

Study of two-hop message spreading in DTNs

Antonis Panagakis, Athanasios Vaios, Ioannis Stavrakakis
Department of Informatics & Telecommunications
National & Kapodistrian University of Athens
Ilissia, 15234 Athens, Greece
{apan, avaios, ioannis}@di.uoa.gr

Abstract—In this paper, a Delay Tolerant Network environment is considered where the source is in full control of the two-hop spreading mechanism by setting key parameters such as the number of copies allowed to be spread in the network and the delay bound of the messages. The analysis allows for a differentiation between the source of the message and the intermediate nodes (in terms of e.g. transmission power or speed). Analytical expressions for the cumulative distribution function (*cdf*) of the delivery delay and the induced overhead are extracted, taking into account the fact that the source node may continue spreading copies after the message delivery. In addition, a fairly accurate approximate expression for the *cdf* of the delivery delay is also derived and validated through simulations.

I. INTRODUCTION

The routing mechanisms considered for Delay Tolerant Networks (DTNs), where typically no contemporaneous path between a potential source-destination pair exists, achieve message (data) delivery by carrying and opportunistically relaying it at each node encounter, [1]-[9], either by using a single copy or multiple copies of the message.

This work focuses on the two-hop relay algorithm [10][11], where the source node relays a copy of the message to every node it encounters; the intermediate nodes that acquire a copy of the message are allowed to forward the copy only to the destination and, thus, one hop or, maximum, two hops of communication are employed for the message delivery. The performance of the two-hop relay algorithm has been studied in [11], for a homogeneous network, in terms of the expected value and variance of the delivery delay as well as of the expected number of transmissions until the delivery of the message. Here, a more general study of the algorithm's performance is provided with respect to the setting that is considered as well as the metrics that are examined.

The setting that is considered in this work allows for a differentiation between the source of the message and the intermediate nodes (in terms of e.g. transmission power or speed), as discussed in detail in the next section. In addition, instead of allowing the source to relay the message to all the intermediate nodes that it encounters, the number of copies allowed to be spread in the network is treated here as a design parameter.

The delivery delay is fully characterized by extracting its *cdf* (cumulative distribution function). Moreover, an approximate approach is proposed that leads to a fairly accurate and much simpler expression for the *cdf*. In addition, the number of

transmissions is considered not only upon the delivery of the message, but until the actual termination of the algorithm, which takes place when the source becomes aware of the delivery. (When the message is delivered to the destination by some intermediate node, the source continues to forward the rest of the copies of the message allowed to be spread until it becomes aware of the successful delivery and, thus, the number of transmissions until message delivery is only a fraction of the total number of transmissions that will eventually take place.)

For all the above derivations, we assume that each message has a certain delay bound upon the expiration of which the message is dropped. This delay bound might be considered either as an application-specific characteristic (e.g. a calendar-related message or a meeting reminder that are expected to be delay-bounded) or as a design parameter of the algorithm (in order, for instance, to limit the spreading of copies by determining an appropriate delay bound for the messages within which their delivery is expected).

In the remainder of this paper, the model considered for this work is introduced and described in Section II. In Section III, the exact expression as well as a fairly accurate approximation are derived for the achieved delivery ratio. In Section IV, the overhead in terms of both the number of transmissions and energy consumption is studied. The conclusions are drawn in Section V.

II. MODEL DESCRIPTION

According to the two-hop relay algorithm employed here, [10][11], the source node is allowed to spread up to a maximum number of copies (K) within the network. Each time it encounters some other node with no copy of the message, it gives it one until it has only one copy available (for the destination node only). The intermediate nodes are not allowed to spread the message copy they may have to any other node than the destination.

Let $N + 1$ be the total number of nodes moving within a square area of size L^2 . It is assumed that the node intermeeting times (i.e., the time elapsed between two consecutive encounters for a given pair of nodes) are exponentially distributed. This assumption has been demonstrated to be fairly accurate in the case the communication range R is such that $R \ll L$ and that nodes move around according to the random waypoint or the random direction model, [11]. As it has been shown in [11], the rate at which a given node encounters some other node, λ ,

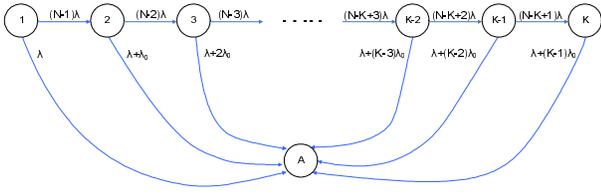


Fig. 1. The Markov chain for the two-hop relay algorithm.

may be linked to the nodes' relative speed v , communication range R and network area L^2 :

$$\lambda = c \frac{vR}{L^2}, \quad (\text{II.1})$$

where c is a constant and its value depends on the mobility model used.

The setting that is considered in this work allows both for the spreading of a restricted ($K \leq N$) number of copies and for diverse intermeeting times between the source and the other nodes represented by two parameters, λ and λ_o , respectively. From equation (II.1), it becomes evident that the differentiation between the parameters λ and λ_o may be directly associated with the transmission range (power) or/and the speed of the source and the intermediate nodes, respectively. In addition, this differentiation may be associated with a possible misbehaviour of the intermediate nodes by assuming, for instance, that an intermediate node discards a copy without further relaying it at its encounter with the destination with a certain probability $1 - q$ (q expresses the degree of cooperation of the intermediate nodes); assuming, for simplicity, a homogeneous network with respect to the transmission range and speed of the nodes, it is clear that the network may be modeled by setting $\lambda_o = q\lambda$.

The above parameters that may lead to differentiation could either be considered as non-tunable (e.g., the capabilities of some nodes to transmit by default at a higher power level or a node's misbehavior or cooperation degree), or as tunable (e.g., adjustment of the transmission range of the nodes within the technological power limitations in order to satisfy specific performance goals).

