Single-Copy Routing in Intermittently Connected Mobile Networks

Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra
Department of Electrical Engineering, University of Southern California
Los Angeles, California 90089–2562
Email: {spyropou, kpsounis, raghu}@usc.edu

***Abstract*— Intermittently connected mobile networks are wireless networks where most of the time there does not exist a complete path from source to destination, or such a path is highly unstable and may break soon after it has been discovered. In this context, conventional routing schemes would fail.**

To deal with such networks we propose the use of an opportunistic hop-by-hop routing model. According to the model, a series of independent, local forwarding decisions are made, based on current connectivity and predictions of future connectivity information diffused through nodes' mobility. The important issue here is how to choose an appropriate next hop. To this end, we propose and analyze via theory and simulations a number of routing algorithms. The champion algorithm turns out to be one that combines the simplicity of a simple random policy, which is efficient in finding good leads towards the destination, with the sophistication of utility-based policies that efficiently follow good leads. We also state and analyze the performance of an oracle-based optimal algorithm, and compare it to the online approaches. The metrics used in the comparison are the average message delivery delay and the number of transmissions per message delivered.

I. Introduction

Intermittently connected mobile networks (ICMN) are mobile wireless networks where, at any time (or most of the time), there does not exist a complete path from a source to a destination or such a path is highly unstable and may change or break soon after it has been discovered (or even while being discovered). This situation arises when the network is quite sparse, in which case it can be viewed as a set of disconnected clusters of nodes. Due to node mobility, clusters may change over time. Additionally, extreme mobility may render a complete path discovery highly inefficient, if not useless. Intermittently connected mobile networks belong to the general category of Delay Tolerant Networks [1], that is, networks were incurred delays can be very large and unpredictable.

Since in the ICMN model there may not exist an end-to-end path between a source and a destination, conventional mobile ad-hoc network routing routing schemes, such as DSR [2], AODV [3], etc., would fail. Specifically, reactive schemes will fail to discover a complete path, while proactive protocols will fail to converge, resulting in a deluge of topology update messages. However, this does not mean that packets can never be delivered in such networks. Over time, different links come up and down due to node mobility. If the sequence of connectivity graphs over a time interval are overlapped, then an end-to-end path might exist. This implies that a message could be sent over an existing link, get buffered at the next hop until the next link in the path comes up, and so on and so forth, until it reaches its destination.

This approach imposes a new model for routing. Routing consists of a sequence of independent, local forwarding decisions, based on current connectivity information and predictions of future connectivity information. In other words, node mobility needs to be exploited in order to deliver a message to its destination. This is reminiscent of the work in [4]. However, there mobility is exploited in order to improve capacity, while here it is used to overcome the lack of end-to-end connectivity.

Hop-by-hop routing implies that a packet gets buffered in an intermediate (relay) node, until an ”appropriate” next hop is found for the message to be forwarded. The important issue here is what constitutes an appropriate next hop. In other words, how can a node, currently carrying a packet for a specific destination, make a forwarding decision that will bring the packet ”closer” to the destination? Of greatest importance for such a forwarding decision is going to be how appropriateness of a node as a potential next hop will be defined.

Depending on the number of copies of a single message that may coexist in the network, one can define two major categories of hop-by-hop routing schemes, namely *single-copy* routing schemes and *multiple-copy* routing schemes. In single-copy routing schemes there's only a single custodian for each message. When the current custodian forwards the copy to an appropriate next hop, this becomes the message's new custodian, and so on and so forth until the message reaches its destination. On the other hand, multiple-copy routing schemes may generate multiple copies of the same message which can be routed independently for increased efficiency and robustness. The majority of routing schemes proposed in the literature in the context of ICMNs are flooding or gossip-based, and, therefore, multiple-copy in nature [5], [6], [7], [8], [9].

Despite their increased robustness, flooding-based protocols like epidemic routing and its derivatives consume a high amount of bandwidth and energy, as has been noted in [6], [8], [10]. Thus, a very high cost may be the price for the anticipated performance gain, or, even worse, these schemes may result in poor performance due to high contention for shared resources [11]. These shortcomings may render such algorithms prohibitive for energy-constrained and bandwidth-constrained applications, which is the common case in wireless networks. Additionally, as the average node degree increases, they are faced with important scalability issues, both in terms of memory size needed [8] (a scarce resource in COTS sensors), and amount of transmissions performed [10]. Consequently, it is highly desirable to design efficient single-copy routing schemes for resource-constrained ICMNs.

In this work we investigate the problem of efficient routing in intermittently connected mobile networks using single-copy approaches. (In [10] we study the same problem using multi-copy approaches.) We assume that the only information available to each node regarding other nodes is a set of timers recording the time elapsed since every other node was last encountered. These timers carry indirect location information about a node, which gets diffused into the network through other nodes' mobility. To capture the amount of useful information contained in these timers it is natural to maintain some kind of a utility function for each node (on a per-destination basis) that would, at all times, reflect the probability that a node will deliver the packet to the destination. We define such a utility function and propose a utility-based routing scheme, based on it. We compare its performance, both analytically and using simulations, to that of a simple randomized routing algorithm. Then, we propose a hybrid routing protocol, called "seek and focus", that combines the best features of both the random and utility-based strategies. Our simulations show that this hybrid scheme presents the best tradeoff in terms of message delivery delay and number of transmissions per message delivered. Finally, we derive and analyze an oracle-based optimal algorithm, and compare its performance to that of the online algorithms.

In the next section we go over some existing related work. Then, in Section III, we present our basic assumptions about the problem in hand, and in Section IV we present the random, the utility-based, and the "seek and focus" routing algorithms. Section V introduces the oracle-based optimal algorithm, and analyzes the performance of this and the random routing scheme. Then, Section VI presents simulation results where the performance of all the strategies is compared with respect to message delivery delay and number of transmissions per message delivered. Finally, Section VII concludes the paper and gives some directions for future work.

