Biologically Inspired Communications Networks, to appear
Lecture Notes in Computer Science, vol. 6745, Springer, Berlin, 2011

Bio-inspired Multi-Period Routing Algorithms in Delay Tolerant Networks

Eyuphan Bulut and Boleslaw K. Szymanski

Department of Computer Science and Center for Pervasive Systems and Networking
Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY, USA
{bulute,szymansk}@cs.rpi.edu

Abstract. In this paper, inspired by the impact of incubation period on epidemic dynamics, we present a class of routing algorithms for Delay Tolerant Networks (DTN) in which the copies or coded blocks of messages are distributed to other nodes in multiple periods. Our objective is to minimize the transmission cost (that is proportional to the number of message copies created in the process of routing), while still achieving the required delivery ratio of messages received at their destinations before their TTL's expired. We investigate two different types of routing, one based on copying of entire messages and the other on erasure coding of messages. In both cases, we use multiple periods either for message copying or for distributing encoded blocks. We present two and three-period versions of the proposed approach and we also describe its extension to multiple periods. We support the proposed model with an in-depth analysis and simulations which show the benefit of the proposed algorithms clearly.

Keywords: Delay Tolerant Network, Routing, Bio-inspired, Efficiency, Epidemics

1 Introduction

Delay tolerant networks (DTN)[1] are wireless networks in which the node connectivity is intermittent due to the movement of nodes and low node density in the network area. Moreover, it is usually not possible to find an end-to-end path from source to destination at any given time instance. Therefore, routing of messages in DTNs is more challenging than in traditional networks, where, most of the time, the node connectivity graph is stable and a path from source to destination does not change during the message delivery. Some of the examples of this kind of DTN networks deployed in real life are wildlife tracking [2], vehicular networks [3] and military networks [4].

The sporadic connectivity between nodes in DTNs necessitates the use of *store-carry-and-forward* paradigm at each node throughout the routing of messages. That is, when a node has a message but it has no connection to any other node in the network, it stores the message in its buffer and carries the message until it meets a new node that does not have this message. If the encountered node is considered useful in terms of the delivery, the message is transferred (forwarded or copied) to it. However, there are two significant issues that need to be decided: (i) to which of the encountered nodes the message should be transferred, and (ii) the maximum number of time a message should be transferred before reaching the destination. The way the routing algorithm handles these issues directly affects the average message delivery ratio, delay and the transmission cost in the network.

In this paper, we study the distribution of copies or encoded blocks of messages among the potential relay nodes. Our aim is to achieve the delivery of a required percentage of all messages by the given delivery deadline (i.e. TTL of messages) with minimum cost.

Inspired by the impact of incubation period on epidemic dynamics [5], we propose to distribute the message copies[1] in multiple periods. In an epidemic [5], after the first infectee (source node in a DTN) becomes contagious, it starts spreading the disease (in our case, a message) to others until it recovers or dies (in our context until the end of the current period of message distribution). Likewise, the new infectees first go through an incubation period during which they are not spreading the disease further. Once the incubation period passes, they become contagious and spread the disease to others.

Inspired by such epidemic dynamics, we have designed a multi-period algorithm to distribute the message copies to other nodes in the network. Our first motivation was the periodic spreading of epidemics, however, we designed the proposed routing protocol considering the necessities of DTN routing protocols. In DTNs, since the nodes meet intermittently, the messages are mostly buffered in nodes and transmitted in bundles during rarely happening meetings of nodes. Hence, the number of times the messages are transferred between the nodes of the network becomes crucial. The more frequently a message is transferred, the higher is the energy cost of the routing, and, consequently, the faster is the consumption of the node power and the shorter is the network lifetime. Message sizes matter for the same reason as well.

An important type of DTNs is sensor networks in which nodes sense the environment and collect corresponding measurements. If the collected data are first buffered at the node, and sent to the destination in bulk, the message sizes are large, so the number of times these messages are transferred before the delivery becomes vital for the total energy consumption in the network.

Consequently, considering the above reasons, with the proposed periodic distribution of message copies, we aim at minimizing the average transmission cost per message while achieving a given delivery rate by the deadline. In our scheme, the number of message copies that are distributed to other nodes depends on the remaining time before the delivery deadline, thus the urgency of meeting the delivery deadline defines the spreading rate of copies. For example, in a two period replication based routing algorithm, we first distribute the number of message

[1] Throughout the paper, when we refer to *message copies*, we mean both the message copies used in replication based routing and also the encoded blocks of the messages used in erasure coding based routing.

copies that is insufficient to guarantee the desired delivery rate by the delivery deadline. If the delivery does not happen in the first period, then in the second period, we distribute more copies, so the total number of copies distributed so far is able to achieve the desired delivery rate in the remaining time to the delivery deadline. Note that if a message is delivered in the first period, the cost of delivery is smaller than it would be if the number of copies distributed at the start of routing were sufficient to achieve the desired delivery rate. On the other hand, if the delivery has not been achieved in the first period, the copies distributed at the start of the second period will make the cost higher than what is needed in the single period case.

In this paper, we compute the average number of copies used and show that we can achieve lower average cost than the cost of distributing the sufficient number of copies from the start. Throughout the paper, we analyze different variants of the algorithm with different periods and demonstrate that cost reduction is possible.

In routing algorithms for DTNs, it is also important to define how the nodes are informed of the message delivery. In the paper, we use two different delivery acknowledgment methods. One of them uses the following biologically inspired idea. Consider an environment with different pathogens with the periods, in which different number of copies of the message are distributed, being the times when the epidemic incubates in infectees. During these times, infectees are not contagious (message copies are not distributed), epidemic does not spread (cost of delivery does not increase). However, this changes at the end of incubation period and infectees start to spread disease. To vaccinate all infectees with a vaccine for all pathogens efficiently, we can wait until the closest end of an incubation period of any infectee and apply the vaccines for all observed diseases to the entire population at that time. By delaying vaccination, we allow emergence of new diseases, possibly caused by new types of pathogens, without allowing those already infected to become contagious and infect others. As a result, we can minimize the number of necessary vaccination campaigns; each with vaccines necessary to stop already started epidemics. Inspired by the above effective vaccination campaign, we have designed the following acknowledgment scheme. When the messages start to be delivered to destination, the destination node waits until the closest period change time of any of the received messages. At that time, it broadcasts an acknowledgment of all so far received messages.

The remainder of the paper is organized as follows. In Section 2, we present background information about replication and erasure coding based routing algorithms in DTNs. There, we also give a brief overview of previous work in each category. In Section 3, we talk about the network model used and assumptions made in the proposed approach. Then in the next two sections, we give the details of multi-period message distribution based routing algorithms in two different types of routing. Section 4 presents the application of our approach to replication based routing. Section 5 presents our approach to erasure coding based routing and the analysis of the resulting routing protocol. In Section 6, we describe the message delivery acknowledgment process in DTNs and propose two different acknowledgment methods. Then, in Section 7, we evaluate the proposed algorithms 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 8.

