Probabilistic Routing in Intermittently Connected Networks

Anders Lindgren[†], Avri Doria[*], and Olov Schelén[†]

[†]Division of Computer Science and Networking
Department of Computer Science and Electrical Engineering
Luleå University of Technology, SE - 971 97 Luleå, Sweden
{dugdale, olov}@sm.luth.se

[*]Electronics and Telecommunications Research Institute (ETRI)
161 Gajeong-don, Yuseong-gu, Daejeon
305-350, Korea
avri@acm.org

Abstract. In this paper, we address the problem of routing in intermittently connected networks. In such networks there is no guarantee that a fully connected path between source and destination exists at any time, rendering traditional routing protocols unable to deliver messages between hosts. There does, however, exist a number of scenarios where connectivity is intermittent, but where the possibility of communication still is desirable. Thus, there is a need for a way to route through networks with these properties. We propose PROPHET, a probabilistic routing protocol for intermittently connected networks and compare it to the earlier presented Epidemic Routing protocol through simulations. We show that PROPHET is able to deliver more messages than Epidemic Routing with a lower communication overhead.

1 Introduction

The dawn of new and cheap wireless networking solutions has created opportunities for networking in new situations, and for exciting new applications that use the network. With techniques such as IEEE 802.11, and other radio solutions (e.g. low power radios designed for use in sensor networks), it has become viable to equip almost any device with wireless networking capabilities. Due to the ubiquity of such devices, situations where communication is desirable can occur at any time and any place, even where no networking infrastructure is available. One research area that has received much attention recently and that remedies many of the situations where no infrastructure is available is that of ad hoc networking [14]. In an ad hoc network, all nodes participate in the routing and forwarding of packets, so if two nodes cannot communicate directly, intermediate nodes aid in forwarding packets between them.

One of the most basic requirements for "traditional" networking, which also holds for ad hoc networking, is that there must exist a fully connected path between communication endpoints for communication to be possible. There are, however, a number of scenarios where this is not the case (thus rendering ad hoc network routing protocols useless), but where it still is desirable to allow communication between nodes (see Sect. 2 for a survey of such scenarios).

One way to enable communication in such scenarios, is by allowing messages to be buffered for a long time at intermediate nodes, and to exploit the mobility of those nodes to bring messages closer to their destination by transferring messages to other nodes as they meet. Figure 1 shows how the mobility of nodes in such scenarios can be used to eventually deliver a message to its destination. In this figure, node A has a message (indicated by the node being shaded) to be delivered to node D, but there does not exist a path between nodes A and D. As shown in subfigures a)-d), the mobility of the nodes allow the message to first be transferred to node B, then to node C, and finally node C moves within range of node D and can deliver the message to its final destination.

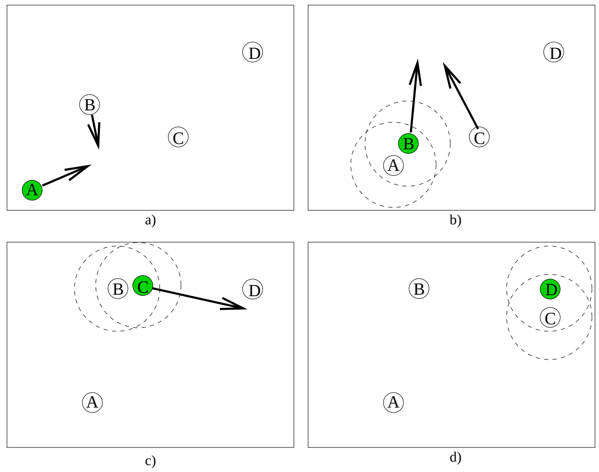

Fig. 1. Transitive communication. A message (shown in the figure by the node carrying the message being shaded) is passed from node A to node D via nodes B and C through the mobility of nodes.

Previous work [3, 6, 15, 17] have tried to either solve this by epidemically spreading the information through the network, or by applying some knowledge of the mobility of nodes.

We have previously proposed the idea of probabilistic routing [11], using an assumption of non-random mobility of nodes to improve the the delivery rate of messages while keeping buffer usage and communication overhead at a low level. This paper presents a framework for such probabilistic routing in intermittently connected networks. A probabilistic metric called delivery predictability is defined. Further, it defines a probabilistic routing protocol using the notion of delivery predictability, and evaluates it through simulations versus the previously proposed Epidemic Routing [17] protocol.

The rest of the paper is organized as follows. Section 2 gives the background to this work, while Sect. 3 describes some related work and in Sect. 4, our proposed scheme is presented. In Sect. 5 the simulation setup is given, and the results of the simulations can be found in Sect. 6. Finally, Sect. 7 discusses some issues and looks into future work and Sect. 8 concludes.

2 Background

The kind of communication networks addressed in this paper are only viable for applications that can tolerate long delays and are able to deal with extended periods of being disconnected. In this section, we survey previous work dealing with deployment of such communication networks in a variety of practical systems.