II. Related Work

In the context of routing for intermittently connected mobile networks a number of efforts exist that mostly try to deal with application-specific problems, especially in the field of sensor networks. In [12] a number of mobile nodes, modeled as performing independent random walks, serve as *DataMules* that carry data from static sensors to base stations, in a sparse sensor network. The statistics of random walks are used to analyze the expected performance of the system. The idea of carrying data through disconnected parts using a virtual mobile backbone has also been used in [7], [9], [13], [14].

In a number of other works, all nodes are assumed to be mobile and algorithms to transfer messages from any node to any other node are sought for [4], [5], [6], [8], [14], [15], [16], [17]. In [5] the concept of epidemic algorithms is applied to routing, as a flooding method in the context of intermittently connected mobile networks. In [8] epidemic routing is used to reliably collect data from sensor nodes attached to zebras. Additionally, a simple method to take advantage of the history of past encounters is implemented in order to reduce the overhead of epidemic routing and improve its performance. The concept of history based or probabilistic routing is further elaborated in [6]. There it is shown that using the age of last encounter with a node, when making a forwarding decision, results in superior performance than flooding. Similar results have been found in the context of regular, connected, wireless networks in [16].

The authors in [15] generalize the concept of history-based/probabilistic routing to that of utility-based routing, where the utility of a node for a destination may be a function of the history of encounter, frequency of encounter, node speed, future or scheduled encounter, node resources, etc. Additionally, they propose a periodic request and reply method, similar to that used in DSR [2] and [16], to query potential next hops, instead of the encounter-based approach of [5], [6], [8].

Despite the variety of existing approaches, a majority of them is based on epidemic-routing or some other form of controlled flooding [5], [6], [7], [8], [9]. Hence, they are multiple-copy schemes, in nature. Furthermore, the minority that deal with single-copy techniques only study utility-based schemes, and provide no theoretical analysis of their performance [15]. To the best of our knowledge, the only prior analytical work is on direct transmission schemes [12]. In this paper, we propose a number of different single-copy routing algorithms, and evaluate their performance both through simulation and analysis. Additionally, we describe an “oracle-base” optimal algorithm, that achieves the minimum delivery delay among all possible single and multiple copy schemes, and analyze its performance. On top of that, we introduce a hybrid single-copy routing algorithm, that is shown to achieve the best performance among all existing and proposed single-copy schemes.

III. Preliminaries

Unless mentioned otherwise, we make the following assumptions regarding our problem setting.

- A.i M nodes perform independent random walks on an $N \times N$ 2D torus (finite lattice). The random walk model is primarily chosen for its analytical tractability. Simulation results are also given for the random waypoint model [18], and, in future work, we plan to experiment with additional mobility models [19], [20].
- A.ii Each node can transmit up to $K > 0$ grid squares away. K/N is much smaller than the value required to guarantee connectivity with high probability [21]. We assume that transmission of a message is faster than movement, and can occur in parallel with the latter.
- A.iii We use Manhattan distance $d_{ab} = |a_x - b_x| + |a_y - b_y|$ to measure proximity between two positions a and b (or between two nodes).
- A.iv Every node i maintains a timer $\tau_i(j)$ for every other node j it has encountered. These timers are the only information available to a node regarding the network (i.e. no location info, speed, etc.). We assume that two nodes “encounter” each other when they come within transmission range of each other. The timers are maintained as follows: Initially all $\tau_i(j)$ are set to ∞. When node i encounters node j set $\tau_i(j)$ to 0. At every time unit elapsed increase $\tau_i(j)$ by 1.
- A.v Let m be a message that originates at some node S and needs to be delivered to some node D. Furthermore, let $R_m(t)$ denote the set of nodes carrying message m at time t. All nodes follow either a single-copy or a multiple-copy forwarding strategy, according to the following definition:
  - *single-copy* strategy: $\|R_m(t)\| \leq 1$, $\forall t, m$.
  - *multiple-copy* strategy: $\|R_m(t)\| \leq M$, $\forall t, m$.

  In this paper, we consider only single-copy strategies and discuss multiple-copy schemes in [10].

We will use the following metrics to evaluate different routing algorithms: i) *message delivery delay*, and ii) *number of transmissions per message delivered*.

Finally, we note that most of the times we use the same notation as in [22]. However, we define here a couple of hitting-time related quantities that we will use repeatedly.

Definition 3.1 (Expected Hitting Time): We define the following expected hitting times (where j can be replaced by a subset of states A):

i. $E_i T_j$: the expected hitting time until a walk starting at position i first arrives at position j; on a symmetric graph, this quantity only depends on d_{ij} and we denote it as $ET(d)$.
ii. $E_\pi T_j$: the expected hitting time until a walk starting from the stationary distribution reaches j; on a symmetric graph, this quantity is independent of j, and we denote it as ET.
iii. EM_{ij} the expected time until two independent random walks, starting at positions i and j, respectively, first meet each other.
iv. EM the expected time until two independent random walks, starting from the stationary distribution, first meet each other.

Throughout Section IV, we shall be mainly referring to the hitting time of a single walk. However, on a symmetric graph (like the 2D torus), the expected meeting time is just half the respective hitting time [22].

IV. Single Copy Routing Strategies

In this section we describe a number of single-copy routing algorithms. Each routing algorithm will decide under what circumstances a node, currently holding the single message copy, will handover a message to another node it encounters. Each forwarding step should, on the average, result in progress of the message towards its destination, measured as a reduction in the distance from or expected meeting time with the destination. Due to lack of space, we omit all the proofs in this section, and give only some intuition or sketches of proofs, instead. The interested reader can find all proofs in [23].