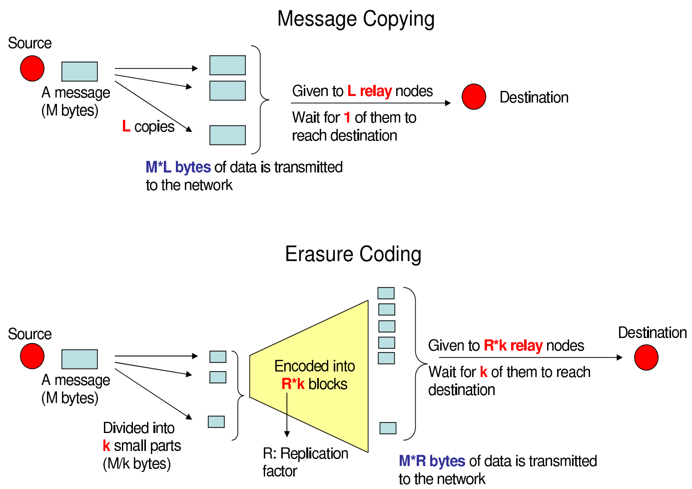

Fig. 1. Comparison of replication based and erasure coding based routing.

2 Background

In this section, we look into two different routing types in DTNs, namely, replication based routing and erasure coding based routing. For each category, first, we give the overview of the main properties of the routing type and then present a brief summary of previous DTN routing algorithms that use that routing type.

2.1 Replication (multi-copy)-based Routing

Overview: In replication based routing, a number of message copies are generated at the source node and distributed to other nodes in the network. Then any of these nodes, independently of others, tries to deliver the message copy to the destination. Here, the distribution of the message copies can also be done in different ways. In source spraying, only the source node is eligible to distribute the messages (see upper part of the Fig. 1 for illustration of source spraying). However, in binary spraying, the nodes that receive a message copy also help in distributing the remaining copies to other nodes. For this, when the source node meets a node without the message copy and gives a copy of the message to that node, it also gives the right of distributing half of the remaining copies to that node. Consequently, a fast spreading of all copies is achieved.

Previous Algorithms: There are many routing algorithms proposed based on replication of messages. However, some of them work under unrealistic assumptions, such as exact knowledge of node trajectories, or node meeting times and durations. Yet, there are also a significant number of studies assuming zero knowledge about the aforementioned features of the nodes. Epidemic Routing [6] is one of the most important and popular replication based algorithms falling in this category. Basically, during each contact between any two nodes, the nodes exchange their data so that they both have the same copies. As the result, the fastest spread of copies is achieved yielding the shortest delivery time and minimum delay. The major drawback of this approach is excessive use of bandwidth, buffer space and energy due to the uncontrolled and greedy spreading of copies. Therefore, several other algorithms limiting the number of copies have been proposed [7]-[14]. In Prophet [7], the copies are exchanged between nodes in probabilistic manner. In [10], MaxProp [11] and SCAR [12], the message is replicated to encountered node only if that node has higher delivery probability (computed from contact history) than the current holder of the message. In Spray and Wait [13] copies are only given to a limited number of nodes but randomly. For many other algorithms, [15] is a good survey to look at.

2.2 Erasure coding based Routing

Overview: Erasure coding (EC(k,R)) [17] is a coding scheme which processes and converts a message of k data blocks into a large set of Φ blocks such that the original message can be constructed from a subset of Φ blocks (see lower part of Fig. 1 for illustration). Here, Φ is usually set as a multiple of k and $R = \Phi/k$ is called replication factor of erasure coding. Under optimal erasure coding, k blocks are sufficient to construct the original message. However, since optimal coding is expensive in terms of CPU and memory usage, near optimal erasure coding is used requiring $k + \epsilon$ blocks to recover the original message. In [16], the average value of ϵ is reported as $k/20$ for Tornado codes. Therefore, following the previous studies, for simplicity we ignore ϵ.

There are various erasure coding algorithms including Reed-Solomon coding and Tornado coding. These algorithms differ in terms of encoding/decoding efficiency, replication factor R and minimum number of code blocks needed to recover the original message. Due to its simplicity and linear time complexity, we will use Tornado codes in this paper. The encoding/decoding complexity in Tornado coding is proportional to $\Phi \ln(1/(\epsilon - 1))P$ where P is the length of encoding packets.

Previous Algorithms: One of the first studies utilizing the erasure coding approach in the routing of DTNs is [17]. In that study, Wang et al. present the advantages (i.e. robustness to failures) of erasure coding based routing over the replication based routing. In [18], optimal splitting of erasure coded blocks over multiple delivery paths (contact nodes) to optimize the probability of successful message delivery is studied. A similar approach focusing on non-uniform distribution of encoded blocks among the nodes is also presented in [19]. As an extension of this work, in [20], authors also utilize the information of a node's available resources (buffer space etc.) in the evaluation of the node's capability to successfully deliver the message. In [21], a hybrid routing algorithm combining the strengths of replication based and erasure coding based approaches is proposed. In addition to encoding each message into large amount of small blocks, the algorithm also replicates these blocks to increase the delivery rate.

Table 1. Notations

Symbol	**Definition**
N	The total number of nodes in the network
L	Number of copies of a message
M	Average size of a message (bytes)
k	Number of equal size blocks that a message is split into in erasure coding based routing (k_{max} is upper bound for k)
R	Replication factor used in erasure coding of a message
Φ	$k \times R$, total number of blocks generated in erasure coding based routing
Φ_i	Total number of encoded blocks distributed to the network by the end of i^{th} period in erasure coding based routing
$\L_i$	Total number of message copies distributed to the network by the end of i^{th} period in replication based routing
R_{opt}	Optimum value of R in single period case
R^*	Replication factor used in multi period case
t_d	Message delivery deadline or TTL of messages (time units)
$p(x)$	Probability of delivery of an encoded block at time x
d_r	Desired delivery rate
τ	Total cost of delivery of a message
$1/\lambda$	Average inter-meeting time of nodes
T_s	End of distributing all messages
$EC(k,R)$	Erasure coding with parameters k and R
α	The percent of kR messages that are distributed in the first period of $EC(k,R)$
p_d	Probability of delivery

3 Network Model and Assumptions

We assume that there are N nodes moving on a 2D torus according to a random mobility model. All nodes are assumed identical and the meeting times of nodes are assumed to be independent and identically distributed (IID) exponential random variables. Furthermore, the nodes are assumed to have sufficient buffer space so that no message will be dropped (Since the proposed algorithm deals with smaller number of message copies, nodes will not need large buffer sizes.). By L, we denote the number of copies of each message distributed to the network.