III. DERIVATION OF THE CDF OF THE DELIVERY DELAY

Under the aforementioned assumptions, the message spreading process according to the K -limited, two-hop relay algorithm may be modeled by the Markov chain depicted in Fig.1, consisting of $K + 1$ states (K states capturing the number of copies spread in the network and one absorbing state assumed to be visited when the message is delivered to the destination; generally, $K \leq N$). When there are i copies ($i \leq K$) in the network, a new copy is forwarded either to one of the $(N - i)$ nodes which do not have a copy yet (at the rate of $\lambda(N - i)$ triggering a transition from i to $i + 1$) or to the destination (at the rate of $\lambda + (i - 1)\lambda_o$ triggering a transition from i to A). Thus, the transition rates of the Markov chain of Fig.1 are

$$q(i, j) = \begin{cases} \lambda(N - i), & i = 1, \dots, K - 1, j = i + 1; \\ \lambda + \lambda_o(i - 1), & i = 1, \dots, K, j = A; \\ 0, & \text{otherwise,} \end{cases}$$

where λ and λ_o denote the exponential distribution parameter of the intermeeting times for the source and for the rest of the nodes, respectively.

Let $f_D(t)$ and $Q(t) = P(D \leq t) = \int_0^t f_D(x)dx$ denote the *pdf* (probability distribution function) and the *cdf* of the message delivery delay, correspondingly. Adapting the notation $\mathcal{L}(g)$ ($\mathcal{L}^{-1}(g)$) for the Laplace (inverse Laplace) transform of some function g , $Q(t)$ may be expressed as $Q(t) = \mathcal{L}^{-1}\left(\frac{F(s)}{s}\right)$, where $F(s) = \mathcal{L}(f_D(t))$.

Let T_i denote the sojourn time in state i and let $f_{T_i}(t) = q_i e^{-q_i t}$ denote its *pdf*. If D_i denotes the conditional total delay until the message reaches the destination given that the destination gets the message when the system is in state i , then $D_i = \sum_{j=1}^i T_j$ and its *pdf* is $f_{D_i}(t) = \mathcal{L}^{-1}(F_i(s))$, where $F_i(s) = \prod_{j=1}^i \mathcal{L}(f_{T_j}(t))$.

Now, the *pdf* of the unconditional total delivery delay D , may be expressed as

$$f_D(t) = \sum_{i=1}^K p_{d,i} f_{D_i}(t),$$

where $p_{d,i}$ denotes the probability to be in state i when the destination gets the message with

$$p_{d,i} = \begin{cases} \frac{q(i,A)}{q_i} \prod_{j=1}^{i-1} \left(1 - \frac{q(j,A)}{q_j}\right), & i = 1, \dots, K - 1; \\ 1 - \sum_{j=1}^{K-1} p_{d,j}, & i = K; \\ 0, & \text{otherwise,} \end{cases}$$

where $q_i = q(i, i + 1) + q(i, A)$.

Thus, and since $\mathcal{L}(f_{T_i}(t)) = \mathcal{L}(q_i e^{-q_i t}) = \frac{q_i}{q_i + s}$, $F(s)$ (the Laplace transform of the *pdf* of D) may be expressed as

$$F(s) = \sum_{i=1}^{K-1} B_i F_{1,i}(s) + B_K F_{1,K}(s),$$

where

$$\begin{aligned} B_i &= \frac{q(i, A)}{q_i} \left(\prod_{j=1}^{i-1} \left(1 - \frac{q(j, A)}{q_j}\right) \right) \prod_{j=1}^i q_j \\ &= (\lambda_d + (\lambda - \lambda_d)i) \lambda^{i-1} \frac{(N-1)!}{(N-i)!}, \quad 1 \leq i \leq K-1, \end{aligned}$$

$$\begin{aligned} B_K &= \left(1 - \sum_{i=1}^{K-1} \frac{q(i, A)}{q_i} \prod_{j=1}^{i-1} \left(1 - \frac{q(j, A)}{q_j}\right) \right) \prod_{j=1}^K q_j \\ &= \lambda^{K-1} (K\lambda - (K-1)\lambda_d) \frac{(N-1)!}{(N-K)!}, \end{aligned}$$

$\lambda_d = \lambda - \lambda_o$ and

$$F_{1,i}(s) = \prod_{j=1}^i \frac{1}{q_j + s}.$$

Based on the above expressions it may be concluded that

$$Q(t) = \mathcal{L}^{-1}(F_2(s)) = \sum_{i=1}^{K-1} B_i f_{2,i}(t) + B_K f_{2,K}(t), \quad (\text{III.1})$$

where $f_{2,i}(t) = \mathcal{L}^{-1}(F_{2,i}(s))$ and

$$F_{2,i}(s) = \frac{F_{1,i}(s)}{s} = \frac{1}{s} \prod_{j=1}^i \frac{1}{q_j + s} = \begin{cases} \frac{1}{s} \prod_{j=1}^i \frac{1}{\lambda N - \lambda_d(j-1) + s}, & 1 \leq i \leq K-1; \\ \frac{1}{s} \frac{1}{\lambda + (\lambda - \lambda_d)(K-1) + s} \prod_{j=1}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1) + s}, & i = K. \end{cases}$$

The derivation of $f_{2,i}(t)$ is lengthy and, thus, quoted in the Appendix.

The above expression for $Q(t)$ holds for the case $K = N$ and $\lambda_d \neq 0$. Following a similar methodology, it may be concluded that for $K < N$ and $\lambda_d = 0$, the *cdf* of the delivery delay, denoted as $Q_0(t)$, is

$$Q_0(t) = \sum_{i=1}^{K-1} B_i f_{2,i}(t) + B_K f_{2,K}(t) = \sum_{i=1}^{K-1} i \lambda^i \frac{(N-1)!}{(N-i)!} \left(\left(\frac{1}{\lambda N} \right)^i - \sum_{k=1}^i \frac{e^{-\lambda N t}}{(k-1)! \lambda N} t^{k-1} \right) + \lambda^K K \frac{(N-1)!}{(N-K)!} \left(\frac{1}{\lambda K (\lambda N)^{K-1}} - \frac{e^{-\lambda K t}}{\lambda K (\lambda(N-K))^{K-1}} - \frac{e^{-\lambda N t}}{\lambda K} \sum_{k=1}^{K-1} \frac{\left(\left(\frac{1}{\lambda N} \right)^{K-k} - \left(\frac{1}{\lambda(N-K)} \right)^{K-k} \right) t^{k-1}}{(k-1)!} \right).$$