A. Direct Transmission

The simplest possible scheme imaginable is the following: *a node A forwards a message to another node B it encounters, only if B is the message's destination*. This scheme has an unbounded delivery delay [4], but has the advantage of performing only a single transmission per message. It has been considered in some previous works [4], [12], and will serve here as our baseline for comparison.

B. Randomized Routing Algorithm

The first non-trivial routing algorithm that we'll look at is a randomized forwarding algorithm, which may use relays to deliver a message to its destination.

Definition 4.1: In the *randomized routing algorithm p*, a node A hands over a message to another node B it encounters with probability $p > 0$.

Our first result, captured in Theorem 4.1, states that even this simple routing strategy results in expected progress at each forwarding step (i.e. locally).

Theorem 4.1: Let a node A carrying a message for some node D at distance $d_{AD} = d$ from A, encounter another node B at distance d_{AB}. The expected progress made by the randomized forwarding algorithm is equal to G_1, when $d_{AB} = 1$, and at least G_1, when $d_{AB} > 1$, where G_1 is given by

$$G_1 = p\left[ET(d) - \frac{ET(d-1) + ET(d+1)}{2}\right] - o(N). \quad (1)$$

This result is slightly counterintuitive, but can be explained by the fact that the expected hitting time $ET(d)$ is a concave monotonically increasing function of distance d [22], [24], and transmission speed is faster than speed of movement.

Nevertheless, this progress is marginal, especially when far from the destination, as can be seen in Fig. 2. This scheme does not take advantage of the only information available to each node regarding the network, that is, the timers containing the time since last encounter with every other node.

C. Utility-based Routing

Position information regarding different nodes gets indirectly logged in the last encounter timers, and gets diffused around it though the mobility process of other nodes. This position information is not absolute, but rather relative to the position of another node. If a node is seen at some time instant having a low timer value for another node, then this other node is expected to be somewhere nearby.

However, the amount of useful information contained in the timer values strongly depends on the speed of movement of nodes and the specific mobility model assumed. Therefore, we would like to define a utility function $U_X(Y)$, maintained by each node for every other node that indicates how useful node X might be in delivering a message to a node Y. A gradient-based scheme can then be used to deliver a message to its destination, as has been noted in [6], [15], [16]. This scheme will try to maximize the utility function for this destination.

We define a pure utility-based routing protocol as follows:

Definition 4.2 (Pure Utility-based Routing): When a pure utility based routing strategy is used, a node A forwards to another node B a message destined to a node D, if and only if $U_B(D) > U_A(D)$, and $\forall X : U_D(D) \geq U_X(D)$.

Before we give the expected local progress by a forwarding step of this scheme, we state the following Lemma that connects last encounter timer values with expected proximity to a node.

Lemma 4.1: Let two nodes A and B have a recorded age of last encounter with node D equal to $\tau_A(D)$ and $\tau_B(D)$, respectively. Furthermore, let $1 \leq d_{AB} \leq K$, and $\tau_B(D) = \tau_A(D) - d\tau$, for some $d\tau > 0$. Finally, let $P_{BA} = P\{d_{BD} < d_{AD} | d_{AB}, d\tau\}$ denote the probability that B is closer to D than A is to D, given their distance and that B has a lower timer value for D than A. Then

$$P_{BA} \leq 1 - P_{BA}.$$

Fig. 1 depicts P_{BA} versus $(1 - P_{BA})$ for two different values of d_{AB} and $d\tau$.

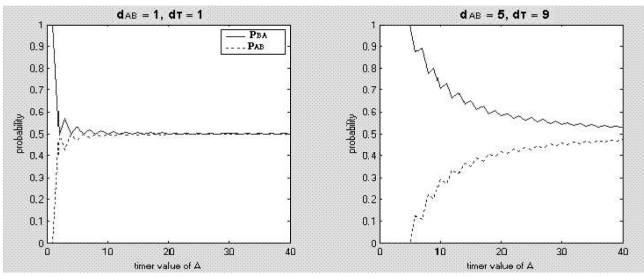

Fig. 1. Probability P_{BA} as a function of A's timer value $\tau_A(D)$, for different d_{AB} and $d\tau$.

We can now state the following result concerning the efficiency of a pure utility-based routing strategy:

Theorem 4.2: A pure utility-based routing protocol, which uses any utility function $U_X(Y)$ that is a monotonic function of $\tau_X(Y)$, guarantees a positive expected progress at each forwarding step, which is larger than that of the randomized routing algorithm p, and for neighboring nodes ($d_{AB} = 1$) is given by:

$$G_2 = ET(d) - [P_{BA}ET(d-1) + (1 - P_{BA})ET(d+1)]. \quad (2)$$

In Fig. 2, we compare the expected progress of the randomized algorithm and the utility-based algorithm, measured as a reduction in the expected hitting time, when handing a message over from A to B.

As mentioned in Theorem 4.2 any monotonic utility function of the timer $\tau_X(Y)$ will have the same expected

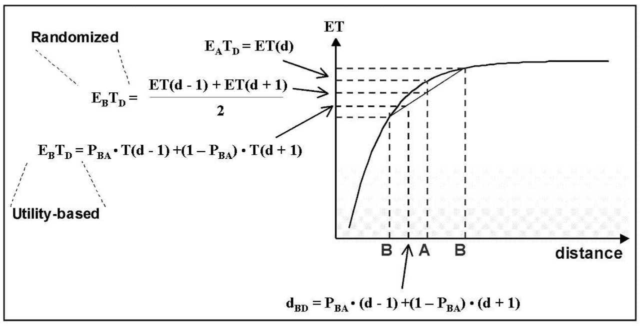

Fig. 2. Expected progress for the randomized and utility-based forwarding policies, measured as a reduction in the expected hitting time to the destination.

progress at every forwarding step. However, there are situations where the anticipated progress by a specific forwarding decision will have to justify the transmission cost (or other costs). In those cases, we would like to have a utility function that accurately quantifies the potential gain of transmission, so that it can be compared to the cost to be incurred, before making a decision.