We also assume that the time elapsing between two consecutive encounters of a given pair of nodes is exponentially distributed with mean EM. Note that EM changes according to the mobility model used for nodes but it can be derived once the network parameters and the assumed mobility model are known [22].

In the paper, the term *period* refers to the time duration from the beginning of a message distribution (spraying) phase to the beginning of the next one. Moreover, to improve readability, we give the list of symbols used in the rest of the paper in Table 1.

4 Multi-period Replication based Routing

In this section, we present the details of the proposed multi-period message distribution based algorithm in replication based routing. We first model the spreading of message copies with respect to time and show under what condition it is more effective to use multi-period spraying than the single period spraying where all the message copies are distributed at the beginning.

We use a similar model as in Spray and Wait algorithm [23] where all the copies are sprayed (distributed) to other nodes at the beginning and the delivery of any of them is waited. Note that the delivery of a message can happen both in spray and wait phases. Since the node meeting times are independent identically distributed random variables, the cumulative distribution function (cdf) of probability of delivery (p_d) at time t when there are L copies of the message in the network is $p_d = 1 - e^{-\alpha L t}$ where $\alpha = 1/EM$ is the inverse of the expected intermeeting time between two consecutive encounters of any pair of nodes. During waiting phase, since L is constant, p_d grows with the same L value. However, since the number of copies increases during the spraying phase, p_d function changes each time a new copy is distributed to other nodes.

To simplify the analysis of message delivery probability, we assume in this paper that $M >> L$ which is often true in DTNs and which we enforce by limiting permissible values of L. Moreover, since for DTNs to be of practical use, the delivery probability p_d must be close to 1, we assume also that desired $p_d \geq 0.9$. From these two assumptions it follows that the formula $p_d = 1 - e^{-\alpha L t}$ is a good approximation of the delivery probability at time $t \geq t_d$ [28].

In the next sections, we elaborate the two, three and multi-period variants of the proposed algorithm in replication based routing. In each, we use updated version (depends on the number of periods) of the above formula for probability of delivery and we derive the average number of copies used by the algorithm.

4.1 Two Period Case

In Fig. 2, we give a sketch of what we want to achieve with two-period algorithm. In this specific version of the algorithm, we allow two different spraying phases. The first one starts without delay and the second one starts 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

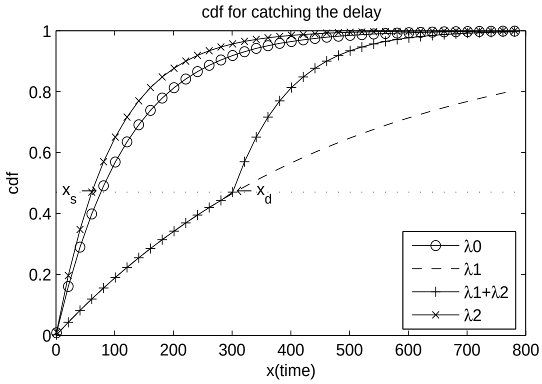

Fig. 2. The cumulative distribution function of probability of message delivery with different number of copies sprayed in two different periods.

proper setting, the average number of copies sprayed until the delivery time can be lower than in the case of spraying all messages at the beginning, while the delivery probability by the deadline remains the same.

If there are two periods until the message delivery deadline, the questions that need to be answered are *"how should we split the time interval until deadline into two periods optimally (optimal x_d in Fig. 2)?"* and *"how many copies should we distribute in each period?"*

Assume that there are L copies (with the copy in the source node) of a message to distribute. Single period spraying distributes all of these copies at the beginning to achieve the desired p_d by the deadline t_d. Let's further assume that the *Two Period Spraying* algorithm sprays L_1 copies to the network at the beginning (first period) and additional $L_2 - L_1$ copies at time x_d, the beginning of the second period. Then, the cdf of the probability of delivery at time x is:

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

Our objective with two-period spraying based algorithm is to meet the delivery probability of single period spraying based routing ($p_d = 1 - e^{-\alpha L t}$) and to obtain an average copy cost smaller than L (the cost of single period routing). Hence, by the delivery deadline, t_d, the following inequality must be satisfied:

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

As x_d gets larger, the average number of copies used decreases when L_1 and L_2 values remain constant. Therefore, to decrease the number of copies used, we need to delay the start of second period as later as possible. Thus, for given t_d, L, L_1 and L_2, the optimal x_d is the largest possible:

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

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

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

Note that if the message is not delivered in the first period, then the cost (we define cost as the number of copies used per message) becomes L_2 copies. Substituting x_d in the above and taking derivative of c_2 with respect to L_2, we get:

$$\begin{aligned} c_2(L_1, L_2) &= L_1 + (L_2 - L_1)e^{-\alpha L_1 t_d \frac{L_2 - L}{L_2 - L_1}} \\ \frac{dc_2}{dL_2} &= \left(1 - \alpha L_1 t_d + \alpha L_1 t_d \frac{L_2 - L}{L_2 - L_1}\right) e^{-\alpha L_1 t_d \frac{L_2 - L}{L_2 - L_1}} \end{aligned}$$

Comparing this derivative to zero, we obtain optimal L_2 for given L_1:

$$L_2^* = L_1 + \alpha L_1 t_d (L - L_1) > L_1$$

Hence $L_2^* - L_1 = \alpha L_1 t_d (L - L_1)$ and therefore:

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

Again, by taking the derivative of c_2^* in regard of L_1, and comparing it to zero, we can obtain the optimum value of L_1 (see the discussion and derivation in [28]).

Algorithm 1 FindOptimalsInTwoPeriods(L, α,t_d)

```
opt_cost = L; opt_cts = [L, L]
for each 0 < L_1 < L do
    L_2floor = max(L + 1, L_1 + ⌊αL_1 t_d (L − L_1)⌋)
    for L_2 = L_2floor, L_2floor + 1 do
        if c_2(L_1, L_2) <opt_cost then
            opt_cost = c_2(L_1, L_2); opt_cts = [L_1, L_2]
        end if
    end for
end for
return opt_cts
```

We can also find the optimal values of L_1 and L_2 by enumeration. From the equation defining $c_2(L_1, L_2)$, it is clear that the average number of copies sprayed by our algorithm is larger than L_1. Therefore, to decrease the average number of copies below L, L_1 must be smaller than L. As a result, $0 < L_1 < L$ must be satisfied. Since the possible values for all L_1 variables are integers, we can use enumeration method as explained in Algorithm 1 and obtain the optimal values relatively quickly, in $O(L)$ steps.