The aboriginal Saami population of reindeer herders in the north of Sweden follow the movement of the reindeer and when in their summer camps, no fixed infrastructure is available. Still, it would be desirable to be able to communicate with the rest of the world through, for example, mobile relays attached to snowmobiles and ATVs [4]. Similar problems exist between rural villages in India and other regions on the other side of the digital divide. The DakNet project [13] has deployed store-and-forward networks connecting a number of villages through relays on buses and motorcycles in India and Cambodia.

In military war-time scenarios and disaster recovery situations, soldiers or rescue personnel often are in hostile environments where no infrastructure can be assumed to be present. Furthermore, the units may be sparsely distributed so connectivity between them is intermittent and infrequent.

In sensor networks, a large number of sensors are usually deployed in the area in which measurements should be done. If sensors are mobile and transitive communication techniques can be used between them, the number of sensors required can be reduced, and new areas where regular sensor networks have been too expensive or difficult to deploy, can be monitored. Experiments have been done with attaching sensors to seals [2], vastly increasing the number of oceanic temperature readings compared to using a number of fixed sensors, and in a similar project sensors are attached to whales [16]. To allow scientists to analyze the collected data, it must somehow be transferred to a data sink, even though connectivity among the seals and whales is very sparse and intermittent, so the mobility of the animals (and their occasional encounters with each other and networked buoys at feeding grounds) must be relied upon for successful data delivery. In a similar project, ZebraNet, an attempt is made to gain a better understanding of the life and movements of the wildlife in a certain part of Africa by equipping zebras with tracking collars communicating in fashions similar to the ones described above [9]. Yet another example concerns weather monitoring of large areas such as a national park, where a number of electronic display boards showing weather reports from other parts of the park have been installed. By equipping hikers with small networked devices, their mobility through the park can be used to spread the weather information throughout the entire park [1].

As shown above, several situations exist where communication in intermittently connected networks is of high importance. Further, in any large scale ad hoc network (even apart from the scenarios above), intermittent connectivity is likely to be the norm, and thus research in this area is likely to have payoff in practical systems.

3 Related work

3.1 Epidemic Routing

Vahdat and Becker present a routing protocol for intermittently connected networks called Epidemic Routing [17]. This protocol relies on the theory of epidemic algorithms by doing pair-wise information of messages between nodes as they get contact with each other to eventually deliver messages to their destination. Hosts buffer messages even if no path to the destination is currently available. An index of these messages, called a *summary vector*, is kept by the nodes, and when two nodes meet they exchange summary vectors. After this exchange, each node can determine if the other node has some message that was previously unseen to this node. In that case, the node requests the messages from the other node. The message exchange is illustrated in Fig. 2. This means that as long as buffer space is available, messages will spread like an epidemic of some disease through the network as nodes meet and "infect" each other.

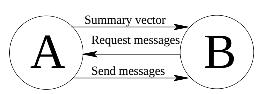

Fig. 2. Message exchange between two nodes using Epidemic Routing.

Each message must contain a globally unique message ID to determine if it has been previously seen. Besides the obvious fields of source and destination addresses, messages also contain a hop count field. This field is similar to the TTL field in IP packets and determines the maximum number of hops a message can be sent, and can be used to limit the resource utilization of the protocol. Messages with a hop count of one will only be delivered to their final destination.

The resource usage of this scheme is regulated by the hop count set in the messages, and the available buffer space at the nodes. If these are sufficiently large, the message will eventually propagate throughout the entire network if the possibility exists. Vahdat and Becker show that by choosing an appropriate maximum hop count, rather high delivery rates can be achieved, while the required amount of resources can be kept at an acceptable level in the scenarios used in their evaluation [17]. In their evaluation, they also compare Epidemic Routing to a slightly modified version of DSR [8], and show that ordinary ad hoc routing protocols perform badly in scenarios like this.

3.2 Other work

In recent work by Small and Haas [16], a new networking model called the Shared Wireless Infostation Model (SWIM) is presented. In this model, mobile nodes work in a manner that is similar to the operation of Epidemic Routing. Nodes cooperate in conveying information from the network to (possibly mobile) *Infostations* that collect this data. Nodes do, however, only give messages to other nodes they meet with a certain, configurable, probability (which is constant throughout a single test). The authors also identify an application for this in the collection of oceanografic data with the use of sensors attached to whales.

A communication model that is similar to Epidemic Routing is presented by Beaufour et al. [1], focusing on smart-tag based data dissemination in sensor networks. The Pollen network proposed by Glance et al. [6] is also similar to Epidemic Routing. It does, however, allow for the possibility to have a centralized entity (a "hive") with which mobile nodes can synchronize their data. The hive can also aid in making more intelligent routing decisions by predicting which mobile nodes are more likely to go where.

