L_k} \quad (1)$$

We can also compute the value of x_{s_k} from the value of $x_{s_{k-1}}$ as follows:

$$x_{s_k} = \frac{x_{s_{k-1}} L_{k-1} + x_{d(k-1)}(L_k - L_{k-1})}{L_k} \quad (2)$$

Algorithm 2 IncreasePartitions(k,x_d[ ],L[ ])

```
min = current copy cost with k periods
for each 1≤i≤k do
    [x'_d,L'] = PartitionIntoTwo(i,x_d[ ],L[ ])
    c = Cost(k+1,x'_d,L')
    if c<min then
        p = [x'_d,L']
    end if
end for
return p
```

We want to increase the number of periods to $k+1$ while decreasing the total cost for spraying with the same delivery rate at the delivery deadline. Algorithm 2 and Algorithm 3 summarize the steps to achieve this goal.

Basically, we partition each period into two periods one by one and find the new cost for the current partitioning. Then among these possible partitionings, we select the one that achieves the lowest cost. Here, for given L_i and x_{di} values, the average cost of spraying (average number of copies) can be found by the following formula:

$$Cost(k, x_d, L) = \sum_{i=1}^{k} (L_i - L_{i-1}) e^{-\alpha L_{i-1}(x_{d(i-1)} - x_{s_{i-1}})}$$

For each period except the last one, there is always a boundary for L_{i1} and L_{i2} due to the fact that $L_i{<}L_{i+1}$. But for the last period, at the first glance it seems that L_{i2} has no upper bound. However, since for all i's $x_{s_i}{>}0$ and $x_{di}{<}t_d$, then $x_{di}{-}x_{s_i}{<}t_d$ so that there is an upper bound for L_{k2} ($L_{k2_{Bound}}$) after which the value of $Cost(k, x_d, L)$ does not become smaller than L. The value of $L_{k2_{Bound}}$ for the new partition is calculated as:

$$\begin{aligned} L &> Cost(k+1, x_d, L) > \sum_{i=1}^{k+1} (L_i - L_{i-1}) e^{-\alpha L_{i-1} t_d} \\ L_{k2} &< L_{k1} + (L - L_1) e^{\alpha L_{k1} t_d} \\ &\quad - \sum_{i=1}^{k-2} (L_{i+1} - L_i) e^{\alpha (L_{k1} - L_i) t_d} \\ &\quad - (L_{k1} - L_{k-1}) e^{\alpha (L_{k1} - L_{k-1})} = L_{k2_{Bound}} \end{aligned}$$

In Algorithm 3, we show how the optimal partitioning of a single period is found. For each possible pair (L_{i1},L_{i2}), the cost of spraying is found and optimal pair which gives the minimum cost is selected. Here x_{mid} denotes the boundary point in which the second inner period starts (i.e, the start of period for spraying L_{i2}-L_{i1} copies). For the period of L_m, to be able to achieve the same cumulative distribution function while using fewer copies on average, the following inequality must be satisfied:

$$\begin{aligned} 1 - e^{-\alpha L_m (x_{d(m)} - x_{s_m})} &\geq 1 - e^{-\alpha L_{m2} (x_{d(m)} - x_{s_{m2}})} \\ L_m (x_{d(m)} - x_{s_m}) &\geq L_{m2} (x_{d(m)} - x_{s_{m2}}) \end{aligned}$$

Algorithm 3 PartitionIntoTwo(i,x_d[ ],L[ ])

```
f_1 = cdf(x_{i-1})
f_2 = cdf(x_i)
minCost = L_i(f_2-f_1) // current cost of period
for each L_{i-1}<L_{i1}<L_i do
    if i=k then
        L_up = L_{k2_Bound}
    else
        L_up = L_{i+1}
    end if
    for each L_{i1}<L_{i2}<L_up do
        Compute x_mid using Eq.3
        Compute x_{s_i1} using Eq.2
        internalCost = L_{i1}(f_2-f_1)+L_{i2}(f_3-f_2)
        if internalCost<minCost then
            minCost = internalCost
            x_m = x_mid
            [L_{opt_1},L_{opt_2}] = [L_{i1},L_{i2}]
        end if
    end for
end for
x'_d[ ] = [x_{d1},...,x_{d(i-1)},x_m,x_{di},...,x_k]
L'[ ] = [L_1,...,L_{i-1},L_{opt_1},L_{opt_2},L_{i+1},...,L_k]
return [x'_d,L']
```

When we calculate the values of x_{s_m} and $x_{s_{m2}}$ using Eq.1 and substitute them in the above formula, we obtain an upper bound for x_{mid}:

$$x_{mid} \leq \frac{x_{d(m)}(L_{m2} - L_m) + x_{d(m-1)}(L_m - L_{m1})}{(L_{m2} - L_{m1})} \quad (3)$$

Obviously, to minimize the average number of copies within a period with given L_{m1} and L_{m2} values, x_{mid} should be as large as possible, hence it should be assigned to this upper bound.