4.2 Three Period Case

Assuming that there will be three spray and wait periods until the delivery deadline, we first find the cdf of delivery probability. Let x_{d_1} and x_{d_2} denote the end time of first and second periods, respectively (thus they also denote the start of second and third periods). Then:

$$cdf(x) = \begin{cases} 1 - e^{-\alpha L_1 x} & [0, x_{d_1}] \\ 1 - e^{-\alpha L_2 (x - x_{s_2})} & (x_{d_1}, x_{d_2}] \\ 1 - e^{-\alpha L_3 (x - x_{s_3})} & (x_{d_2}, x] \end{cases}$$

where $x_{s_2} = x_{d_1} \frac{L_2 - L_1}{L_2}$ and $x_{s_3} = x_{d_2} \frac{L_3 - L_2}{L_3} + x_{d_1} \frac{L_2 - L_1}{L_3}$.

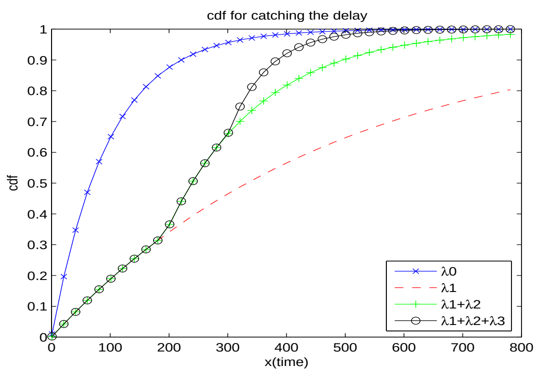

Fig. 3. The cumulative distribution function of delivery probability with copies sprayed in three different periods.

Fig. 3 illustrates what we want to achieve with three period variant of proposed algorithm. We want to obtain at least the same p_d by t_d while minimizing the average number of copies used. That is, we need to satisfy the following inequality:

$$\begin{aligned} 1 - e^{-\alpha L t_d} &\leq 1 - e^{-\alpha L_3 (t_d - x_{s_3})} \\ L t_d &\leq L_3 (t_d - x_{s_3}) \\ x_{d_2}(L_3 - L_2) + x_{d_1}(L_2 - L_1) &\leq t_d (L_3 - L) \end{aligned}$$

When all other parameters L_1, L_2, L_3, x_{d_1} are kept constant, minimum average copy cost is achieved when:

$$x_{d_2} = \frac{t_d(L_3 - L) - x_{d_1}(L_2 - L_1)}{L_3 - L_2}$$

Furthermore, the average number of copies used in this three period spraying is:

$$c_3(L_1, L_2, L_3, x_{d_1}) = L_1 + (L_2 - L_1)e^{-\alpha L_1 x_{d_1}} + (L_3 - L_2)e^{-\alpha L_2(x_{d_2} - x_{s_2})}$$

Substituting x_{s_2} and x_{d_2}, taking partial derivative and comparing it to zero, as in two-period case, we can obtain the formula for optimum x_{d_1} (See derivation and discussion in [28]).

$$x_{d_1} = \frac{\alpha t_d L_2(L_3 - L) + \ln(L_1/L_3)(L_3 - L_2)}{\alpha L_2(L_3 - L_1)}$$

Then, we can easily obtain formula $c_3^*(L_1, L_2, L_3)$ by substituting x_{d_1} in the cost function. Since $L_1 < L < L_3$ and $L_1 \leq L_2 \leq L_3$ and all these values are integers, by enumeration similar to the one in two-period case, we can simply find the (L_1, L_2, L_3) tuple that gives the minimum average number of copies for a given L. However, to use enumeration, we need to establish bounds on both L_2 and L_3. In [28], we compute and show these boundaries and give a detailed discussion of enumeration that works for three-period case.

4.3 Recursive Partitioning of Periods

In this section, we show how we can increase the number of periods in which the message copies are distributed by recursive partitioning to decrease the cost of spraying even more. Similar to the analysis in *Two Period Case* section which finds the optimum partitioning of the entire time interval from the start to the delivery deadline into two periods, we can obtain optimal partitioning of each period. Although this may not generate the optimal partitioning of entire time interval into the resulting number of periods, it still decreases the spraying cost.

In Fig. 4 we give an illustration of the recursive partitioning idea. To obtain three periods from two periods, we can partition either the first period (with parameter λ_1) or the second period (with λ_2) and select the one which gives the minimum cost. That is, we select (λ_3, λ_4, λ_2) or (λ_1, λ_5, λ_6) as the exponential factors in the corresponding three exponential functions. Moreover, once we have three periods, we can 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 number of copies after spraying in i^{th} period and x_{d_i} denote the end time of that period. Then, the cdf of delivery probability by time x is:

$$cdf(x) = \begin{cases} 1 - e^{-\alpha L_1(x - x_{s_1})} & [0, x_{d_1}] \\ 1 - e^{-\alpha L_2(x - x_{s_2})} & (x_{d_1}, x_{d_2}] \\ \ldots & \\ 1 - e^{-\alpha L_k(x - x_{s_k})} & (x_{d_{k-1}}, x] \end{cases}$$

where $x_{s_i} = \frac{x_{s_{i-1}} L_{i-1} + x_{d_{i-1}}(L_i - L_{i-1})}{L_i}$ and $x_{s_1} = 0$.

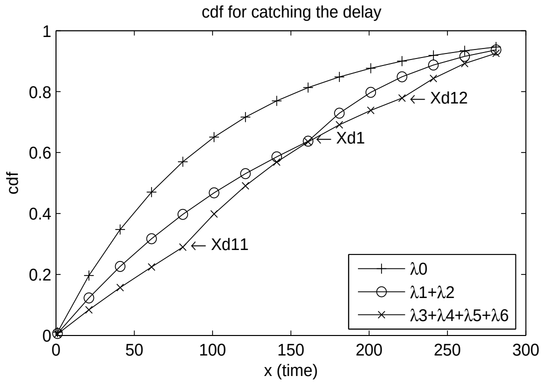

Fig. 4. Recursive partitioning algorithm for defining additional spraying in an attempt to further decrease the total cost of spraying.

Our objective is to increase the number of periods to $k+1$ while decreasing the total cost for spraying with at least the same delivery probability by the deadline. Algorithms 2 and 3 summarize the steps to achieve this goal.

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

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

We partition each period into two periods and compute the new cost for the current partitioning. Then, from all partitions, we select the one that achieves the lowest cost. For each period i, we need to find new number of copies L_i^- , L_i^+ to assign to each of the two newly created periods into which the original period is split. The delivery probability at the end of the both periods needs to stay unchanged but the average cost should be smaller than the original average cost of period i. For each period, except the last one, we need to satisfy $L_{i-1} < L_i^- < L_i^+ < L_{i+1}$. Then, for the given L_i^-, L_i^+, optimal start point of second inner period, x_{split}, (where spraying of additional $L_i^+ - L_i^-$ copies starts) is [28]:

$$x_{split} = \frac{x_{d_i}(L_i^+ - L_i) + x_{d_{i-1}}(L_i - L_i^-)}{L_i^+ - L_i^-} \tag{1}$$

For the last period, the boundary for L_k^+ is [28]:

$$L_k^+ < L_k^- + (L_k - L_k^-)\frac{1 - p_k}{1 - p_d} = L_{k+1}.$$

where p_k denotes the probability of message delivery before the period k starts. In Algorithm 3, we show how the optimal partitioning of a single period i (where $0<i<k+1$) can be found. For convenience, we denote $L_0=0$. For each pair of numbers (L_i^-, L_i^+) such that $L_{i-1} < L_i^- < L_i^+ < L_{i+1}$, the cost of spraying is calculated and the pair with the lowest cost is selected. Clearly, the complexity of this algorithm is $O(L_2)$.