Chen and Murphy propose a protocol called Disconnected Transitive Communication (DTC) [3]. It utilizes a *utility* function to locate the node in the cluster of currently connected nodes that is most suitable to forward the message to based on the needs of the application. The utility function can be tuned by the application by modifying the weights of different parts of the function. In every step, a node searches the cluster of currently connected nodes for a node that is "closer" to the destination, where the closeness is given by a *utility* function that can be tuned by the application to give appropriate results.

Dubois-Ferriere et al. present an idea based on the concept of encounter ages to improve the route discovery process of regular ad hoc networks [5]. These encounter ages bear some resemblance to our delivery predictability metric and create a gradient for the route request packets.

Other related work that deals with similar problems but does not have a direct connection to our work include work by Nain et al. [12], Shen et al. [15], Li and Rus [10], and Grossglauser and Tse [7].

4 Probabilistic routing

Although the random way-point mobility model is popular to use in evaluations of mobile ad hoc protocols, real users are not likely to move around randomly, but rather move in a predictable fashion based on repeating behavioral patterns such that if a node has visited a location several times before, it is likely that it will visit that location again.

In the previously discussed mechanisms to enable communication in intermittently connected networks, such as Epidemic Routing, very general approaches have been taken to the problem at hand. The Pollen network has the possibility of using predictions of node mobility for routing, but that requires the presence of a central entity to control this. There have, however, not been any attempts to make use of assumed knowledge of different properties of the nodes in the network in a truly distributed way.

Further, we note that in an environment where buffer space and bandwidth are infinite, Epidemic Routing will give an optimal solution to the problem of routing in an intermittently connected network with regard to message delivery ratio and latency. However, in most cases neither bandwidth nor buffer space is infinite, but instead they are rather scarce resources, especially in the case of sensor networks. Therefore, it would be of great value to find an alternative to Epidemic Routing, with lower demands on buffer space and bandwidth, and with equal or better performance in cases where those resources are limited, and without loss of generality in scenarios where it is applicable.

4.1 PROPHET

To make use of the observations of the non-randomness of mobility and to improve routing performance we consider doing *probabilistic routing* and propose PROPHET, a Probabilistic ROuting Protocol using History of Encounters and Transitivity.

To accomplish this, we establish a probabilistic metric called *delivery predictability*, $P_{(a,b)} \in [0,1]$, at every node a for each known destination b. This indicates how likely it is that this node will be able to deliver a message to that destination. When two nodes meet, they exchange summary vectors, and also a delivery predictability vector containing the delivery predictability information for destinations known by the nodes. This additional information is used to update the internal delivery predictability vector as described below. After that, the information in the summary vector is used to decide which messages to request from the other node based on the forwarding strategy used (as discussed in Sect. 4.1).

Delivery predictability calculation The calculation of the delivery predictabilities has three parts. The first thing to do is to update the metric whenever a node is encountered, so that nodes that are often encountered have a high delivery predictability. This calculation is shown in Eq. 1, where $P_{init} \in (0,1]$ is an initialization constant.

$$P_{(a,b)} = P_{(a,b)_{old}} + \left(1 - P_{(a,b)_{old}}\right) \times P_{init} \tag{1}$$

If a pair of nodes does not encounter each other in a while, they are less likely to be good forwarders of messages to each other, thus the delivery predictability values must *age*, being reduced in the process. The aging equation is shown in Eq. 2, where $\gamma \in (0,1)$ is the *aging constant*, and k is the number of time units that have lapsed since the last time the metric was aged. The time unit used can differ, and should be defined based on the application and the expected delays in the targeted network.

$$P_{(a,b)} = P_{(a,b)_{old}} \times \gamma^k \tag{2}$$

The delivery predictability also has a *transitive* property, that is based on the observation that if node A frequently encounters node B, and node B frequently encounters node C, then node C probably is a good node to forward messages destined for node A to. Eq. 3 shows how this transitivity affects the delivery predictability, where $\beta \in [0,1]$ is a scaling constant that decides how large impact the transitivity should have on the delivery predictability.

$$P_{(a,c)} = P_{(a,c)_{old}} + \left(1 - P_{(a,c)_{old}}\right) \times P_{(a,b)} \times P_{(b,c)} \times \beta \tag{3}$$

Forwarding strategies In traditional routing protocols, choosing where to forward a message is usually a simple task; the message is sent to the neighbor that has the path to the destination with the lowest cost (usually the shortest path). Normally the message is also only sent to a single node since the reliability of paths is relatively high. However, in the settings we envision here, things are completely different. For starters, when a message arrives at a node, there might not be a path to the destination available so the node have to buffer the message and upon each encounters with another node, the decision must be made on whether or not to transfer a particular message. Furthermore, it may also be sensible to forward a message to multiple nodes to increase the probability that a message is really delivered to its destination.