In the special case where $K = N$ and $\lambda_d \neq 0$ it is obtained that the *cdf* of the delivery delay, denoted as $Q_N(t)$, is

$$Q_N(t) = 1 - e^{-\lambda N t} \left(1 + \frac{\lambda}{\lambda_d} (e^{\lambda_d t} - 1) \right)^{N-1}. \quad (\text{III.2})$$

For $K = N$ and $\lambda_d = 0$ the *cdf* may be obtained by taking the limit as $\lambda_d \rightarrow 0$,

$$\lim_{\lambda_d \rightarrow 0} Q_N(t) = \lim_{\lambda_d \rightarrow 0} 1 - e^{-\lambda N t} \left(1 + \frac{\lambda}{\lambda_d} (e^{\lambda_d t} - 1) \right)^{N-1} = 1 - e^{-\lambda N t} (1 + \lambda t)^{N-1}.$$

The analysis was validated through simulations. Figure 2 depicts an indicative example for the cases of $K = 4, 8, 100$ when $N = 100$, $\lambda = 0.08$ and $\lambda_o = 0.04$. For the simulations, a network of 100 nodes that move according to the random direction mobility model with a speed of $2m/sec$ within a square area of a side of $1000m$ has been considered. The source node has a transmission range of $50m$ and spreads up to 3, 7 and 99 copies of the message to the intermediate nodes ($K = 4, 8, 100$, respectively) that have a transmission range of $25m$; the results refer to the average of 20000 runs.

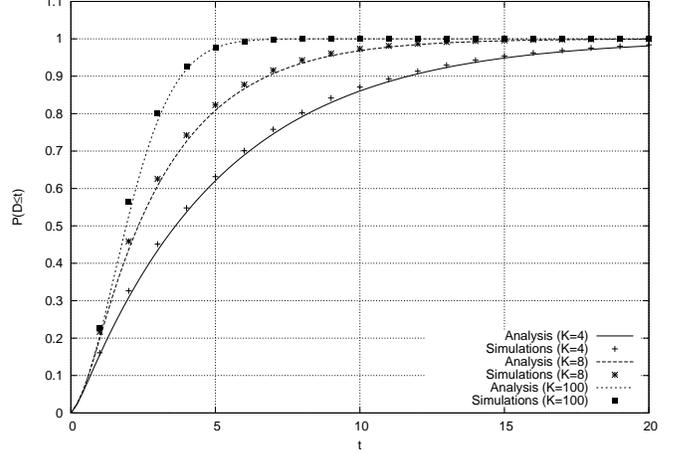


Fig. 2. Theoretical and simulation results for $N = 100$, $\lambda = 0.08$, $\lambda_o = 0.04$ and for the cases of $K = 4, 8, 100$.

A. Approximate expression for the *cdf* of the delivery delay

Instead of using the equation (III.1) in order to calculate the *cdf* of the delivery delay, an approximate expression is derived here, giving a much simpler expression that approximates fairly accurately the exact one. Its derivation is based on the observation that the *cdf* for a specific number of copies K where $K \leq N$, named *accurate cdf* hereafter, is upper bounded by two *cdfs*.

The first one, named *maximum-copy cdf* hereafter, refers to the case where the number of copies employed in the network equals the number of nodes ($K = N$), is denoted as $Q_N(t)$ and is given by equation (III.2). This is because the algorithm has exactly the same behaviour up to the time point where the first K copies are spread in the network, while, afterwards, the setting corresponding to the maximum-copy *cdf* takes advantage of the surplus copies ($N - K$) allowed to be spread to enhance its performance.

The second one, named *zero-spreadtime cdf* hereafter, refers to the *hypothetical ideal* case where upon the generation of a message, all K copies are assumed to have been spread *instantly* in the network; one copy is owned by the source node and $(K - 1)$ by some intermediate nodes. In that case, a transition to the state of the destination's having acquired the message takes place at a rate of $(\lambda + \lambda_o(K - 1))$ and, thus, the zero-spreadtime *cdf* of the delivery delay, denoted as $Q_{\hat{K}}(t)$, is given by

$$Q_{\hat{K}}(t) = 1 - e^{-(\lambda + \lambda_o(K-1))t}.$$

What may be intuitively expected and is indeed observed regarding the zero-spreadtime *cdf* is that when it is shifted to be tangent to the maximum-copy one, the part of the *cdf* from the contact point and afterwards is a fairly accurate approximation of the original *cdf*.

Based on the above observations, an approximate expression for the *cdf* of the delivery delay, denoted as $\hat{Q}_K(t)$, may be defined as a two-part function, consisting of the maximum-

copy *cdf* until being tangent to the shifted zero-spreadtime one and of the latter afterwards, or

$$\hat{Q}_K(t) = \begin{cases} Q_N(t) = 1 - e^{-\lambda N t} \left(1 + \frac{\lambda}{\lambda_d} (e^{\lambda_d t} - 1)\right)^{N-1}, & 0 \leq t \leq t_{cr}; \\ Q_{\hat{K}}(t - t_0) = 1 - e^{-(\lambda + \lambda_o(K-1))(t-t_0)}, & t \geq t_{cr}, \end{cases}$$

where t_0 denotes the time shift of the zero-spreadtime *cdf* needed to be tangent to the maximum-copy one and t_{cr} corresponds to the contact point of the above two *cdfs*.

By using a second order series expansion of both sides of the equation $Q_N(t_{cr}) = Q_{\hat{K}}(t_{cr} - t_0)$, and by demanding that this equation has only a single solution, since the two *cdfs* should have a single contact point, it is obtained

$$t_0 = \frac{\lambda_o(K-1)^2 N}{2\lambda(\lambda K - \lambda_d(K-1))((N-1)N - K + 1)},$$

$$t_{cr} = \frac{1}{\lambda} \frac{b_1 e^{-b_1 \lambda t_0} - a_1}{2a_2 - b_1^2 e^{-b_1 \lambda t_0}}.$$

Figure 3 illustrates how the approximate *cdf* is constructed for the case of $N = 100$, $K = 8$, $\lambda = 0.08$ and $\lambda_o = 0.04$.