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

f_1 = $cdf(x_{i-1})$, f_2 = $cdf(x_i)$
min_cost = $L_i(f_2$-$f_1)$ //current cost of period
for each $L_{i-1} < L_i^- < L_i$ **do**
  for each $L_i^- < L_i^+ < L_{i+1}$ **do**
    Compute x_{split} and x_{s^-}
    f_3 = $cdf(x_{split})$
    internal_cost = $L_i^-(f_2$-$f_1)$+$L_i^+(f_2$-$f_3)$
    if internal_cost<min_cost **then**
      min_cost = internal_cost
      $x_{opt} = x_{split}$ and $[L_{opt}^-, L_{opt}^+] = [L_i^-, L_i^+]$
    end if
  end for
end for
x'_d[ ] = $[x_{d_1},...,x_{d_{i-1}},x_{opt},x_{d_i},...,x_k]$
L'[ ] = $[L_1,...,L_{i-1},L_{opt}^-,L_{opt}^+,L_{i+1},...,L_k]$
return $[x'_d,L']$

5 Multi-period Erasure Coding based Routing

In this section, we present the details of applying multi-period idea to erasure-coding based routing. Since the message at source node is encoded into different blocks which have different contents than each other and at least k of these blocks are needed at destination node to reconstruct the original message, the problem here requires a different analysis than it is in replication based routing (See Fig. 1 for comparison of replication and erasure coding based routing).

Let $p(x)$ denote the cdf of a single node's probability of meeting the destination at time x after it received an encoded message[2]. The probability that there are already k messages gathered at the destination node at time x is then:

$$P(x, \Phi) = \sum_{i=k}^{\Phi} \binom{\Phi}{i} p(x)^i (1 - p(x))^{\Phi - i}$$

Here, note that the erasure coding based routing reduces to the replication based routing when $k = 1$.

When t_d and d_r given, we can compute the optimum parameters minimizing the cost while achieving d_r at t_d using the following relation:

$$(k, R) = \arg\min\{\tau | P(t_d, \Phi) \geq d_r\}$$

where τ is the cost[3] of erasure coding based routing. In above, although the value of k can change from 1 to infinity in theory, when k is large, many small blocks are created (might exceed the total number of nodes) incurring high processing cost and low bandwidth utilization. Therefore we assume an upper bound, k_{max}, for k and compute the optimal (k, R) accordingly.

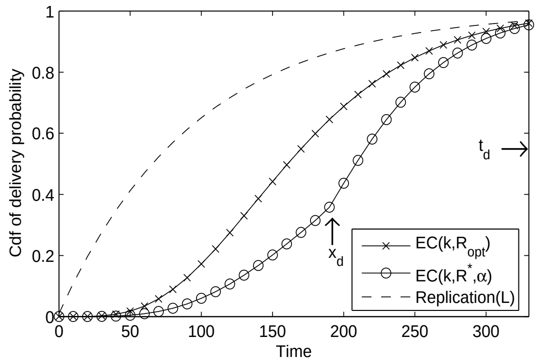

Fig. 5. Cumulative distribution function of probability of message delivery in two period erasure coding routing.

In Fig. 5, we give a comparison of the cdf's of optimal replication and erasure coding based routing algorithms. Clearly, with erasure coding, much lower cost can be achieved while still achieving the desired d_r by t_d.

[2] For simplicity, we assume that the total number of encoded messages to distribute is not too large ($t_d >> T_s$) and all relay nodes in the network get encoded messages at about the same time.

[3] When source spraying is used $\tau = O(MR)$, however, since the contents of encoded blocks are different, when binary spraying is used, $\tau = O(MR\log(kR))$ [25]. Therefore, we use source spraying in erasure coding based routing.

Furthermore, we can decrease the cost of erasure coding based routing via distributing the encoded blocks of the message in multiple periods. For example, in two period erasure coding routing, instead of distributing all encoded blocks of the messages at the beginning, we spray only some of them at time 0 and wait for the delivery of sufficient number of messages at the destination. If the delivery has not happened yet until x_d, we distribute more encoded blocks to the network so that we increase the probability of delivery of at least k blocks to destination.

In Fig. 5, we illustrate the goal we want to achieve here with plot $EC(k, R^*, \alpha)$. Assume that the optimum parameters in single period case are k and R_{opt}. In two-period erasure coding routing, source node generates[4] $\Phi_2 = kR^*$ encoded blocks at the beginning and allows the distribution of only $\Phi_1 = \alpha k R^*$ of them $(0 < \alpha < 1)$ in the first period. Then, with the start of second period, remaining $\Phi_2 - \Phi_1$ message blocks are distributed. In the first period, the cdf of delivery probability at time x is P(x,Φ_1), however, in the second period, we need to combine the independent delivery probabilities of the first and second period messages to derive a formula:

$$P(x, \Phi_1, \Phi_2) = \sum_{i=k}^{\Phi_2} \left(\sum_{j=l_1}^{l_2} P'(x, j, \Phi_1) P'(x\text{-}x_d, i\text{-}j, \Phi_2\text{-}\Phi_1) \right)$$

$$\text{where } P'(x, j, \Phi_1) = \binom{\Phi_1}{j} p(x)^j (1 - p(x))^{\Phi_1 - j}$$

$$l_1 = \max\{0, i - \Phi_2 + \Phi_1\} \text{ and } l_2 = \min\{i, \Phi_1\}$$

Here, for a given Φ, we want to find a (Φ_1, Φ_2) pair that gives the minimum average cost while maintaining d_r by t_d. To meet the delivery rate of single period, we need to satisfy:

$$R^* > R_{opt}$$

$$P(t_d, \Phi_1, \Phi_2) \geq P(t_d, \Phi)$$

Also, to achieve a lower average cost than in single period case:

$$P(x_d, \Phi_1)\Phi_1 + (1 - P(x_d, \Phi_1))\Phi_2 \leq \Phi$$

$$\frac{\Phi_2 - \Phi}{\Phi_2 - \Phi_1} \leq P(x_d, \Phi)$$

Using the above inequalities, we can find optimal Φ_1 and Φ_2 values using again enumeration method [25]. Here, we only presented the analysis of two-period erasure coding based routing, but a similar analysis for more periods can be performed. Moreover, recursive partitioning idea presented in previous section can also be applied to increase the number of periods to achieve lower cost (see more discussion in [25]).