Unfortunately, these decisions are not trivial to make. In some cases it might be sensible to select a fixed threshold and only give a message to nodes that have a delivery predictability over that threshold for the destination of the message. On the other hand, when encountering a node with a low delivery predictability, it is not certain that a node with a higher metric will be encountered within reasonable time. Thus, there can also be situations where we might want to be less strict in deciding who to give messages to. Furthermore, there is the problem of deciding how many nodes to give a certain message to. Distributing a message to a large number of nodes will of course increase the probability of delivering a message to its destination, but in return, more system resources will be wasted. On the other hand, giving a message to only a few nodes (maybe even just a single node) will use less system resources, but the probability of delivering a message is probably lower, and the incurred delay higher.

In the evaluations in this paper, we have chosen a rather simple forwarding strategy – when two nodes meet, a message is sent to the other node if the delivery predictability of the destination of the message is higher at the other node. The first node does not delete the message after sending it as long as there is sufficient buffer space available (since it might encounter a better node, or even the final destination of the message in the future). If buffers are full when a new message is received, a message must be dropped according to the queue management system used. In our evaluations, we have used FIFO queues.

4.2 PROPHET Example

To help grasp the concepts of PROPHET, an example is provided to give a understanding of the transitive property of the delivery predictability, and the basic operation of PROPHET. In Fig. 3, we revisit the scenario where node A has a message it wants to send to node D. In the bottom right corner of subfigures a)-c), the delivery predictability tables for the nodes are shown. Assume that nodes C and D encounter each other frequently (Fig. 3a), making the delivery predictability values they have for each other high. Now assume that node C also frequently encounters node B (Fig. 3b). B and C will get high delivery predictability values for each other, and the transitive property will also increase the value B has for D to a medium level. Finally, node B meets node A (Fig. 3c) that has a message for node D. Figure 3d) shows the message exchange between node A and node B. Summary vectors and delivery predictability information is exchanged, delivery predictabilities are updated, and node A then realized that $P_{(b,d)} > P_{(a,d)}$, and thus forwards the message for D to node B.

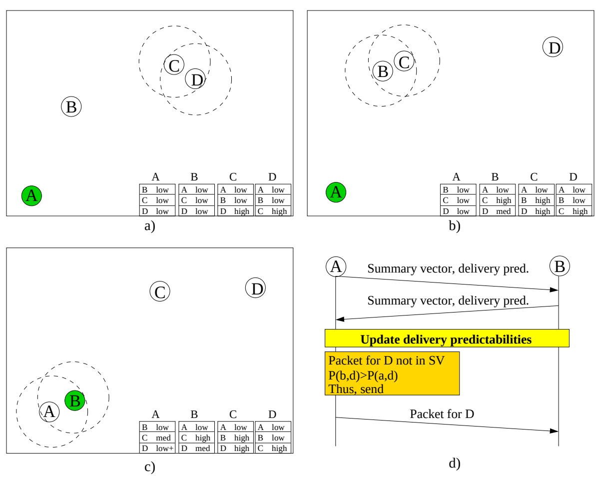

Fig. 3. PROPHET example. a)-c) show the transitive property of the delivery predictability, and d) show the operation of the protocol when two nodes meet.

5 Simulations

To evaluate the protocol, we developed a simple simulator. The reason for implementing a new simulator instead of using one of the large number of widely available simulators was mainly due to a desire to focus on the operation of the routing protocols instead of simulating the details of the underlying layers. In some cases, accurate modeling of physical layer phenomenon such as interference is important as it can affect the results of the evaluation. In this case however, such details should not influence our results.

5.1 Mobility Model

Since we base our protocol on making predictions depending on the movements of nodes, it is vital that the mobility models we use are realistic. This is unfortunately not the case for

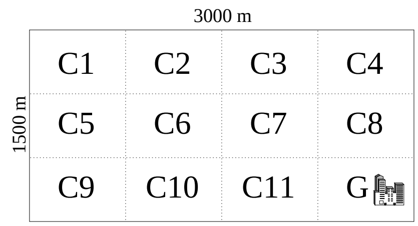

Fig. 4. Community model

the commonly used random waypoint mobility model [8]. Thus, it is desirable to model the mobility in another way to better reflect reality.

We have designed a mobility model that we call the "community model". In this model, we have a $3000m \times 1500m$ area as shown in Fig. 4. This area is divided into 12 subareas; 11 communities (C1-C11), and one "gathering place" (G). Each node has one home community that it is more likely to visit than other places, and for each community there are a number of nodes that have that as home community. Furthermore, in each community, and at the gathering place, there is a fixed (non-mobile) node as well that could be acting as a gateway for that community. The mobility in this scenario is such that nodes select a destination and a speed, move there, pause there for a while, and select a new destination and speed. The destinations are selected such that if a node is at home, there is a high probability that it will go to the gathering place (but it is also possible for it to go to other places), and if it is away from home, it is very likely that it will return home. Table 1 shows the probabilities of different destinations being chosen depending on the current location of a node. Real-life scenarios where this kind of mobility can occur include human mobility where the communities are, for example, villages, and the gathering place a large town, but also sensor network applications where sensors are attached to animals – in such cases the gathering place may be a feeding ground, and the communities can be herd habitats.