The approximate *cdf* may be used in order to obtain closed form solutions to design problems for which the exact analysis allows only for a numerical solution. For example, given the characteristics of the network and the nodes, which would allow to determine the intermeeting parameters λ and λ_o , it might be desirable to estimate the number of copies K that the source should spread in the network in order to achieve a specific delivery ratio, referred to as Q_d , within some time t . Using the above approximation, the following estimation may be obtained:

$$K_{approx} = \frac{-\lambda_d(N + (N-2)(N+1)\lambda t)}{\lambda_o(N - 2\lambda t)} + \frac{\lambda(N + (N(N-1) - 1)\lambda t + \ln(1 - Q_d)) \pm \sqrt{C}}{\lambda_o(N - 2\lambda t)},$$

where $C = \lambda(2\lambda\lambda_o(N-1)N^2 + (\lambda_d(N-1)N + \lambda(1 - N^2 + N))^2\lambda t) + \lambda \ln(1 - Q_d)(-2\lambda_d(N-1)N(N - \lambda t) + 2\lambda((N-1)N(N - \lambda t) + \lambda t) + \lambda \ln(1 - Q_d))$, and $\lambda_d = \lambda - \lambda_o$. From the two values obtained for K_{approx} , the positive one that fulfils the condition $t_0 \leq t_{cr}$ should be selected. It should be noted that the derived K_{approx} is found to provide a performance close to the targeted one even if it suggests a considerably lower number of copies.

IV. CALCULATION OF THE OVERHEAD

A number of transmissions is required for the message to be delivered to the destination node; this number of transmissions corresponds to the overhead of the message spreading mechanism. Since the setting introduced in this work allows for a differentiation between the source and all other nodes in terms of transmission power, we also consider the overhead in terms of energy consumption.

As mentioned, each message is assumed to have a delay bound (that might be either an application-specific characteristic or a design parameter of the algorithm itself) after which

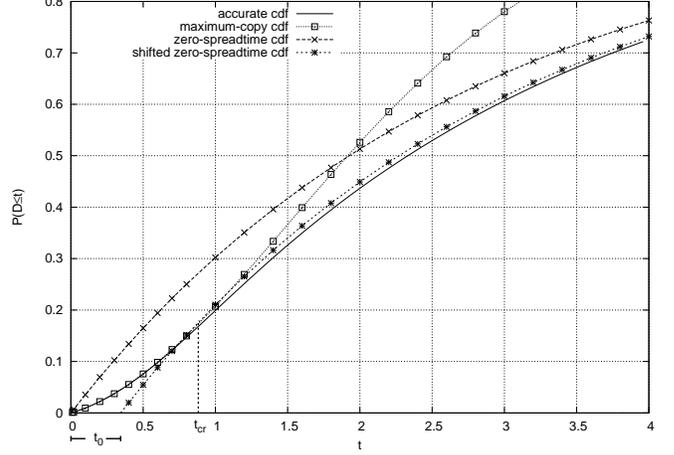


Fig. 3. Approximating the accurate *cdf* with the maximum copy *cdf* for $t \leq t_{cr}$ and the shifted zero-spreadtime *cdf* for $t > t_{cr}$.

it is dropped, no matter if it has been delivered or not to the destination.

The total overhead spent for the delivery of the delay-bounded messages is determined in two phases; first, the overhead until the delivery or drop of the message is extracted (both in terms of transmissions and energy consumption) and then the overhead that might occur after the delivery of the message until the source becomes aware of the message delivery, referred to as additional overhead, is calculated.

A. Overhead until the delivery or drop of the message

The expected overhead consumption until the delivery or drop of the message, $\bar{O}_{del|drop}$, may be expressed as

$$\bar{O}_{del|drop} = \sum_{i=1}^K \bar{O}_{del|drop,i} P_{del|drop,i},$$

where $\bar{O}_{del|drop,i}$ denotes the expected overhead consumption provided that the message is delivered or dropped when in state i and $P_{del|drop,i}$ denotes the probability that the system is in state i when the destination is reached or the message is dropped. Then, for $1 \leq i < K$,

$$P_{del|drop,i} = \left(1 - (1 - P_{drop,i}) \frac{q(i, i+1)}{q_i}\right) \prod_{j=1}^{i-1} (1 - P_{drop,j}) \frac{q(j, j+1)}{q_j},$$

where $P_{drop,i}$ denotes the probability that the message is dropped provided that the system is in state i . For $i = K$, we use the complementary probability

$$P_{del|drop,K} = 1 - \sum_{i=1}^{K-1} P_{del|drop,i}.$$

Now, the probability $(1 - P_{drop,i})$ that the message is not dropped provided that the system is in state i can be expressed

as

$$1 - P_{drop,i} = \begin{cases} \int_0^D f_{D_1}(t)dt, & i=1; \\ \frac{\int_0^D f_{D_i}(t)dt}{\int_0^D f_{D_{i-1}}(t)dt}, & 1 < i \leq K-1, \end{cases}$$

where D_i depicts the total delay until the message reaches the destination provided that the destination gets the message when the system is in state i . More specifically, since the system is in state i , it holds $\sum_{j=1}^{i-1} T_j \leq D$, where T_j denotes the sojourn time in state j . In addition, the message is not dropped only if $\sum_{j=1}^i T_j \leq D$. Furthermore, $\int_0^t f_{D_i}(x)dx = \left(\prod_{j=1}^i q_j\right) f_{2,i}(t)$. Thus,

$$1 - P_{drop,i} = \begin{cases} q_1 f_{2,1}(D), & i=1; \\ \frac{q_i f_{2,i}(D)}{f_{2,i-1}(D)}, & 1 < i \leq K-1. \end{cases}$$

Now, the expected overhead consumption provided that the message is delivered or dropped when the system is in state i , $\bar{O}_{del|drop,i}$ can be expressed as

$$\bar{O}_{del|drop,i} = (i-1)E_s + P_{dels,i}E_s + P_{delo,i}E_o,$$

where $P_{dels,i}$ ($P_{delo,i}$) denotes the probability that the source (some intermediate node) delivers the message to the destination provided that the message is delivered or dropped when the system is in state i . To allow for the calculation of the consumed energy when different power levels are used for the transmissions of the source and the intermediate nodes, the parameters E_s and E_o are used respectively. In order to calculate the overhead only in terms of the number of transmissions, E_s and E_o should be assumed to be equal to 1.