We propose such a utility function that directly takes into account the statistics of the specific mobility process, and quantifies the exact expected reduction in hitting time by a specific forwarding decision. Specifically, let $E_j T_j^+$ denote the *first return time*, that is, the time until a walk starting at state/position j first returns back to j. Then, we propose the utility function $U_X(Y)$ to be defined as *the expected hitting time of X on Y, given that X has not seen Y for $\tau_X(Y)$ time units*:

$$U_X(Y) = E_X \left[T_Y | E_Y T_Y^+ \geq \tau_X(Y) \right].$$

The above quantity is only a function of $\tau_X(Y)$, and therefore can be calculated by any node, with no additional information.

D. The Seek and Focus Routing Protocol - A Hybrid Approach

We saw that, in the randomized routing protocol case, handing over a packet to a neighbor is better than holding it. The intuition behind this is that transmissions are faster than physical movement. However, this results in a large number of wasted transmissions. On the other hand, utility-based routing protocols make better forwarding decisions by taking advantage of indirect location information. Nevertheless, utility-based protocols suffer from a *slow start* initial phase, which is more manifested in large networks. Specifically, in a large network, where expected distance between a source and a destination is large, it will take the source a long time until it finds a higher utility next hop at the beginning. In other words, it will have to wait until it moves within a certain vicinity around the destination, where diffused location information has managed to propagate.

We have therefore implemented a hybrid routing protocol that aims to avoid the slow-start phase of utility-based protocols, while still taking advantage of the higher efficiency of utility-based forwarding. We call it the "seek and focus" routing protocol, due to its going through the following two phases.

Definition 4.3 (Seek and Focus Algorithm (Hybrid)): The seek and focus routing algorithm consists of two phases: (*seek phase*) if the utility around the node is low, perform randomized forwarding with parameter p to quickly search nearby nodes; (*focus phase*) when a high utility node (i.e. above a pre-specified threshold) is discovered, switch to utility-based forwarding.

This scheme initially looks around greedily for a *good lead* towards the destination, and then uses a utility-based approach to follow that lead efficiently.

V. Performance Analysis

A. Direct Transmission – An Upper Bound on Delay

We first state here some useful, known, results regarding the expected hitting time of a single random walk, and meeting times of independent random walks on a $\sqrt{N} \times \sqrt{N}$ torus [22], [24]. These results also give the performance of the direct transmission scheme, which will serve as our baseline.

Lemma 5.1: Let independent random walks be performed on a $\sqrt{N} \times \sqrt{N}$ torus. Then:

i. $ET = cN \log N$, where $c = 0.34$. (This results is valid as $N \to \infty$. However, according to [24], [12] this results becomes quite accurate for $N > 25$.)
ii. The hitting time probability distribution function can be approximated by an exponential function: $P(T > t) = \exp\left(-\frac{t}{cN \log N}\right)$.
iii. $EM = \frac{1}{2} ET$.

Proof: See [22] ■

B. An "Oracle-based" Optimal Algorithm – A Lower Bound on Delay

In this section we analyze the performance of an optimal, "oracle-based" algorithm. Although a random mobility process governs the movement of all nodes, the oracle-based algorithm is aware of all future movement of nodes. A "die" is thrown at the beginning of each scenario, whose outcome decides the full trajectory of all nodes. The algorithm then takes as input all these trajectories, and computes the *optimal set of forwarding decisions* (i.e. time and next hop), which delivers a message to its destination in the minimum amount of time. Therefore, it will serve as a lower bound on the performance of all online routing strategies, both single-copy and multiple-copy ones. (Note that this algorithm will find the same path as epidemic routing with infinite capacity and buffer space.)

In order to analyze the expected delivery time of the optimal algorithm we need to average over all possible scenarios. Since all node movements, at every time instant, are independent of each other, and of past movement, we can use the following coloring problem analog, in order to analyze the performance of the optimal algorithm:

- A number M of nodes are assumed to perform independent random walks on a $\sqrt{N} \times \sqrt{N}$ torus.
- A dice is thrown whose outcome designates a source node and a destination node for a message.
- The source node is colored *red* and all other nodes (including the destination) are colored *blue*.
- Whenever a red node *encounters* a blue node, the latter is colored red, too.

It is evident that the expected time until the destination node is colored red is equal to the expected message delivery time of the optimal algorithm.

C. Performance of Optimal Algorithm for Transmission Range K = 0

Here we assume that two nodes can communicate with (i.e. *encounter*) each other only if they lie on the same position. The following theorem calculates the expected time until the destination node is colored red. It uses the meeting time of direct transmission in order to calculate the expected time until a new node is colored red, and then estimates the total number of coloring steps necessary.

Theorem 5.1: Let ED_{opt} denote the expected message delivery delay of the optimal algorithm. When transmission range K is equal to zero

$$ED_{opt} = \frac{cN \log N}{2(M-1)} \cdot H_{M-1}, \quad (3)$$

where H_n is the Harmonic Number of order n, i.e, $H_n = \sum_{i=1}^{n} \frac{1}{i} = \Theta(\log n)$.