Note that, in all the above algorithms, since there are bounds for each L_i value, the complexity of algorithms are limited and much lower than the indicated bound.

3) Acknowledgment of Delivery: The designs of most of the routing protocols for delay tolerant networks do not explain in detail how the nodes in the network learn about the delivery of a message to the destination so spraying after the message delivery is suppressed. Yet, this is a crucial issue because it directly affects the cost of copying of messages. If a message is delivered to destination, but a specific node is not notified about the delivery, this node will continue spraying the message, increasing the average cost of copying.

In this paper, we study two types of acknowledgments for notifying the nodes that the message has been delivered.

TYPE I: When destination receives a message, it first creates an acknowledgment for that message and sends it to other nodes within its range, which is assumed to be same for all the nodes in this case. Then, using epidemic routing, this acknowledgment is spread to all other nodes whenever there is a contact between a node having the acknowledgment and a node without it. Note that, since the acknowledgment packets are much smaller than data messages, the cost of this acknowledgment epidemic routing is small compared to the cost of routing the data packets. More costly is the delay with which all nodes in the network learn about the delivery of the message. During this delay, there may be useless spraying of the message increasing the total cost of copying.

TYPE II: In this type acknowledgment, we assume that the destination uses one time broadcast over the more powerful radio than the other nodes (case often present in practice) so the broadcast reaches all the nodes in the network. Like in the previous case, the acknowledgment message is short, so its broadcast is inexpensive. However, to make the scheme more efficient, we use the following idea. If the destination receives some messages then it waits until the closest period change time (x_d) of any message. During that time, if the destination receives new messages, it also stores the ids of these messages and at the end of this time it broadcasts an acknowledgment message to all nodes including the ids of all received messages. As a result, the destination acknowledges all messages using only one broadcast of high powered radio and without letting more spray of any received message after the delivery time.

The second case results of course in better performance of delayed spraying than the first one. However, it may require higher energy consumption. In simulations we compare the performances of both ideas by showing how they affect the results of our algorithm.

IV. Simulation Results

In our simulations, we implemented the original Source Spray and Wait algorithm using a Java based visual simulator. We deployed 100 mobile nodes (including the sink) onto a torus of the size 300 m by 300 m. All nodes (except the sink that has high range of acknowledgment broadcast in TYPE II case) are assumed to be identical and their transmission range is set at R = 10 m. Nodes move according to random direction mobility model [10]. We have created messages at randomly selected source nodes for delivery to the sink node whose initial location is also decided randomly. Then, we collected some useful statistics from the network. The results are averaged over 1000 runs.

Using the Algorithm 1 we first found optimum combination of copy counts (L_1,L_2) for different L values. Then, using the Algorithm 2 and Algorithm 3 we have found optimum L_i combination when there are three periods. Table I shows the values of these optimum L_i's for different L values. In these results, we assumed TYPE II acknowledgment which provides end of spraying at the exact delivery time.

We have calculated the average number of copies in each of these optimum L_i combinations with simulations as well. Figure 4 and Figure 5 present the comparison of results when there are two and three periods, respectively. In the two period case, results are very close to each other, however in the three period case, the difference gets bigger because in our analysis we ignored the effect of spraying phase. When number of periods increases, period lengths get smaller, so the effect of spraying phase on the cumulative distribution function increases.

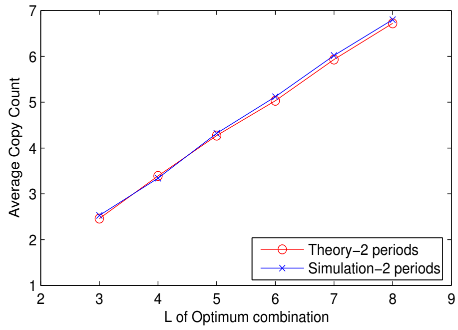

Fig. 4. The comparison of the average number of copies obtained via analysis and simulation for the two period case.