When being in state i , the message is dropped with probability $P_{drop,i}$, it is delivered to the destination by some of the $i-1$ intermediate nodes with probability $(1 - P_{drop,i}) \frac{(i-1)\lambda_o}{q_i}$ and it is delivered to the destination by the source with probability $(1 - P_{drop,i}) \frac{\lambda}{q_i}$. Taking the above into consideration, it is obtained that

$$P_{dels,i} = \frac{(1 - P_{drop,i}) \frac{\lambda}{q_i}}{P_{drop,i} + (1 - P_{drop,i}) \frac{\lambda + (i-1)\lambda_o}{q_i}}, 1 \leq i \leq K,$$

and

$$P_{delo,i} = \frac{(1 - P_{drop,i}) \frac{(i-1)\lambda_o}{q_i}}{P_{drop,i} + (1 - P_{drop,i}) \frac{\lambda + (i-1)\lambda_o}{q_i}}, 1 \leq i \leq K.$$

B. Calculation of the additional overhead

When the message is delivered to the destination by an intermediate node, the source node will continue spreading the rest (if any) of the predefined number of copies (K); the overhead that is unnecessarily spent in this case is referred to as additional overhead. Here, the additional overhead is calculated for two distinct cases: (a) until the source node meets the destination (referred to as *single notification*), and (b) until it meets either the destination or the intermediate node that delivered the message to the destination (referred to as *double notification*).

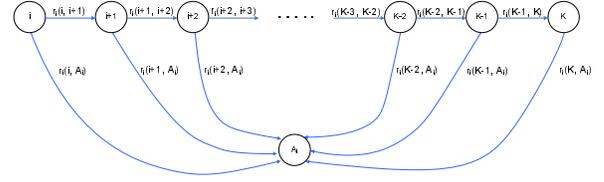


Fig. 4. The Markov chain for the additional overhead calculation.

The single notification may be considered as the straightforward notification procedure of the two-hop relay algorithm, since the source node is obviously expected to end the message copy spreading process after meeting the destination node. On the other hand, the double notification is the simplest notification procedure engaging intermediate nodes and requires that the node which has delivered the message to the destination keep the source in its memory and notify it after their encounter; this procedure is used as an indicative example of how a little more sophisticated mechanism may limit the overhead spent.

The expected additional overhead may be expressed as

$$\bar{O}_{add} = \sum_{i=2}^K P_{del|drop,i} P_{delo,i} \bar{O}_{add,i},$$

where $\bar{O}_{add,i}$ denotes the expected additional overhead provided that the message is delivered by some intermediate node when the system (as modeled by the Markov chain of Fig.1) is in state i . In order to derive $\bar{O}_{add,i}$ the system may be modeled as a Markov chain, depicted in Fig.4, starting from state i (where there are i copies of the message, one for the source node and the rest for the intermediate ones) up to state K and having an absorbing state A_i that corresponds to the case that the source node has been informed of the delivery success.

The transition rates for the Markov chain of Fig.4 are

$$r_i(j,k) = \begin{cases} \lambda(N-j), & j=i, \dots, K-1, k=j+1; \\ \lambda_{n,i} + \lambda_{n,d}, & j=1, \dots, K, k=A_i; \\ 0, & \text{otherwise,} \end{cases}$$

where $\lambda_{n,i}$ and $\lambda_{n,d}$ denote the intermeeting time parameters for the intermediate node and the destination node when they notify the source, respectively. (In the case where this parameter is assumed to be the same as for message transmissions, then $\lambda_{n,i} = \lambda_{n,d} = \lambda_o$.)

The expected additional energy consumption provided that the message is delivered by an intermediate node when the system is in state i , $\bar{O}_{add,i}$, may be expressed as

$$\bar{O}_{add,i} = \sum_{j=i}^K \bar{O}_{add,i,j} P_{del|drop,i,j},$$

where $\bar{O}_{add,i,j}$ denotes the expected overhead provided that the source is notified or the message is dropped when the system is in state j (Fig. 4) and $P_{del|drop,i,j}$ denotes the probability that the system is in state j when the source is

notified or the message is dropped. For $i \leq j < K$, the latter term is given by

$$P_{del|drop,i,j} = \left(1 - (1 - P_{drop,i,j}) \frac{r_i(j, j+1)}{r_{i,j}} \right) \prod_{k=1}^{j-1} (1 - P_{drop,i,k}) \frac{r_i(k, k+1)}{r_{i,k}},$$

where $P_{drop,i,j}$ denotes the probability that the message is dropped provided that the system is in state j and $r_{i,j} = r_i(j, j+1) + r_i(j, A_i)$. (In case $j = K$, there is no additional overhead spent.)

Now, the message will not be dropped when the system is in state j if and only if $D_{i,j} = \sum_{k=1}^i T_k + \sum_{k=i}^j T_{i,k} \leq D$, where T_k denotes the sojourn time in state k of the chain in Fig.1 and $T_{i,k}$ denotes the sojourn time in state k of the chain in Fig.4. Provided that the system is in state j in Fig.4, it holds that $D_{i,j-1} = \sum_{k=1}^i T_k + \sum_{k=i}^{j-1} T_{i,k} \leq D$. Thus,

$$1 - P_{drop,i,j} = \begin{cases} F_{D_{i,1}}(D), & j=1; \\ \frac{F_{D_{i,j}}(D)}{F_{D_{i,j-1}}(D)}, & 1 < j \leq K-1, \end{cases}$$

with

$$F_{D_{i,j}}(t) = \int_0^t f_{D_{i,j}}(x) dx,$$

where $f_{D_{i,j}}(t)$ denotes the *pdf* of $D_{i,j}$. $F_{D_{i,j}}(t)$ is derived in the Appendix.