Proof: We will use the coloring problem analog. All M nodes start from the stationary (i.e. uniform) distribution and perform independent random walks. Thus, they remain in the stationary distribution at all times. Let us assume that at some time instant we have m red nodes and $M-m$ blue nodes, and let us pick a red node i and a blue node j. According to Lemma 5.1, the meeting time X_{ij} of nodes i and j is exponentially distributed with average $cN \log N/2$. Let X_i denote the meeting time of i with *any* of the $M-m$ blue nodes, that is, $X_i = \min_j(X_{ij})$. Finally, let $X^{(m)}$ denote the time until *any* of the red nodes meets *any* of the blue ones, when there are m total red nodes. Then, $X^{(m)} = \min_i(X_i) = \min_{ij}(X_{ij})$. However, all X_{ij} are IID exponential random variables with average $cN \log N/2$. Thus, $X^{(m)}$ is also an exponential random variable with average $\frac{cN \log N}{2m(M-m)}$.Finally, since we have started with 1 red node, the time until all nodes are colored red is given by $\sum_{m=1}^{M-1} X^{(m)}$, whose expected value can be calculated by

$$\frac{cN \log N}{2} \sum_{m=1}^{M-1} \frac{1}{m(M-m)}.$$

This is the expected time until *all* nodes are colored red. However, the destination may be colored red in any of the $M-1$ total coloring steps with equal probability. Consequently, the expected coloring time for the destination is the following:

$$\frac{cN \log N}{2(M-1)} \sum_{m=1}^{M-1} \sum_{n=1}^{m} \frac{1}{n(M-n)} = \frac{cN \log N}{2} \frac{H_{M-1}}{M-1}.$$

■

Corollary 5.1: When transmission range is equal to zero, the asymptotic improvement in the expected delivery delay of the optimal algorithm over any online single-copy routing algorithm is equal to

$$\Theta\left(\frac{M}{\log M}\right). \quad (4)$$

Proof: When transmission range is zero, a node can only forward a message to another node at the same position (state) with it. This however means that the two nodes are statistically equivalent (i.e. the two independent walks are *coupled*) at that time instant. Consequently, any online single-copy forwarding strategy has the same expected performance as *direct transmission*, which performs a factor $\Theta\left(\frac{M}{\log M}\right)$ worse than the optimal, as can be seen by Lemma 5.1 and Theorem 5.1. ■

D. Performance of Optimal Algorithm For Transmission Range $K > 0$

We will now assume that every node has transmission range $K \geq 1$. This means that any node that comes within (Manhattan) distance $d \leq K$ from a node i, can communicate with i (i.e. *encounters* i).

We first state here a useful result, regarding hitting times on subsets. Its proof is based on the electric network analogy for random walks on graphs [22], [25]. We omit the proof here, due to lack of space, but can be found in [23].

Lemma 5.2: Let j be a position in the torus, and let $A(K)$ denote the subset of all positions a, such that $|x_a - x_j| + |y_a - y_j| = K$. Let further T_A denote the time until a random walk starting from j first hits $A(K)$. The probability $P(T_A < T_j^+)$ that a walk starting from j hits $A(K)$ before it returns to j is then given by

$$P(T_A < T_j^+) = \frac{2^K - 1}{2^{K+1} - K - 2}. \tag{5}$$

Lemma 5.3 extends the expected hitting and meeting time from Lemma 5.1 to the case where transmission range is equal to $K \geq 1$.

Lemma 5.3: Let us pick uniformly a position j on an $\sqrt{N} \times \sqrt{N}$ torus. Let further $T(K)$ denote the *hitting time* until a random walk, starting from the stationary distribution, comes within range $K \geq 1$ of some position j, and let $M(K)$ denote the meeting time of two such walks. Then, for large N,

i. $$ET(K) = N\left(c\log N - \frac{2^{K+1} - K - 2}{2^K - 1}\right), \tag{6}$$

ii. $$EM(K) = \frac{1}{2}ET(K). \tag{7}$$

Proof: i) Let us assume that the random walk starts outside of j's range. The expected time $T(K)$ until the walk first comes within range (i.e. *hits*) of j is equal to the first hitting time $E_\pi T_A$ at $A(K)$. Now let $\pi(X)$ denote the stationary probability of subset X, and π_A denote the stationary distribution on set $A(K)$, that is, $\pi_A(a) = \pi(a)/\pi(A), \forall a \in A(K)$. The expected hitting time of the walk on j can be calculated as a function of $ET(K)$ as follows:

$$E_\pi T_j = ET(K) + E_{\pi_A} T_y.$$

Let us further express the first return time of a walk starting at j, $E_j T_j^+$, as a weighted average on the cases of the walk reaching or not reaching $A(K)$ before it returns back to j.

$$E_j T_j^+ = P_A \cdot (E_{\pi_A} T_y + g_1(K)) + (1 - P_A) g_2(K),$$

where $g_1(K) = E_j T_{A(K)} \leq |A(K)|^2 = O(K^2)$ and $g_2(K) = E_j[T_j^+ | P(T_A > T_j^+)] = O(K)$, according to [22]. Furthermore, using Kac's formula [22] we get that $E_j T_j^+ = 1/\pi(j) = N$, for the $\sqrt{N} \times \sqrt{N}$ torus. When $N \gg K$, the above equation can be rewritten as $E_j T_j^+ = P_A \cdot E_{\pi_A} T_y$. Replacing P_A from Eq.(5) we get that

$$E_{\pi_A} T_y = \left(\frac{2^{K+1} - K - 2}{2^K - 1}\right) N.$$

Finally, replacing $E_{\pi_A} T_y$ and $E_\pi T_j$ from Lemma 5.1 in the equation for $E_\pi T_j$ we get the desired relation for $ET(K)$. ii) The proof is a straightforward extension of the proof for Lemma 5.1. ■

Finally, Theorem 5.2 calculates the expected performance of the optimal algorithm, when the transmission range is equal to $K \geq 1$.