Table 1. Destination selection probabilities

From \ To	**Home**	**Gathering place**	**Elsewhere**
Home	-	0.8	0.2
Elsewhere	0.9	-	0.1

5.2 Simulation setup

We have used two different scenarios in our evaluation of the protocols. To compare PROPHET to Epidemic Routing in a scenario that Epidemic Routing is known to be able to handle, we used a scenario that is very similar to the one used by Vahdat and Becker [17] as a reference. This scenario consists of a $1500m \times 300\ m$ area where 50 nodes are randomly placed. These nodes move according to the random waypoint mobility model [8] with speeds of $0-20\ m/s$. From a subset of 45 nodes, one message is sent every second for 1980 seconds of the simulation (each of the 45 nodes sending one message to the other 44 nodes), and the simulation is then run for another 2020 seconds to allow messages to be delivered.

The second scenario we have used is based on the community mobility model defined in Sect. 5.1. For each community, there are five nodes that have that as their home community. After each pause, nodes select speeds between 10 and 30 m/s. Every tenth second, two randomly chosen community gateways generate a message for a gateway at another community or at the gathering place. Five seconds after each such message generation, two randomly chosen mobile nodes generate a message to a randomly chosen destination. After 3000 seconds the message generation ceases and the simulation is run for another 8000 seconds to allow messages to be delivered.

In both scenarios, a *warm up* period of 500 seconds is used in the beginning of the simulations before message generation commence, to allow the delivery predictabilities of PROPHET to initialize.

Table 2. Parameter settings

Parameter	Value
P_{init}	0.75
β	0.25
γ	0.98

We have focused on comparing the performance of the protocols with regard to the following metrics. First of all, we are interested in the *message delivery ability*, i.e. how many of the messages initiated the protocol is able to deliver to the destination. Even though applications using this kind of communication should be relatively delay-tolerant, it is still of interest to consider the *message delivery delay* to find out how much time it takes a message to be delivered. Finally, we also study the number of *message exchanges* that occur between nodes. This indicates how the system resource utilization is affected by the different settings, which is crucial so that valuable resources such as bandwidth and energy are not wasted.

We ran simulations for each scenario, varying the queue size at the nodes (the number of messages a node can buffer), the communication range of nodes, and the hop count value set in the messages. For each setup, we made 5 simulation runs with different random seed. Table 2 shows the values for parameters kept fixed in our simulations (initial simulations indicated that those values were reasonable choices for the parameters).

6 Results

The results presented here are averages from 5 simulation runs, and the error bars in the graphs represent the 95% confidence intervals. For each metric and scenario, there are two graphs with two different values of the hop count setting. Each of these graphs contain curves for both Epidemic Routing and PROPHET for the two different communication ranges. We plot the different metrics versus the queue size in the nodes.