The term $\bar{O}_{add,i,j}$ may be expressed as

$$\bar{O}_{add,i,j} = (j-1)E_s + P_{nd,i,j}E_{nd} + P_{ni,i,j}E_{ni},$$

where $P_{nd,i,j}$ ($P_{ni,i,j}$) denotes the probability that the source is notified by the destination (the intermediate node that delivered the message) provided that the source is notified or the message is dropped when being in state j ; E_{nd} and E_{ni} denote the energy consumed for the transmission of the notification message by the destination node and an intermediate node, respectively.

Finally, the following equations may be used:

$$P_{nd,i,j} = \frac{(1 - P_{drop,i,j}) \frac{\lambda_{n,d}}{r_{i,j}}}{P_{drop,i,j} + (1 - P_{drop,i,j}) \frac{\lambda_{n,i} + \lambda_{n,d}}{r_{i,j}}},$$

and

$$P_{ni,i,j} = \frac{(1 - P_{drop,i,j}) \frac{\lambda_{n,i}}{r_{i,j}}}{P_{drop,i,j} + (1 - P_{drop,i,j}) \frac{\lambda_{n,i} + \lambda_{n,d}}{r_{i,j}}}.$$

It should be noted that for single notification $\lambda_{n,i} = 0$; thus, $P_{ni,i,j}$ becomes equal to zero.

Figures 5 and 6 depict the number of transmissions and the energy consumed, respectively, as a function of λ , for the case of $K = N = 100$ and $\lambda_o = 0.08$. (The energy consumed for a transmission is assumed to be proportional to the square of λ and equal to 1 for $\lambda = 0.08$.) As expected, there is a significant difference between the number of transmissions and the energy consumed in heterogeneous networks where different transmission powers among the nodes may be employed.

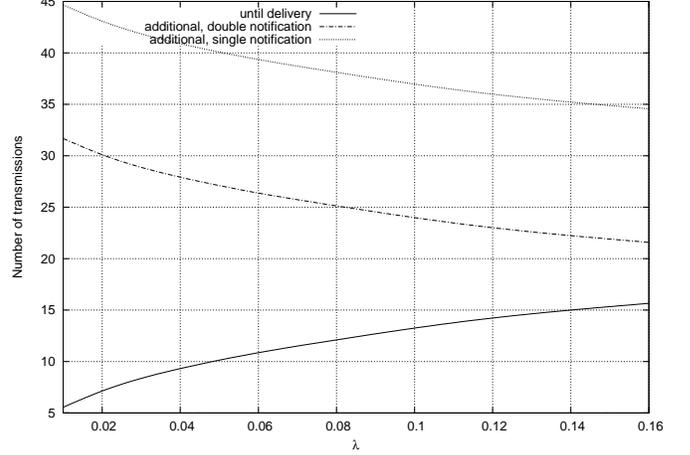


Fig. 5. The number of transmissions as a function of λ , where $K = N$ and $\lambda_o = 0.08$.

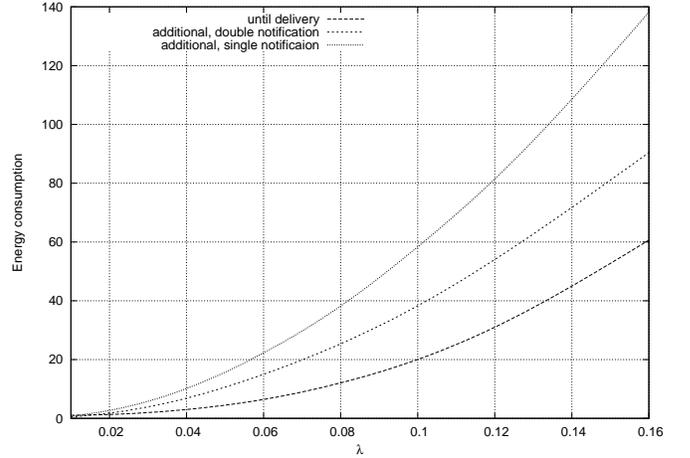


Fig. 6. The energy consumption as a function of λ , where $K = N$ and $\lambda_o = 0.08$.

By both figures 5 and 6, it may be concluded that the number of transmissions or consumed energy until the message delivery time are just a small portion of the corresponding totals and cannot be ignored. In addition, the participation of the intermediate node in notifying the source node of the message delivery (double notification) limits noticeably the additional energy spent; consequently, it seems that even a simple approach as the one introduced in this work can be proved to be a valuable mechanism.

V. CONCLUSIONS

In this work, we investigated the performance of the two-hop message spreading mechanism within the framework of a DTN environment, where differentiation on the characteristics between the source and the intermediate nodes – either intentionally (adjusting transmission range, changing mobility parameters, misbehaving), or unintentionally (buffer or antenna limitations) – could exist. Analytical expressions for the

message delivery ratio as well as the number of transmissions and consumed energy were derived. An approximation for the delivery ratio was also introduced that fairly accurately captures the performance of the two-hop relay algorithm. It was shown that the consumed energy or needed transmissions after the delivery of the message might be significant. In order to limit the unnecessary transmissions, even a simple message delivery notification procedure was shown to be a valuable mechanism when the number of copies to be spread in the network is high.

VI. ACKNOWLEDGMENT

This work has been supported in part by the: i) IST BIONETS program under contract FP6-027748; ii) PENED project that is co-financed by E.U.-European Social Fund (75%) and the Greek Ministry of Development-GSRT (25%); and iii) PYTHAGORAS II: Support of Universities' research groups, co-funded by the Operational Programme for Education and Initial Vocational Training (O.P. "Education") and the European Social Funds.