[4] Since complexity of encoding is linear in Tornado codes, this will cause a linear increase in the complexity.

6 Acknowledgment of Delivery

In DTN routing, since the nodes are intermittently connected, the way the nodes are acknowledged about the delivery of the messages is a crucial issue. Even though the message has already been delivered to destination, some nodes may still continue to distribute message copies or erasure coded blocks of the message to other nodes unless they are informed about the delivery. In this paper, we propose two acknowledgment mechanisms to notify the nodes about the delivery of the messages. Both have advantages and disadvantages over each other. Hence, we compare the performances of both types of acknowledgment by showing how they affect the results of our algorithm in simulations.

Type I Acknowledgment When the message is delivered, destination node starts an epidemic routing [6] based spreading of acknowledgment packets. That is, each node receiving this packet also distributes a copy of it to other nodes. Note that, since acknowledgment packets usually carry only the id of the delivered message, the cost of routing here is much smaller than it is in epidemic routing with data messages. However, since epidemic spreading of acknowledgment packets requires some time to reach all nodes, cost of spraying can be increased due to redundant spraying of already delivered message.

Type II Acknowledgment When the destination receives the messages, it sends an acknowledgment to all nodes with one time broadcast over a powerful radio. Although using powerful radio can potentially generate more cost than type I acknowledgment, since the acknowledgment messages are short, the broadcast is expected to be inexpensive. Besides, to make this scheme more efficient, we use the following bio-inspired idea.

Consider an environment where individuals are infected by different pathogens at different times. Each pathogen has an incubation period during which the infectee is not contagious. As the incubation period ends, an infectee starts to infect others. We assume that there are effective vaccines for all pathogens and we want to vaccinate the entire population with the proper mix of vaccines in the most efficient way. The best way to achieve this goal is to wait until the closest end of an incubation period of any infectee and to apply the vaccines for all observed diseases to the entire population at that time. Such delayed vaccination campaign allows emergence of new diseases, possibly with new types of pathogens, before letting infectees infect others and decreases the number of necessary vaccination campaigns, each with all vaccines necessary to stop already started epidemics.

Inspired by this effective vaccination idea, we use the following efficient acknowledgment scheme. As the destination receives messages, it waits until the closest period change time (x_d) of any of the received messages. Then, it broadcasts an acknowledgment of all so far received messages at that time. Hence, the destination broadcasts acknowledgments relatively infrequently. Even though acknowledgments of some messages are delayed, spraying of any received messages after the delivery time are suppressed.

7 Simulation Model and Results

To evaluate the proposed algorithms, we have developed a Java-based discrete event-driven simulator and performed extensive simulations for each routing type.

First, we randomly deployed 100 mobile identical nodes (including the sink) on a 300 m 300 m torus. The nodes move according to random walk mobility model[5]. Each node selects a random direction ([0, 2π]) and a random speed from the range of [4m/s, 13m/s], then goes in that direction during a randomly selected epoch of duration from the range of [8s, 15s]. When the epoch ends, the same process runs again and new direction, speed and epoch duration are selected. The transmission range of each node (except the sink that has high range of acknowledgment broadcast in TYPE II case) is set to 10 m. Note that, the generated network under this setting provides a very sparse mobile network which is the most common case in real DTN deployments.

7.1 Results for Multi-period Replication based Routing

Firstly, assuming that the desired p_d by given t_d is 0.99[6], we have found the optimum number of copies for both two period (2p) and three period (3p) cases. Table 2 shows the values of these optimum L_i's for different t_d values and the minimum L value that achieves the desired p_d in single period (1p) algorithm. Clearly, as the deadline decreases, L_{min} (minimum L achieving p_d by t_d) in 1p increases because more copies are needed to meet the desired p_d by t_d. Such an increase is also observed for L_i values used in both 2p and 3p algorithms.

With optimum x_{d_1}, x_{d_2} and L_i values computed from theory, we performed simulations to find the average number of copies used per message when these optimum values are used. We generated 3000 messages from randomly selected nodes to the sink node whose initial location was also chosen randomly. Furthermore, since in replication based routing, binary spraying provides faster spraying than source spraying does with the same cost [25], we used binary spraying while distributing the allowed number of copies in each period. We took the average of 10 different runs with different seeds.

In Fig. 6 and Fig. 7, we show the average numbers of copies used when the optimum L_i values are used in 2p and 3p variants of proposed algorithm. Since our analysis considers the cost at the exact delivery time while computing the optimum L_i values, to make a fair comparison of theory results with simulation

[5] We also performed simulations using other mobility models (random waypoint etc.). Since the results are similar, for brevity, we did not include them here. However, these extensive results can be reached from [25]-[28].

[6] We have selected a high desired delivery probability because it is the most likely case in real applications. However, in [28], we looked at the effects of different p_d values.

t_d(sec)	L_{min} in 1p	Optimum L_i's in 2p	Optimum L_i's in 3p
200	12	7,22	6,12,27
250	9	5,15	5,9,19
300	8	5,14	4,8,18
400	6	4,11	3,6,14
500	5	3,9	2,4,11
600	4	2,7	2,4,9
700	4	2,8	2,4,10
800	3	2,5	1,2,6
900	3	2,6	1,2,7

Table 2. Optimum L_is, the number of copies in each period that minimize the average number of copies and preserve the desired probability of delivery.

results, we report the average number of copies simulating Type II acknowledgments. However, we also include the results when Type I acknowledgment is used. From both figures, we observe that analysis results are very close to Type II results but as the deadline gets tight, they become an upper bound for Type II results. The reason behind this is as t_d gets smaller, the sufficient number of copies (L_{min}) to achieve desired d_r by t_d increases, thus optimum L_i values in 2p and 3p become larger. Hence, spraying period takes longer. Besides, this also increases the difference between the average numbers of copies with Type I and Type II acknowledgments because as L_i values gets larger, more nodes carrying message copies need to be acknowledged about the delivery when Type I acknowledgment is used.

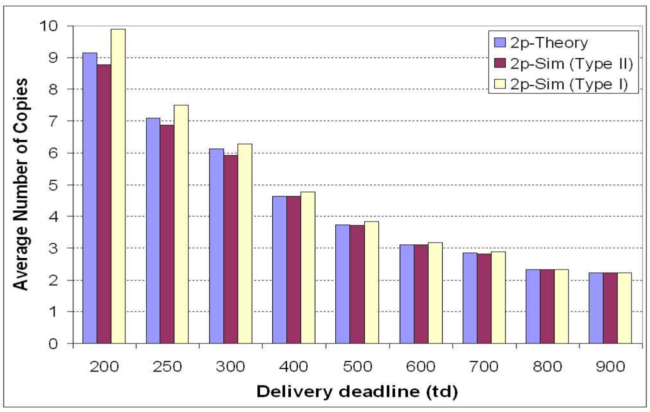

Fig. 6. The comparison of the average number of copies obtained via analysis and simulation for the two-period case.