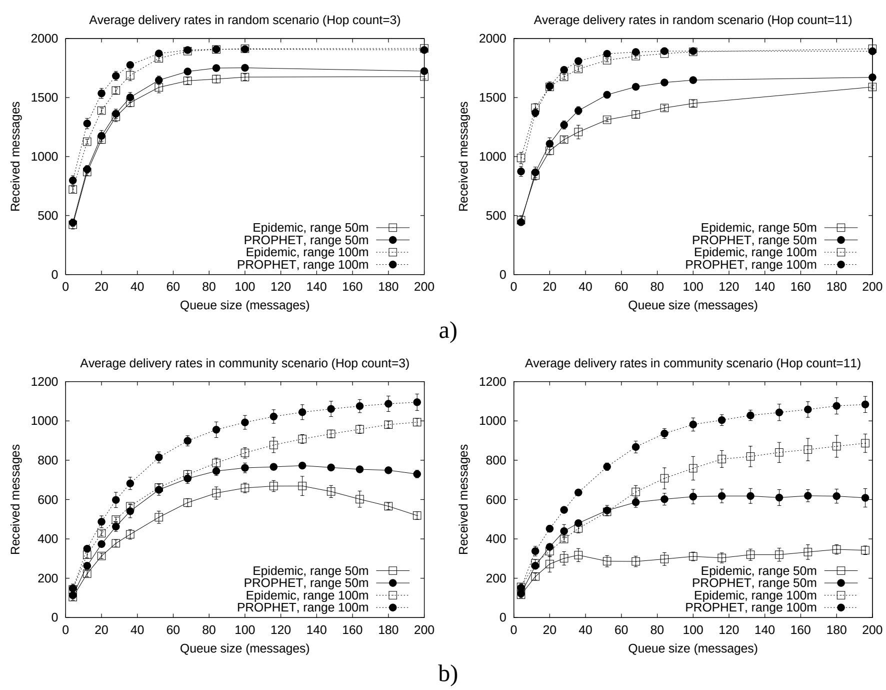

Fig. 5. Received messages. a) random mobility scenario b) community scenario

First, we investigate the delivery rates of the protocols in the different scenarios, shown in Fig. 5. It is easy to see that the queue size impacts performance; as the queue size increases, so does the number of messages delivered to their destination for both protocols. This is intuitive, since a larger queue size means that more messages can be buffered, and the risk of throwing away a message decreases. In the random mobility scenario, the performance is similar for both protocols, even though PROPHET seem to perform slightly better, especially with short communication range, and a high hop count. It is interesting to see that even though mobility is completely random, PROPHET still operates in a good way, and even outperforms Epidemic Routing slightly. In the community model scenario, there is a significant difference between the performance for the two protocols, and it can be seen that PROPHET is at times able to deliver up to twice as many messages as Epidemic Routing. Interesting to note is that the delivery rate (especially for the short communication range) is adversely affected by an increase in the hop count. This is probably due to the fact that with a higher hop count, messages can spread through a larger part of the network, occupying resources that otherwise would be used by other messages, while with a lower hop count, the mobility of the nodes has greater importance.

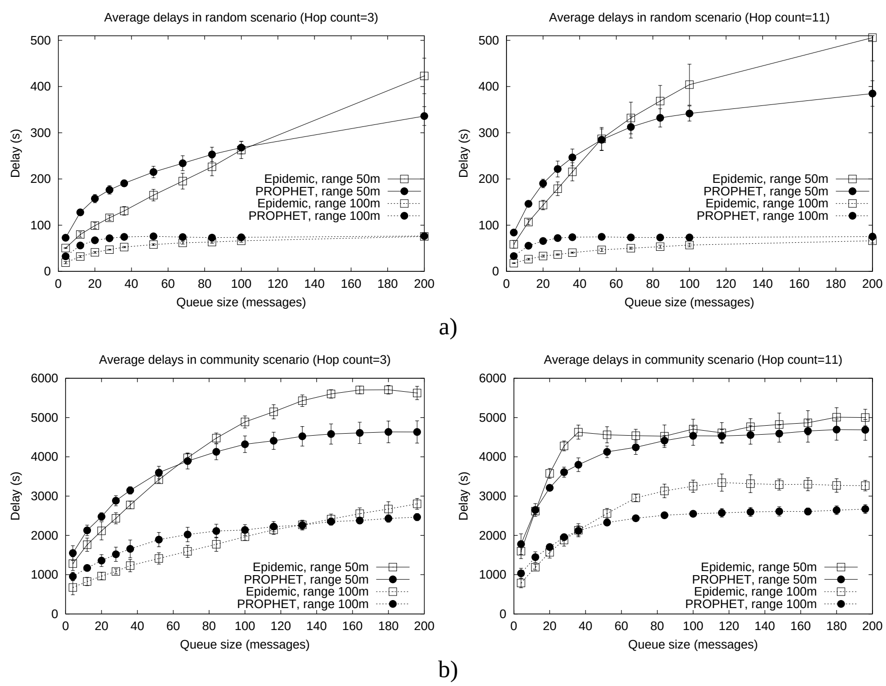

Fig. 6. Delay of messages. a) random mobility scenario b) community scenario

Looking at the delivery delay graphs (Fig. 6), it seems like increasing the queue size, also increases the delay for messages. However, the phenomenon seen is probably not mainly that the delay increases for messages that would be delivered even at a smaller queue size (even though large buffers might lead to problems in being able to exchange all messages between two nodes, leading to a higher delay), but the main reason the average delay is higher is coupled to the fact that more messages are delivered. These extra delivered messages are messages that were dropped at smaller queue sizes, but now are able to reside in the queues long enough to be delivered to their destinations. This incurs a longer delay for these messages, increasing the average delay. This theory is corroborated by Fig. 8, which shows a CDF of the message delivery delays for a selected scenario for some different queue sizes. Both PROPHET and Epidemic Routing have similar delays in both scenarios, but as queue sizes grow large, PROPHET seems to have shorter delays.