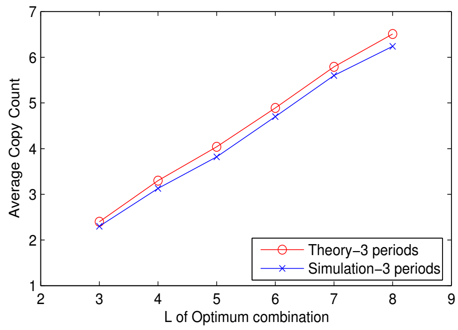

Fig. 5. The average comparison of the average number of copies obtained via analysis and simulation for the three period case.

To compare the performance of our algorithm with the original spraying algorithm, we first compare the average number of copies used in both algorithms when different types of acknowledgment mechanisms are used. Table II shows the average number of copies achieved by these two algorithms. As it is seen, with both Type I and Type II mechanisms, our algorithm uses fewer copies on average than standard spraying does. Moreover, in most of the cases, our algorithm with Type I mechanism uses fewer copies on average than the standard spraying algorithm with Type II mechanism does.

We have also measured some metrics from the simulations of both algorithms. In these simulations, we used our algorithm with two periods. Figure 6 shows the average value of delivery delay for messages. Figure 7 shows the average time of completing spraying. This value does not contain the average of cases when the message is delivered before spraying of all

L	3	4	5	6	7	8
2 periods	(2,5)	(3,6)	(3,8)	(4,9)	(5,10)	(6,12)
3 periods	(2,3,6)	(2,4,7)	(3,5,9)	(4,6,10)	(5,7,11)	(5,8,14)

TABLE I
THE OPTIMUM L_i COMBINATIONS THAT ACHIEVE THE MINIMUM AVERAGE NUMBER OF COPIES WHILE PRESERVING THE DELIVERY RATE.

L	Time-Based Spraying		Standard Spraying	
	Type I	Type II	Type I	Type II
3	2.59	2.53	2.99	2.97
4	3.65	3.43	3.96	3.90
5	4.66	4.38	4.93	4.82
6	5.61	5.23	5.91	5.70
7	6.52	6.05	6.83	6.51
8	7.40	6.81	7.76	7.36

TABLE II
AVERAGE NUMBER OF COPIES USED IN OUR ALGORITHM AND STANDARD SPRAYING ALGORITHM WHEN DIFFERENT TYPES OF ACKNOWLEDGMENTS ARE USED.

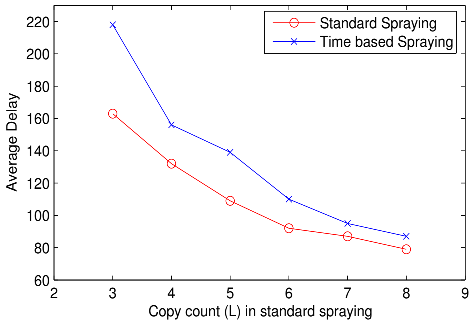

Fig. 6. The average delay comparison for standard spraying and time based spraying.

potential copies. In Figure 8, we show the success rate which is actually the percentage of all simulations that have delivery time shorter than or equal to the given deadline t_d.

When we look at these three graphs, we observe that our time based spraying algorithm incurs higher average delay but it achieves the same delivery rate before the deadline as the standard spraying algorithm. Moreover, since our scheme postpones the spraying of all copies to later times, it finishes spraying later than the standard Spray and Wait algorithm.

Finally, Figure 9 shows the improvement achieved by our algorithm in the average number of copies per message for different L values. While the two period case demonstrates about 16% benefit, the three period case shows higher improvement

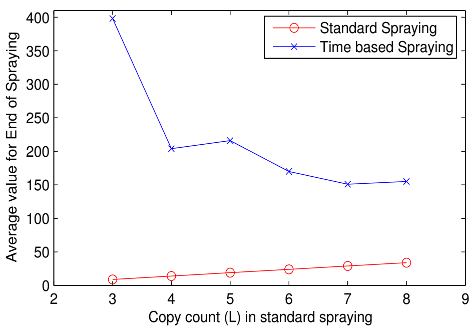

Fig. 7. A comparison of average times for end of spraying in standard spraying and time based spraying.