VII. APPENDIX

A. Derivation of $f_{2,i}(t)$

A.1 For $1 \leq i < K$ the poles of $F_{2,i}(s)$ are $s_{i,0}=0$ and $s_{i,k} = \lambda_d(k-1) - \lambda N, k = 1, \dots, i$, which are all single poles. Thus,

$$f_{2,i}(t) = \sum_{k=0}^i \bar{c}_{i,k} e^{s_{i,k} t},$$

where $\bar{c}_{i,k} = (s - s_{i,k}) F_{2,i}(s)|_{s=s_{i,k}}$. For $0 < k \leq K$,

$$\bar{c}_{i,k} = \frac{1}{\lambda_d(k-1) - \lambda N} c_{i,k},$$

where

$$\begin{aligned} c_{i,k} &= \prod_{j=1, j \neq k}^i \frac{1}{\lambda N - \lambda_d(j-1) + \lambda_d(k-1) - \lambda N} \\ &= \frac{1}{\lambda_d^{i-1}} \prod_{j=1, j \neq k}^i \frac{1}{k-j} = \frac{1}{\lambda_d^{i-1}} \frac{1}{(k-1)!} \frac{(-1)^{i-k}}{(i-k)!}. \end{aligned}$$

For $k = 0$,

$$\begin{aligned} \bar{c}_{i,0} &= \prod_{j=1}^i \frac{1}{\lambda N - \lambda_d(j-1)} = \sum_{k=1}^i \frac{1}{\lambda N - \lambda_d(k-1)} \\ &\quad \prod_{j=1, j \neq k}^i \frac{1}{\lambda_d(k-1) - \lambda_d(j-1)} = \sum_{k=1}^i -\bar{c}_{i,k}. \end{aligned}$$

Thus,

$$\begin{aligned} f_{2,i}(t) &= \sum_{k=1}^i \bar{c}_{i,k} (e^{-\lambda N t} e^{\lambda_d(k-1)t} - 1) \\ &= \sum_{k=1}^i \frac{1}{\lambda_d(k-1) - \lambda N} c_{i,k} (e^{-\lambda N t} e^{\lambda_d(k-1)t} - 1) \\ &= \sum_{k=1}^i \frac{1}{\lambda_d(k-1) - \lambda N} \frac{1}{\lambda_d^{i-1}} \frac{1}{(k-1)!} \\ &\quad \frac{(-1)^{i-k}}{(i-k)!} (e^{-\lambda N t} e^{\lambda_d(k-1)t} - 1). \end{aligned}$$

A.2 For $i = K$ the poles of $F_{2,K}(s)$ are $s_{K,0} = 0, s_{K,K} = -(\lambda + (\lambda - \lambda_d)(K-1)) = (K-1)\lambda_d - K\lambda$ and $s_{K,k} = \lambda_d(k-1) - \lambda N, k = 1, \dots, K-1$. The above poles are all single for $-(\lambda + (\lambda - \lambda_d)(K-1)) \neq \lambda_d(j-1) - \lambda N, \forall j, 1 \leq j \leq K-1$ (it can be easily concluded that $\lambda(N-K) + j\lambda_d \neq 0, \forall j, 1 \leq j \leq K-1$, so none of the following denominators is zero), $s_{K,K} \neq 0$ and $s_{K,k} \neq 0, k = 1, \dots, K-1$. The case of poles with multiplicity greater than 1 is not considered in order to avoid unnecessary complexity; however, in the numerical results the aforementioned condition is technically never met by adding some "noise" digits. Thus,

$$f_{2,K}(t) = \sum_{k=0}^K \tilde{c}_{K,k} e^{s_{K,k} t},$$

where $\tilde{c}_{K,k} = (s - s_{K,k}) F_{2,K}(s)|_{s=s_{K,k}}$. For $0 < k < K$,

$$\begin{aligned} \tilde{c}_{K,k} &= \frac{1}{\lambda_d(k-1) - \lambda N} \\ &\quad \frac{1}{\lambda + (\lambda - \lambda_d)(K-1) + \lambda_d(k-1) - \lambda N} \\ &\quad \prod_{j=1, j \neq k}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1) + \lambda_d(k-1) - \lambda N} \\ &= \frac{1}{\lambda_d(k-1) - \lambda N} \frac{-1}{\lambda(N-K) + \lambda_d(K-k)} c_{K-1,k} \\ &= \frac{1}{\lambda_d(k-1) - \lambda N} \hat{c}_{K,k}. \end{aligned}$$

where

$$\begin{aligned} \hat{c}_{K,k} &= \frac{1}{\lambda + (\lambda - \lambda_d)(K-1) + \lambda_d(k-1) - \lambda N} \\ &\quad \prod_{j=1, j \neq k}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1) + \lambda_d(k-1) - \lambda N} \\ &= \frac{-1}{\lambda(N-K) + \lambda_d(K-k)} c_{K-1,k}. \end{aligned}$$

For $k = K$,

$$\begin{aligned} \tilde{c}_{K,K} &= \frac{1}{\lambda_d(K-1) - \lambda K} \\ &\quad \prod_{j=1}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1) + (K-1)\lambda_d - K\lambda} \\ &= \frac{1}{\lambda_d(K-1) - \lambda K} \hat{c}_{K,k}, \end{aligned}$$

where

$$\begin{aligned}\widehat{c}_{K,K} &= \prod_{j=1}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1) + (K-1)\lambda_d - K\lambda} \\ &= \prod_{j=1}^{K-1} \frac{1}{\lambda(N-K) + j\lambda_d} \\ &= \sum_{k=1}^{K-1} \frac{1}{\lambda(N-K) + k\lambda_d} \prod_{j=1, j \neq k}^{K-1} \frac{1}{\lambda_d j - \lambda_d k} \\ &= \sum_{k=1}^{K-1} \frac{1}{\lambda(N-K) + k\lambda_d} \frac{1}{\lambda_d^{K-2}} \frac{(-1)^{k-1}}{(k-1)!} \frac{1}{(K-k-1)!}.\end{aligned}$$

For $k=0$,

$$\begin{aligned}\widetilde{c}_{K,0} &= \frac{1}{\lambda + (\lambda - \lambda_d)(K-1)} \prod_{j=1}^{K-1} \frac{1}{\lambda N - \lambda_d(j-1)} \\ &= \frac{1}{\lambda K - \lambda_d(K-1)} \\ &\sum_{k=1}^{K-1} \frac{1}{\lambda N - \lambda_d(k-1)} \frac{1}{\lambda_d^{K-2}} \frac{1}{(k-1)!} \frac{(-1)^{K-k-1}}{(K-k-1)!}.\end{aligned}$$