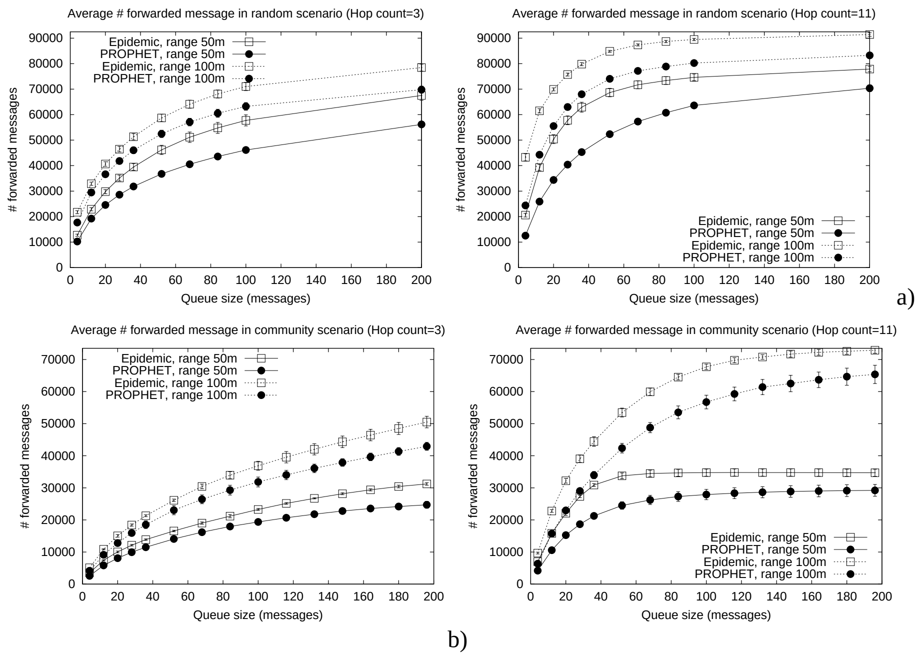

Fig. 7. Communication overhead. a) random mobility scenario b) community scenario

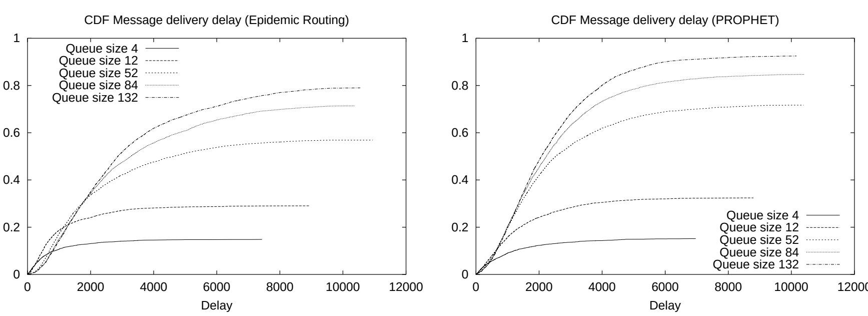

Fig. 8. CDF of delay of messages (community scenario, hop count=11, range=100m)

Finally, looking at the graphs in Fig. 7, it can be clearly seen that PROPHET has a lower communication overhead and sends fewer messages than Epidemic routing does. This is due to the fact that when using PROPHET messages are only sent to “better” nodes, while Epidemic routing sends all possible messages to nodes encountered.

Another thing that can be seen from the graphs is that increasing the communication range generally increases the performance in terms of delivery rate and delay, but also increases the communication overhead. This is not very surprising, since a larger communication range allows nodes to communicate directly with a larger number of other nodes and increases the probability of two nodes meeting each other.

Interesting to note is that even in the random mobility scenario, the performance of PROPHET with regard to delivery rate and delay is comparable to that of Epidemic Routing, but with lower communication overhead, thus being more efficient. At first glance, this could be considered somewhat remarkable, since because of the total randomness in the mobility of nodes in this scenario, predicting good forwarding nodes should be difficult. However, since the delivery predictability favors nodes frequently met, and the fact that even if mobility is random, nodes that previously were close, probably have not moved that far away from each other, it actually is reasonable that this occur. It is because of similar reasons that the approach taken by Dubois-Ferriere et al. to improve route discovery works [5].

7 Discussion and future work

The new networking possibilities introduced by the ubiquitous deployment of lightweight wireless devices have the potential to give rise to a plethora of new applications and networked solutions where such were previously impossible. Since applicable infrastructure can not be expected to be omnipresent, it is vital that solutions that can handle periods of intermittent connectivity are developed. Thus, we feel that the routing aspect of the problem studied in this paper is important to work on. This paper shows that it is possible to, in a relatively simple way, do better than to just epidemically flood messages through a network – an aspect that is likely to be valuable from a scalability point of view as networks grow larger. We believe that this is a field of research where much work remains to be done in the future. Some of the issues to work on in the future are outlined below.

In a real scenario, it is very likely that the network will be a mix of truly intermittently connected nodes that only encounter other nodes occasionally, and clusters of currently connected nodes. It is also possible that in some areas, connectivity will be present, e.g. through a satellite or GSM link, but that the bandwidth of this link is highly limited, or its use might be so expensive that it is not feasible to do bulk data transfer over it. Still, it might be of interest to use these carriers to send requests for bulk data transfers that are then delivered through the intermittently connected network (approximately halving the delivery time of the data). The protocol should be extended to handle situations such as these, while still performing equally well in the case of a truly opportunistic network. Since such extensions are likely to increase the complexity of the protocol, it is important that studies such as this one are conducted to gain an understanding of the basic protocol before moving on to more complex systems.

In our evaluation we have used a FIFO queue at the nodes, so whenever a new message arrives to a full queue, the message that has been in the queue for the longest time is dropped. It might be better to use some other strategy here; for example, dropping the message that has already been forwarded to the largest number of other nodes.