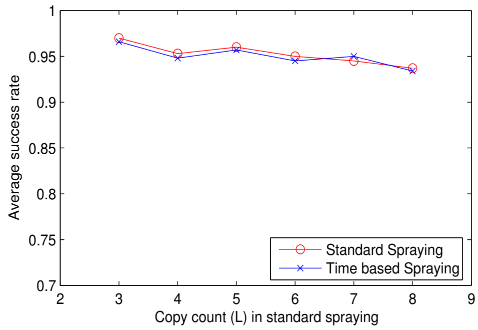

Fig. 8. The delivery rate comparison for standard spraying and time based spraying.

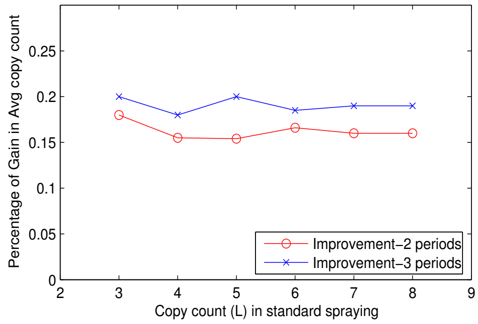

Fig. 9. The improvement obtained by the time based spraying in the number of average copies used.

of about 20%.

V. CONCLUSION AND FUTURE WORK

In this paper, we focus on the problem of routing for Delay Tolerant Networks in which the nodes are disconnected most of the time. We propose a time dependent spray and wait algorithm and evaluate its performance with simulations. We first show analytically how to partition a standard spraying algorithm into two separate periods. Then, we present a generalization to larger number of periods which reduces the cost even further. Finally, we discuss results of simulations of our algorithm that confirm that the average number of copies used by our algorithm is lower than that is used by the original spray and wait algorithm while achieving the same delivery rate at the expected time of delivery. As a future work, we plan to study different aspects of our algorithm and also aim to apply it to a real test bed such as a disconnected bus network.

ACKNOWLEDGMENT

Research was sponsored by US Army Research Laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the U.S. Government, the UK Ministry of Defence, or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, *Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet*, in Proceedings of ACM ASPLOS, 2002.
[2] *Disruption tolerant networking*, http://www.darpa.mil/ato/solicit/DTN/.
[3] A. Vahdat and D. Becker, *Epidemic routing for partially connected ad hoc networks*, Duke University, Tech. Rep. CS-200006, 2000.
[4] J. Ott and D. Kutscher, *A disconnection-tolerant transport for drive-thru internet environments*, in Proceedings of IEEE INFOCOM, 2005.
[5] A. Lindgren, A. Doria, and O. Schelen, *Probabilistic routing in intermittently connected networks*, SIGMOBILE Mobile Computing and Communication Review, vol. 7, no. 3, 2003.
[6] Y. Wang, S. Jain, M. Martonosi, and K. Fall, *Erasure coding based routing for opportunistic networks*, in Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.
[7] T. Small and Z. Haas, *Resource and peroformance tradeoffs in delat tolerant wireless networks*, in Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.
[8] T. Spyropoulos, K. Psounis,C. S. Raghavendra, *Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks*, ACM SIGCOMM Workshop, 2005.
[9] K. Harras, K. Almeroth, and E. Belding-Royer, *Delay Tolerant Mobile Networks (DTMNs): Controlled Flooding Schemes in Sparse Mobile Networks*, In IFIP Networking, Waterloo, Canada, May 2005.
[10] T. Spyropoulos, K. Psounis,C. S. Raghavendra, *Performance Analysis of Mobility-assisted Routing*, MobiHoc, 2006.
[11] C. Mascolo and M. Musolesi, *SCAR: Contextaware Adaptive Routing in Delay Tolerant Mobile Sensor Networks*, Proceedings of International conference on Wireless communications and mobile computing, 2006.
[12] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, *MaxProp: Routing for Vehicle-Based Disruption- Tolerant Networks*, In Proc. IEEE Infocom, April 2006.
[13] A. Balasubramanian, B. N. Levine, A. Venkataramani, *DTN routing as a resource allocation problem*, ACM SIGCOMM, 2007.
[14] T. Spyropoulos, K. Psounis,C. S. Raghavendra, *Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-copy Case*, IEEE/ACM Transactions on Networking, 2008.