Theorem 5.2: When every node has a transmission range of $K \geq 1$, the expected message delivery time of the optimal algorithm is given by

$$ED_{opt(K)} = \frac{N H_{M-1}}{2(M-1)}\left(c\log N - \frac{2^{K+1} - K - 2}{2^K - 1}\right). \tag{8}$$

Proof: The proof follows directly from Lemma 5.3, using the methodology of Theorem 5.1. ■

It is easy to see by Eq.(6),(7), and(8) that the asymptotic performance improvement of the oracle-based (optimal) scheme over the direct transmission one when $K \geq 1$ is the same as when $K = 0$. However, unlike the case when transmission range K is zero, when $K \geq 1$ the direct transmission scheme is not statistically equivalent to all other single-copy routing algorithms. One can gain from transmitting the single copy of a message to an intermediate node, both locally, as explained in Section IV, and globally, as we shall see in the following analysis of the randomized algorithm, and in the simulation section.

E. Performance of the Randomized Routing Algorithm

In the case of the randomized routing algorithm the single message copy performs a random walk on the dynamically changing connectivity graph.

Definition 5.1 (Message's Random Walk): Let a message, currently at position j in the torus, be routed according to the randomized routing algorithm p. Then the message can be modelled as performing the following random walk: with probability $1 - p \cdot p_r$ it performs a pure random walk (i.e. moves to neighboring positions); with probability $p \cdot p_r$ it jumps to any of the states in subset $N(K) = \{a : d_{a,j} \leq K\}$ with equal probability, where p_r is the probability that there is at least one more node within the range of the current custodian.

The following Lemma calculates the probability of transmission p_{tx} and the average transmission distance.

Lemma 5.4: Let a message perform a random walk according to Definition 5.1, in a network consisting of M nodes performing independent random walks on an $\sqrt{N} \times \sqrt{N}$ torus. Then, the transmission probability p_{tx} of the message at any time is given by

$$p_r = 1 - \left(1 - \frac{2K^2 + 2K + 1}{N}\right)^{M-2}. \tag{9}$$

Additionally, the average transmission distance is given by

$$f(K) = \frac{K(8/3 + 2K + 4/3K^2)}{2K^2 + 2K + 1}. \tag{10}$$

Proof: Both equations can then be derived using elementary probability theory and combinatorics [23]. ■

We are now ready to analyze the performance of the randomized algorithm. Theorem 5.3 provides a tight (upper) bound on its delivery delay. We calculate the probability of transmission, at every step, and the average length of a *transmission jump*, and modify the proof of Lemma 5.1 to derive the bound.

Theorem 5.3: When every node has a transmission range of $K \geq 1$, the expected message delivery time of the randomized algorithm is given by

$$ED_{rnd} \leq \frac{N\left(c\log N - \frac{2^{K+1}-K-2}{2^K-1}\right)}{2 - p\cdot p_r + p\cdot p_r f(K)}. \qquad (11)$$

Proof: Two independent random walks S and D on a torus start from states x and y, respectively. Walk D models the destination's movement, which is a pure random walk on the torus, while walk S models the message movement and moves according to Definition 5.1. When no transmission occurs, S (the message) will move only to a neighboring state, while it makes a jump of average length $f(K)$ when a transmission occurs. Consequently, the average length of a jump of S is given by

$$\overline{d}(K) = 1 + (1 - p\cdot p_r) + p\cdot p_r f(K).$$

Consider now the function $f(x, y) = E_xT_y(K) - E_\pi T_y(K)$. The expected hitting times in $f(x, y)$ correspond to hitting times for transmission range $K \geq 1$. $E_\pi T_y(K) = ET(K)$, $\forall y$, and given by Eq.(6). Consider further the random walk S', starting also at x, which whenever walk S jumps d states far, it performs d *independent* single step movements (i.e. performs d pure random walk steps), *within one time unit*. It is evident that S will cover (i.e. bring within its transmission range) a higher number of new positions than S', at every step, and therefore S is expected to meet the destination node D faster than S'.

Now let $(\hat{X}_t, \hat{Y}_t)$ be the positions of (S', D) after t moves. Consider $W_t \equiv \overline{d}(K)t + f(\hat{X}_t, \hat{Y}_t)$, and define as M_{xy} the first meeting time between S' and D (i.e. assuming that S' started from position x and D from position y). It is easy to verify that $(W_t; 0 \leq t \leq M_{xy})$ is a martingale [26] (using similar arguments as in [22]:Ch.3 – Proposition 3). According to the optional stopping theorem [26] $EW_0 = EW_{M_{xy}}$. This means that

$$E_xT_y(K) - E_\pi T_y(K) = \overline{d}(K)EM_{xy} + Ef(\hat{X}_{M_{xy}}, \hat{Y}_{M_{xy}}).$$

However, by definition $\hat{X}_{M_{xy}} = \hat{Y}_{M_{xy}}$ and therefore $E_{\hat{X}_{M_{xy}}}T_{\hat{Y}_{M_{xy}}}(K) = 0$. Finally, $E_\pi T_{\hat{Y}_{M_{xy}}}(K) = ET(K)$. Consequently, $E_xT_y = \overline{d}(K)EM_{xy}$, which for uniformly chosen x, y implies that $EM(K) = ET(K)/\overline{d}(K)$. Replacing $ET(K)$ from Eq.(6) gives us the expected meeting time of S' and D, which is an upper bound on the delivery time of the randomized algorithm (i.e. meeting time of S and D). ∎

The theoretical results derived in this section are compared with simulation results in Section VI-E. As a final note, since the analysis of the utility-based and seek and focus routing schemes is more complicated than the optimal and randomized case, we evaluate their performance using simulations only.