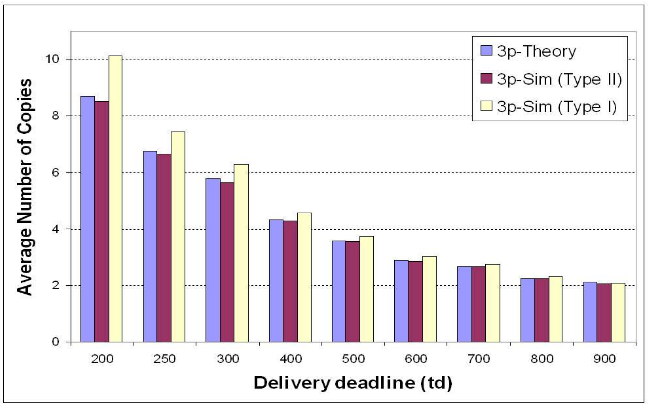

Fig. 7. The comparison of the average number of copies obtained via analysis and simulation for the three-period case.

In Table 3, we present the average number of copies used in three variants of the algorithm with different types of acknowledgment mechanisms and different t_d values. Here, we observe that in both acknowledgment types, 3p algorithm uses fewer copies on average than 2p or 1p algorithm does. However, when Type I acknowledgment is used, the saving in the number of copies obtained by 3p algorithm decreases. Furthermore, in some cases ($t_d = 200$s), its performance becomes worse than 2p algorithm. This is because when the deadline gets tight, the number of copies that are sprayed to the network increases so that the number of nodes carrying the message copies increases and epidemic like acknowledgment takes longer. As a result, more redundant copies are sprayed by the nodes having message copy before they are informed about the delivery.

Moreover, we also notice that in the proposed algorithms even with Type I acknowledgment, we can achieve lower average cost than in single period spraying algorithm with Type II acknowledgment. Also, remark that in single period spraying algorithm with L message copies, the average number of message copies sprayed to the network is less than L. This is simply because even in single period spraying which does all spraying at the beginning, there is a non-zero chance that the message will be delivered before all copies are made.

When we compute the percentage of the savings[7] achieved in the number of copies used with the proposed multi-period algorithms, we obtain the chart in Fig. 8. From the results, we observe that 3p algorithm provides higher savings than 2p algorithm. Moreover, it is clear that the savings with Type II acknowl-

[7] We define saving as $(L\text{-}L_{avg})/L$ with the given t_d. Here, L is the average copy count used in single period spraying and L_{avg} is the average copy count used in the multi-period spraying algorithm.

		Type I			Type II		
t_d(sec)	L_{min}	1p	2p	3p	1p	2p	3p
200	12	11.61	9.89	10.12	10.92	8.77	8.51
250	9	8.79	7.50	7.44	8.52	6.88	6.65
300	8	7.80	6.28	6.28	7.58	5.94	5.62
400	6	5.87	4.78	4.55	5.78	4.64	4.28
500	5	4.91	3.84	3.72	4.86	3.73	3.54
600	4	3.96	3.18	3.02	3.93	3.10	2.85
700	4	3.96	2.89	2.74	3.93	2.83	2.66
800	3	2.97	2.33	2.31	2.95	2.31	2.24
900	3	2.97	2.24	2.09	2.96	2.23	2.07

Table 3. Average number of copies used in single-period (1p), two-period (2p) and three-period (3p) replication based algorithms.

edgment are higher than the savings with Type I acknowledgment in both 2p and 3p algorithms. The difference between the savings of Type I and Type II acknowledgments gets smaller as the deadline increases. This is because larger t_d decreases the number of copies sprayed to the network, resulting in acknowledgments reaching all nodes carrying message copies earlier.

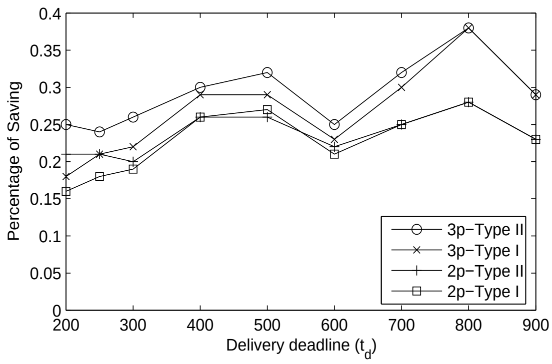

Fig. 8. The percentage of savings achieved by the proposed algorithms with two different acknowledgment schemes.

On the other hand, we also observe fluctuations even in the savings of a single algorithm with different delivery deadlines. This is because for some consecutive t_d values (i.e., t_d = 600s, 700s), L_{min} value in 1p algorithm which achieves the desired p_d is the same (i.e. $L_{min} = 4$) while L_i values in multi-period algorithms are different. In these cases, multi-period algorithms take the advantage of spraying in multiple periods and delay the spraying further when the deadline is larger (for example in 2p algorithm, if t_d =600s, then x_{d_1} =360s and the optimum (L_1, L_2) = (2, 7) but if t_d =700s, then x_{d_1} =466s and the optimum (L_1, L_2) = (2, 8)). Hence, multi-period algorithms can provide more saving over single period algorithm in such cases.

In addition to the evaluation of the proposed protocol with random mobility models, we have also looked at its performance on real DTN traces. In [28], we present the simulation results based on RollerNet [24] traces and show the reduction of cost by using multi-period idea experimentally.

7.2 Results for Multi-period Erasure Coding based Routing

In this section, we present the results obtained for multi-period erasure coding based routing. Table 4 shows the minimum costs incurred by $EC-1p$ and $EC-2p$ algorithms with two different types of acknowledgments. In both algorithms, we computed the optimal parameters which provide minimum average costs and used them in simulations. In $EC-2p$ algorithm, we used $k_{max}=5$. First of all, even though we did not show it here for the sake of brevity, in both algorithms the desired delivery rate is achieved by the given deadlines. Yet, their costs are different. For all t_d values shown, the cost of the algorithm when Type I acknowledgment is used is higher than the cost of the algorithm when Type II acknowledgment is used. This result is expected because in Type I acknowledgment, extra time is needed to inform the source node about the delivery with epidemic like acknowledgment. However, during this extra time, the source node continues to distribute the remaining encoded messages it has, thus the cost of the algorithm increases.

t_d	Cost of EC-1p			Cost of EC-2p		
sec	Opt(R,k)	Type I	Type II	Opt(R^*,α, x_d)	Type I	Type II
600	(3,2)	3.43	3.42	(5, 0.4, 410)	3.23	3.19
500	(3,3)	3.57	3.56	(5, 0.4, 345)	2.99	2.95
400	(4,3)	4.45	4.43	(6, 0.5, 270)	4.12	3.98
300	(5,3)	5.36	5.32	(7, 0.5, 200)	5.15	4.95
250	(5,5)	5.25	5.17	(8, 0.5, 185)	5.38	5.10

Table 4. Minimum average costs of single and two period erasure coding algorithms.