Thus,

$$\begin{aligned}f_{2,K}(t) &= \sum_{k=0}^K \widetilde{c}_{K,k} e^{s_{K,k}t} = \frac{1}{\lambda K - \lambda_d(K-1)} \\ &\sum_{k=1}^{K-1} \frac{1}{\lambda N - \lambda_d(k-1)} \frac{1}{\lambda_d^{K-2}} \frac{1}{(k-1)!} \frac{(-1)^{K-k-1}}{(K-k-1)!} \\ &+ \frac{1}{\lambda_d^{K-2}} \sum_{k=1}^{K-1} \frac{1}{(k-1)!} \frac{1}{(K-1-k)!} \\ &\left(\frac{(-1)^{k-1} e^{((K-1)\lambda_d - K\lambda)t}}{(\lambda_d(K-1) - \lambda K)(\lambda(N-K) + k\lambda_d)} \right. \\ &\left. + \frac{(-1)^{K-k} e^{-\lambda N t} e^{\lambda_d(k-1)t}}{(\lambda_d(k-1) - \lambda N)(\lambda(N-K) + \lambda_d(K-k))} \right).\end{aligned}$$

B. Derivation of $F_{D_{i,j}}(t)$

$F_{D_{i,j}}(t)$ is derived starting from

$$\mathcal{L}(F_{D_{i,j}}(t)) = \frac{1}{s} \left(\prod_{k=1}^i \frac{q_k}{s + q_k} \right) \left(\prod_{k=i}^j \frac{r_{i,k}}{s + r_{i,k}} \right).$$

The poles $\mathcal{L}(F_{D_{i,j}}(t))$ are

$$s_{i,j,k} = \begin{cases} 0, & k=0; \\ -q_k, & 1 \leq k \leq i-1; \\ -r_{i,k}, & i \leq k \leq j; \\ -q_i, & k=j+1. \end{cases}$$

(We consider $-q_i$ as the $j+1$ pole.) We consider only the case where the multiplicity of each pole is equal to one; this

requires that:

$$\begin{aligned}q_k &\neq q_{k'}, & 1 \leq k, k' \leq i, k \neq k'; \\ r_{i,k} &\neq r_{i,k'}, & i \leq k, k' \leq j, k \neq k'; \\ q_k &\neq r_{i,k'}, & 1 \leq k \leq i, i \leq k' \leq j; \\ q_k &\neq 0, & 1 \leq k \leq i; \\ r_{i,k} &\neq 0, & i \leq k \leq j.\end{aligned}$$

(In the numerical results it is ensured that these conditions are met by using some additional ‘‘noise’’ digits for the values of the parameters.) Then,

$$F_{D_{i,j}}(t) = \sum_{k=0}^{j+1} c_{i,j,k} e^{s_{i,j,k}t},$$

where

$$c_{i,j,k} = (s - s_{i,j,k}) \mathcal{L}(F_{D_{i,j}}(t)) \big|_{s=s_{i,j,k}}, \quad 0 \leq k \leq j+1.$$

Thus,

$$c_{i,j,k} = \begin{cases} 1, & k=0; \\ - \left(\prod_{m=1}^{k-1} \frac{q_m}{q_m - q_k} \right) \left(\prod_{m=k+1}^{i-1} \frac{q_m}{q_m - q_k} \right) \\ \quad \left(\prod_{m=i}^j \frac{r_{i,m}}{r_{i,m} - q_k} \right), & 1 \leq k \leq i-1; \\ - \left(\prod_{m=1}^{i-1} \frac{q_m}{q_m - r_{i,k}} \right) \left(\prod_{m=i}^{k-1} \frac{r_{i,m}}{r_{i,m} - r_{i,k}} \right) \\ \quad \left(\prod_{m=k+1}^j \frac{r_{i,m}}{r_{i,m} - r_{i,k}} \right), & i \leq k \leq j; \\ - \left(\prod_{m=1}^{i-1} \frac{q_m}{q_m - q_i} \right) \left(\prod_{m=i}^j \frac{r_{i,m}}{r_{i,m} - q_i} \right), & k=j+1. \end{cases}$$

REFERENCES

- [1] A. Vahdat and D. Becker, ‘‘Epidemic routing for partially connected ad-hoc networks,’’ Duke Technical Report CS-2000-06, 2000.
- [2] T. Spyropoulos, K. Psounis, and C. Raghavendra, ‘‘Single-copy routing in intermittently connected networks,’’ in *IEEE SECON*, October 2004.
- [3] E. Jones and P. Ward, ‘‘Routing strategies for delay-tolerant networks,’’ submitted to Computer Communication Review, available at <http://www.ccong.uwaterloo.ca/~pasward/Publications/dtn-routing-survey.pdf>.
- [4] T. Spyropoulos, K. Psounis, and C. Raghavendra, ‘‘Spray and wait: An efficient routing scheme for intermittently connected mobile networks,’’ in *Proceedings of SIGCOMM 2005*, August 2005.
- [5] A. Lindgren, A. Doria, and O. Schelen, ‘‘Probabilistic routing in intermittently connected networks,’’ in *The First International Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR 2004)*, 2004.
- [6] J. Byers and G. Nasser, ‘‘Utility-based decision-making in wireless sensor networks,’’ Tech. Rep. 2000-014, 1 2000.
- [7] M. Musolesi, S. Hailes, and C. Mascolo, ‘‘Adaptive routing for intermittently connected mobile ad hoc networks,’’ in *Proceedings of IEEE WoWMoM*, 2005, pp. 183–189.
- [8] W. Zhao, M. Ammar, and E. Zegura, ‘‘A message ferrying approach for data delivery in sparse mobile ad hoc networks,’’ in *Proceedings of ACM Mobihoc*, 2004.
- [9] —, ‘‘Controlling the mobility of multiple data transport ferries in a delay tolerant network,’’ in *Proceedings of IEEE Infocom*, 2005.
- [10] M. Grossglauser and D. Tse, ‘‘Mobility increases the capacity of ad hoc wireless networks,’’ in *IEEE/ACM Transactions on Networking*, 2002, vol. 10.
- [11] R. Groenevelt, ‘‘Stochastic models in mobile ad hoc networks,’’ Ph.D. dissertation, University of Nice Sophia Antipolis, April 2005.