VI. Simulation Results

A. Simulation Environment and Routing Protocols

We have used a custom discrete event-driven simulator to evaluate and compare the performance of different routing protocols. A slotted collision avoidance MAC protocol has been implemented, in order to arbitrate between nodes contenting for the shared channel. The single-copy routing protocols we have implemented and simulated are the following:

1) Randomized routing with probability $p = 0.5$;
2) Randomized routing with probability $p = 1.0$. This essentially is a deterministic greedy version, where a message is always handed over to a node that hasn't seen the message yet;
3) Pure utility-based routing;
4) Seek and focus (Hybrid) routing protocol with probability $p = 0.5$ when in "seek" phase;
5) Seek and focus (Hybrid) protocol with probability $p = 1.0$ when in "seek" phase;
6) Direct transmission.

As we explained in Section IV, any monotonic utility function of the last encounter timer τ, suffices to achieve the expected performance of utility-based schemes. For this reason, in our simulations we have used a simpler utility function than the one in Eq.(IV-C), namely the one described in [6]. This utility function is maintained as follows: whenever node X encounters node Y, $U_X^{(new)}(Y) = U_X(Y) + (1 - U_X(Y))0.75$; else, at every time unit $U_X^{(new)}(Y) = 0.99U_X(Y)$.

B. Scenario A—Random Walk on a 50×50 Torus

In this scenario we assume that 20 nodes perform independent random walks on a 50×50 two-dimensional torus. The transmission range K of each node is equal to 5. A single message between a randomly chosen source and destination node is routed using each of the aforementioned routing protocols. Results are averaged over a large number of runs (1000).

Fig. 3 depicts the average delivery delay and total number of transmissions. As one can see from the figure, the hybrid scheme has the best performance in terms of message delivery delay, while only slightly increasing the total number of transmissions from the pure utility-based protocol. It is interesting to note that, in this scenario, the

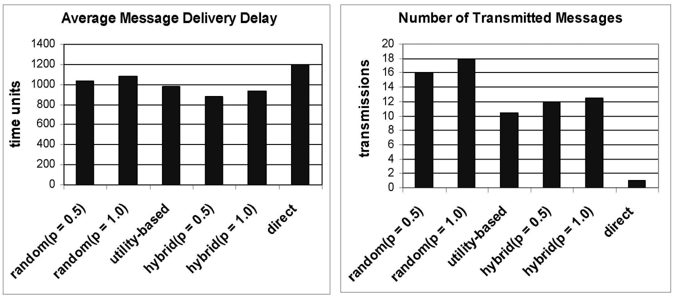

Fig. 3. Number of transmissions and average delay for Scenario A

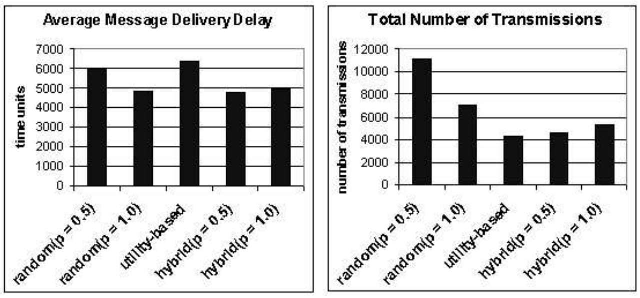

Fig. 4. Number of transmissions and average delay for Scenario B.

utility-based scheme is faster than the randomized scheme. This occurs because the network is quite small, letting location information to get quickly diffused throughout the entire network and allowing the utility-based scheme to avoid the slow start phase. Finally, it can be seen that, in a small scenario, it might be worth considering the direct transmission scheme, when transmission cost is high.

C. Scenario B—Random Walk on a 500×500 *Torus*

Here, we evaluate the performance of different routing protocols in a larger network. We assume that 50 nodes perform independent random walks on 500×500 torus. The transmission range K of each node is now equal to 60. Finally, 50 messages are routed instead of 1, in order to evaluate the performance of different protocols, when some slight contention for the channel takes place. We do not present here simulation results for the direct transmission scheme which performs very poorly.

As can be seen in Fig. 4, in this larger scenario the utility-based scheme has the largest average delivery delay, as expected. The hybrid scheme manages to overcome this, and achieve the lowest delivery delay, along with the randomized scheme (with parameter $p = 1.0$). However, as is evident from the same figure, it manages to do so with much less transmissions than the randomized schemes, and only a slight increase compared to the utility-based one. We conclude therefore that the hybrid scheme presents the best tradeoff in terms of message delivery delay and number of transmissions.

D. Scenario C—Random Waypoint on a 500×500 *Torus*

We have also evaluated all single-copy routing algorithms under a mobility model commonly used in simulations, namely the Random Waypoint model [18]. Again, there are a total of 50 nodes on a 500×500 torus. Transmission range is now equal to $K = 20$ (i.e. the network is “more disconnected”).

As can be seen in Fig. 5, the average delay in this scenario is similar to that in Scenario B, despite the shorter transmission range. This is because the nature of the mobility model is such that allows nodes to move faster between disconnected clusters. Furthermore, we can see in the same figure that the relevant performance between the five schemes is similar to that in the random walk case. However, the performance gain of the hybrid scheme is not as pronounced here. The reason for this is that the expected absolute distance covered in n steps by a node moving according to the random waypoint model, is higher than that by a node performing a random walk. Consequently, the utility function in the former case is not as informative as in the latter. We conclude therefore that, as usual, the choice of mobility model plays a significant role in the performance of different routing algorithms.

Fig. 5. Number of transmissions and average delay for Scenario C.