To reduce the required buffer space, and to further improve performance, it would be interesting to evaluate the impact of allowing nodes to request an ACK to their message. This would allow messages that already have been delivered to be purged from the network, leaving more resources for the other messages, most likely increasing the probability of those messages being delivered.

The simple forwarding strategy used by PROPHET in our evaluation worked fairly well, and outperformed Epidemic Routing. Nevertheless, it is still interesting to investigate other forwarding strategies to see if performance can be further enhanced. It can, for example, be beneficial to investigate if it always is a good idea to give messages to nodes with a higher delivery predictability than yourself and only such nodes, or if some other strategy should be used sometimes. Similarly, the number of nodes that a message is forwarded to could be limited (reducing the resource usage), and then it is vital to find out what the optimal number of forwards is to avoid performance degradations.

8 Conclusions

In this paper we have looked at intermittently connected networks, an area where a lot of new applications are viable, vouching for an exciting future if the underlying mechanisms are present. Therefore, we have proposed the use of probabilistic routing using observations of non-randomness in node mobility in such networks. To accomplish this, we have defined a *delivery predictability* metric, reflecting the history of node encounters and transitive and time dependent properties of that relation. We have proposed PROPHET, a probabilistic protocol for routing in intermittently connected networks, that is more sophisticated than previous protocols. PROPHET uses the new metric to enhance performance over previously existing protocols. Simulations performed have shown that in a community based scenario, PROPHET clearly gives better performance than Epidemic Routing. Further, it is also shown that even in a completely random scenario (for which PROPHET was not designed), the performance of PROPHET is still comparable with (and often exceeds) the performance of Epidemic Routing. Thus, it is fair to say that PROPHET succeeds in its goal of providing communication opportunities to entities in a intermittently connected network with lower communication overhead, less buffer space requirements, and better performance than existing protocols.

References

1. Allan Beaufour, Martin Leopold, and Philippe Bonnet. Smart-tag based data dissemination. In *First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA02)*, June 2002.
2. George W. Boehlert, Daniel P. Costa, Daniel E. Crocker, Phaedra Green, Todd O'Brien, Syd Levitus, and Burney J. Le Boeuf. Autonomous pinniped environmental samplers; using instrumented animals as oceanographic data collectors. *Journal of Atmospheric and Oceanic Technology*, 18(11):1882–1893, 2001. 18 (11).
3. Xiangchuan Chen and Amy L. Murphy. Enabling disconnected transitive communication in mobile ad hoc networks. In *Proc. of Workshop on Principles of Mobile Computing, colocated with PODC'01, Newport, RI (USA)*, pages 21–27, aug 2001.
4. Avri Doria, Maria Udén, and Durga Prasad Pandey. Providing connectivity to the saami nomadic community. In *Proceedings of the 2nd International Conference on Open Collaborative Design for Sustainable Innovation (dyd 02), Bangalore, India*, Dec 2002.
5. Henri Dubois-Ferriere, Matthias Grossglauser, and Martin Vetterli. Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages. In *Proceedings of The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003)*, June 2003.
6. Natalie Glance, Dave Snowdon, and Jean-Luc Meunier. Pollen: using people as a communication medium. *Computer Networks*, 35(4):429–442, March 2001.
7. Matthias Grossglauser and David Tse. Mobility increases the capacity of ad-hoc wireless networks. *IEEE/ACM Transactions on Networking*, 10(4), Aug 2002.
8. David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.
9. Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li-Shiuan Peh, and Daniel Rubenstein. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In *Proceedings of Tenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X), San Jose, CA*, October 2002.
10. Qun Li and Daniela Rus. Communication in disconnected ad-hoc networks using message relay. *Journal of Parallel and Distributed Computing*, 2003.
11. Anders Lindgren, Avri Doria, and Olov Schelén. Poster: Probabilistic routing in intermittently connected networks. In *Proceedings of The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003)*, June 2003.
12. Delphine Nain, Noshirwan Petigara, and Hari Balakrishnan. Integrated routing and storage for messaging applications in mobile ad hoc networks. In *Proceedings of WiOpt'03:Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, Sophia-Antipolis, France*, March 2003.
13. Alex Pentland, Richard Fletcher, and Amir A. Hasson. A road to universal broadband connectivity. In *Proceedings of the 2nd International Conference on Open Collaborative Design for Sustainable Innovation (dyd 02), Bangalore, India*, Dec 2002.
14. Elizabeth M. Royer and Chai-Keong Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications Magazine*, 6(2):46–55, April 1999.
15. Chien-Chung Shen, Girish Borkar, Sundaram Rajagopalan, and Chaiporn Jaikaeo. Interrogation-based relay routing for ad hoc satellite networks. In *Proceedings of IEEE Globecom 2002, Taipei, Taiwan*, November 17-21 2002.
16. Tara Small and Zygmunt Haas. The shared wireless infostation model - a new ad hoc networking paradigm (or where there is a whale, there is a way). In *Proceedings of The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003)*, pages 233–244, June 2003.
17. Amin Vahdat and David Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, April 2000.