Moreover, for almost all t_d values, the cost of $EC-2p$ algorithm is smaller than the cost of $EC-1p$ regardless of the type of acknowledgment used. This clearly shows the superiority of $EC-2p$ over $EC-1p$ algorithm. We also observe that as the deadline gets tight (decreases), the improvement achieved by $EC-2p$ algorithm decreases because with shorter deadline, more encoded blocks are generated. Hence, the required time to distribute all encoded blocks and also the time needed to inform source node in Type I acknowledgment increases. Consequently, in some cases (t_d=200s), the cost of $EC-2p$ algorithm becomes higher than the cost of $EC-1p$ algorithm. However, in most of the cases, $EC-2p$ still performs better than $EC-1p$ algorithm does. Besides, for the same t_d values, the cost difference between Type I and Type II acknowledgments in $EC-2p$ is larger than it is in $EC-1p$ algorithm. This is because more encoded message blocks are generated in $EC-2p$ algorithm due to usage of multi-period idea but this also caused spraying of more redundant encoded blocks before the acknowledgment arrives to source node.

8 Conclusion and Future Work

In this paper, we introduced a bio-inspired idea of spraying message copies in multiple periods. To this end, we applied our idea to both replication based and erasure coding based routing. Then, using analysis and simulations, we compare the performance of the proposed approaches with the corresponding single period algorithms. In simulation results, we validated our analysis and showed that the cost of routing can be decreased by using multi-period idea while maintaining the desired delivery rate by the deadline.

In the future work, we will investigate how more realistic radio links and mobility models affect our algorithm. Moreover, we will update the proposed protocol for heterogeneous networks in which node meeting behaviors vary among different pairs of nodes.

References

1. Delay tolerant networking research group, http://www.dtnrg.org.
2. 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.
3. J. Ott and D. Kutscher, A disconnection-tolerant transport for drive-thru internet environments, in Proceedings of IEEE INFOCOM, 2005.
4. Disruption tolerant networking, http://www.darpa.mil/ato/solicit/DTN/.
5. P. E. Sartwell, The incubation period and the dynamics of infectious disease, American Journal of Epidemiology, vol. 82, no. 2, 1966.
6. A. Vahdat and D. Becker, Epidemic routing for partially connected ad hoc networks, Duke University, Tech. Rep. CS-200006, 2000.
7. A. Lindgren, A. Doria, and O. Schelen, Probabilistic routing in intermittently connected networks, SIGMOBILE Mobile Computing and Communication Review, vol. 7, no. 3, 2003.
8. S. Jain, K. Fall, and R. Patra, Routing in a delay tolerant network, in Proceedings of ACM SIGCOMM, Aug. 2004.
9. E. P. C. Jones, L. Li, and P. A. S. Ward, Practical routing in delay tolerant networks, in Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.
10. 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.

11. J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, *MaxProp: Routing for Vehicle-Based Disruption- Tolerant Networks*, In Proc. IEEE Infocom, April 2006.
12. C. Mascolo and M. Musolesi, *SCAR: Context-aware Adaptive Routing in Delay Tolerant Mobile Sensor Networks*, Proceedings of International conference on Wireless communications and mobile computing, 2006.
13. T. Spyropoulos, K. Psounis,C. S. Raghavendra, Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks, ACM SIGCOMM Workshop, 2005.
14. A. Balasubramanian, B. N. Levine, A. Venkataramani, DTN routing as a resource allocation problem, ACM SIGCOMM, 2007.
15. I. Psaras, L. Wood and R. Tafazolli, Delay-/Disruption-Tolerant Networking: State of the Art and Future Challenges, Technical Report, University of Surrey, UK, 2009.
16. J. Byers, M. Luby, and M. Mitzenmacher, Accessing Multiple Mirror Sites in Parallel: Using Tornado Codes to Speed Up Downloads, in Proceedings of INFOCOM'99, pp. 275-284, 1999.
17. Y. Wang, S. Jain, M. Martonosi, and K. Fall, Erasure-coding based routing for opportunistic networks, in WDTN 05: Proceeding of the ACM SIGCOMM workshop on Delay-tolerant networking, 2005.
18. S. Jain, M. Demmer, R. Patra, K. Fall, Using redundancy to cope with failures in a delay tolerant network, in Proceedings of ACM Sigcomm 2005.
19. Y. Liao, K. Tan, Z. Zhang, and L. Gao, Estimation based erasure coding routing in delay tolerant networks, in International Wireless Communications and Mobile Computing Conference, Delay Tolerant Mobile Networks Symposium, Vancouver, Canada, July 2006.
20. Y. Liao, Z. Zhang, B. Ryu, and L. Gao, Cooperative robust forwarding scheme in DTNs using erasure coding, In MILCOM 2007, Oct. 2007.
21. L. Chen, C. Yu, T. Sun, Y. Chen, H. Chu, A hybrid routing approach for opportunistic networks, Proceedings of the 2006 SIGCOMM workshop on Challenged networks, p.213-220, September 11-15, 2006, Pisa, Italy.
22. T. Spyropoulos, K. Psounis,C. S. Raghavendra, Performance Analysis of Mobility-assisted Routing, MobiHoc, 2006.
23. T. Spyropoulos, K. Psounis,C. S. Raghavendra, Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-copy Case, IEEE/ACM Transactions on Networking, 2008.
24. P. U. Tournoux, J. Leguay, F. Benbadis, V. Conan, M. Amorim, J. Whitbeck, The Accordion Phenomenon: Analysis, Characterization, and Impact on DTN Routing, in Proc. INFOCOM, 2009.
25. E. Bulut, Z. Wang, B. Szymanski, Cost Efficient Erasure Coding based Routing in Delay Tolerant Networks, to appear in Proc. ICC 2010.
26. E. Bulut, Z. Wang, B. Szymanski, Minimizing Average Spraying Cost for Routing in Delay Tolerant Networks, in Ann. Conf. Int. Technology Alliance (ITA), ACITA, Imperial College London, September 2008.
27. E. Bulut, Z. Wang, B. Szymanski, Time Dependent Message Spraying for Routing in Intermittently Connected Networks, in Proc. IEEE Global Telecommunications Conference (GLOBECOM), vol., no., pp.1-6, New Orleans, November 2008.
28. E. Bulut, Z. Wang, B. Szymanski, Cost Effective Multi-Period Spraying for Routing in Delay Tolerant Networks, to appear in IEEE/ACM Transactions on Networking, 2010.