E. Simulation vs. Analysis

In this final section we compare our analytical results, regarding the expected delay of different algorithms, to simulation results. In the left plot in Fig. 6 we fix M to 20, K to 5, and depict the expected delay of the direct transmission (upper bound) and optimal scheme (lower bound) as a function of N. As one can see from this figure, simulation and analytical plots present a relatively close match. Some minor discrepancies are due to some approximations (stated in the proofs) and statistical errors (due to the finite number of simulation runs). Additionally, the highlighted area indicates to the performance range of *any* implementable routing scheme. Finally, in the right plot, we fix N to 2500 and M to 20, and compare the performance of direct transmission, randomized, and

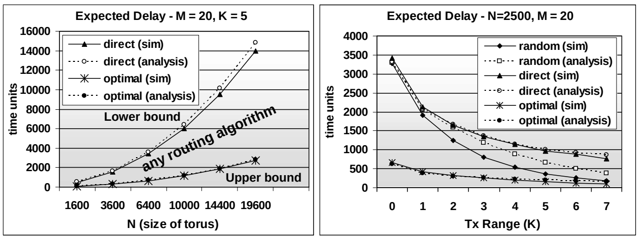

Fig. 6. Comparison of analytical and simulation results

optimal schemes, for increasing K. Both analytical and simulation plots are provided. Note that, in the case of the randomized algorithm, the analytical plot is only an upper bound (as explained in Section V).

VII. CONCLUSION

In this work, we have dealt with the problem of single-copy routing in intermittently connected mobile networks. We proposed a number of different single-copy routing strategies, and evaluated their performance through analysis and simulations. We conclude that the best algorithm is a hybrid scheme we call "seek and focus", which combines the simplicity of a simple random policy with the sophistication of utility-based policies. We also state and analyze the performance of an oracle-based optimal algorithm, and compare it to the online approaches.

In future work, we plan to perform a rigorous analysis of the performance of utility-based schemes. Additionally, we intend to look into routing schemes that attempt to justify the cost incurred by a single transmission, when making a forwarding decision. Finally, we plan to evaluate the performance of some of the routing schemes proposed under more realistic mobility models, that exhibit correlation in both space and time [20], [19].

REFERENCES

[1] Delay tolerant networking research group. [Online]. Available: http://www.dtnrg.org

[2] D. B. Johnson, D. A. Maltz, and J. Broch, *DSR: the dynamic source routing protocol for multihop wireless ad hoc networks*. Addison-Wesley Longman Pub. Co., Inc., 2001, ch. 5.

[3] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. (2002) Ad-hoc on-demand distance vector routing. IETF Mobile Ad Hoc Networking Working Group INTERNET DRAFT. [Online]. Available: http://www.ietf.org/internetdrafts /draft-ietf-manet-aodv-12.txt

[4] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, 2002.

[5] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke University, Tech. Rep. CS-200006, Apr. 2000.

[6] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.

[7] A. Beaufour, M. Leopold, and P. Bonnet, "Smart-tag based data dissemination," in *Proc. ACM International Workshop on Wireless Sensor Networks and Applications(WSNA'02)*, 2002.

[8] 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 *Proc. ASPLOS'02*, 2002, pp. 96–107.

[9] N. Glance, D. Snowdon, and J.-L. Meunier, "Pollen: using people as a communication medium," *Computer Networks*, vol. 35, no. 4, pp. 429–442, Mar. 2001.

[10] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Multiple-copy routing in intermittently connected mobile networks," USC, Tech. Rep. CENG-2004-12, 2004.

[11] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wirel. Netw.*, vol. 8, no. 2/3, pp. 153–167, 2002.

[12] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks," *Elsevier Ad Hoc Networks Journal*, vol. 1, pp. 215–233, Sept. 2003.

[13] T. Small and Z. J. Haas, "The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)," in *Proc. MobiHoc'03*, 2003.

[14] Q. Li and D. Rus, "Communication in disconnected ad hoc networks using message relay," *J. Parallel Distrib. Comput.*, vol. 63, no. 1, pp. 75–86, 2003.

[15] X. Chen and A. L. Murphy, "Enabling disconnected transitive communication in mobile ad hoc networks," in *Proc. of Workshop on Principles of Mobile Computing, colocated with PODC'01*, Aug. 2001.

[16] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Age matters: efficient route discovery in mobile ad hoc networks using encounter ages," in *Proc. ACM/IEEE MobiHoc'03*, 2003, pp. 257–266.

[17] Sensor networking with delay tolerance (sendt). [Online]. Available: http://down.dsg.cs.tcd.ie/sendt/

[18] C. Bettstetter, H. Hartenstein, and X. Perez-Costa, "Stochastic properties of the random waypoint mobility model: epoch length, direction distribution, and cell change rate," in *Proc. ACM Int. Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*. ACM Press, 2002.

[19] F. Bai, N. Sadagopan, and A. Helmy, "The important framework for analyzing the impact of mobility on performance of routing for ad hoc networks," 2003.

[20] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri, "Towards realistic mobility models for mobile ad hoc networks," in *Proc. Mobicom'03*, Sept. 2003.

[21] P. Gupta and P. R. Kumar, "Critical power for asymptotic connectivity in wireless networks," in *in Stoch. Analysis, Control, Optimization and Applications*. Birkhauser, Boston, 1998.

[22] D. Aldous and J. Fill. Reversible markov chains and random walks on graphs. monograph in preparation. [Online]. Available: http://stat-www.berkeley.edu/users/aldous/RWG/book.html

[23] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," USC, Tech. Rep. CENG-2004-11, 2004.

[24] Torus hitting times and green's functions. [Online]. Available: http://www.math.tamu.edu/ rellis/comb/torus/torus.html

[25] A. K. Chandra, P. Raghavan, W. L. Ruzzo, and R. Smolensky, "The electrical resistance of a graph captures its commute and cover times," in *Proc. of ACM Symp. on Theory of Comp.*, 1989.

[26] R. Durrett, *Probability: Theory and Examples*, 2nd ed. Duxbury Press, 1995.