Probabilistic Routing in Intermittently Connected Networks

Anders Lindgren†   Avri Doria*   Olov Schelén†

dugdale@sm.luth.se   avri@acm.org   olov@sm.luth.se

Abstract—**In this paper we consider 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 do 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 such networks. We propose PROPHET, a probabilistic routing protocol for such 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.**

I. 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, Bluetooth, 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 networking enabled devices, situations where communication is desirable can occur at any time and any place, even where no networking infrastructure is available. One area that have received much attention recently and that remedies many of the situations where no infrastructure is available is that of ad hoc networking [1]. In an ad hoc network, all nodes participate in the routing and forwarding of packets, so if two nodes can not communicate directly, intermediate nodes aid in forwarding the packet 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 the use of ad hoc networking protocols impossible), but where it still is desirable to allow communication between nodes. Such scenarios include communication between villages and summer camps of the Saami population of reindeer herders in the north of Sweden [2], living in locations where no fixed infrastructure is available. Similar problems exist between rural villages in India and other poor regions [3]. Other fields where this kind of communication scenarios may occur are satellite communication [4], military and disaster recovery operations, sensor networking and monitoring. For example, experiments have been done with attaching sensors to seals [5] and whales [6] to be able to get a large number of sensor readings from the oceans. 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 must be relied upon for successful data delivery. 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 [7].
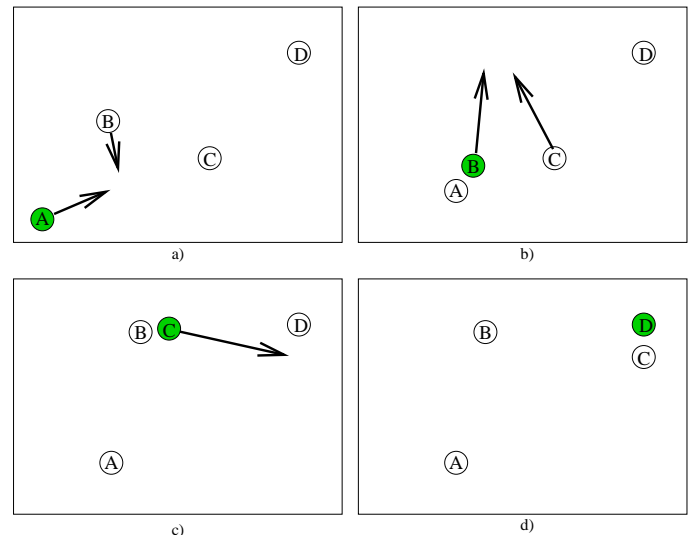

Fig. 1. Transitive communication. A message (shown in the figure by the node carrying the message being green) is passed from node A to node D via nodes B and C through the mobility of nodes.

Common to all the scenarios exemplified above is that to enable communication, messages may have to be buffered for a long time by intermediate nodes, and the mobility of those nodes must be exploited to bring messages closer to their destination by exchanging messages between 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 green) 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.

Quite some effort has been put into work regarding applications and architectures for this kind of networks, while underlying mechanisms such as routing have been neglected. For example, the Rover Toolkit [8] provides a set of programming and communication abstractions that enable applications to be aware of and handle mobility. Within the context of Delay Tolerant Networking (DTN), which this kind of network is a special case of, an extensive architecture has been defined [9], but the

†Division of Computer Science and Networking, Department of Computer Science and Electrical Engineering, Luleå University of Technology, SE - 971 87 Luleå, Sweden

*Electronics and Telecommunications Research Institute (ETRI), 161 Gajeong-don, Yuseong-gu, Daejeon 305-350, Korea

details of how to solve routing problem is left open for now. Thus, we believe it to be important to investigate how routing can be done in an efficient way in intermittently connected networks. Previous work [10–13] 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 propose the use of probabilistic routing [14], using an assumption of non-random mobility of nodes to improve the delivery rate of messages while keeping buffer usage and communication overhead at a low level. This paper presents our probabilistic routing protocol, and evaluates it through simulations versus the Epidemic Routing [10] protocol.

The rest of the paper is organized as follows. Sect. II describes some related work and in Sect. III, our proposed scheme is presented. In Sect. IV the simulation setup is given, and the results of the simulations can be found in Sect. V. Finally, Sect. VI discusses some issues and looks into future work and Sect. VII concludes.

II. RELATED WORK

A. Epidemic Routing

Vahdat and Becker present a routing protocol for intermittently connected networks called Epidemic Routing [10]. This protocol relies on the theory of epidemic algorithms [15, 16] 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 it there is currently no path to the destination 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. 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.

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 do however show that by choosing an appropriate maximum hop count, delivery rates can still be kept high while the resource utilization is lower in the scenarios used in their evaluation [10].

B. Other work

A communication model that is similar to Epidemic Routing is presented by Beaufour et al. [7], focusing on data dissemination in sensor networks. The Pollen network proposed by Glance et al. [11] is also similar to Epidemic Routing. Thus, we will not go into the details of these proposals here.

Chen and Murphy propose a protocol called Disconnected Transitive Communication (DTC) [12]. It utilizes an application-tunable *utility* function to locate the node in the cluster of currently connected nodes that it is best to forward the message to based on the needs of the application. 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.

Shen et al. propose Interrogation-Based Relay Routing, a routing protocol for routing in ad hoc space networks with Scientific Earth Observing (SEO) satellites [13], characterized by frequently changing topologies, and sparse and intermittent connectivity. The satellites interrogate each other to learn more about network topology and nodal capacity to make intelligent routing decisions.

Work by Li and Rus [17] deal with a similar problem of communication in disconnected networks. They propose a solution where nodes actively change their trajectories to create connected paths to accommodate the data transmission. While this might work in military applications and in some robotic sensor networks, in most scenarios it is not likely that nodes will move just to accommodate communication of other nodes (if it is even possible to communicate the need for it).

Grossglauser and Tse looks at the utility of using the mobility of nodes to deliver messages to their destination from a slightly different point of view. One major problem with ad hoc networks is that due to interference of concurrent transmissions between nodes they scale badly. Grossglauser and Tse show the by only doing local communications between neighbors and instead relying on the movement of nodes to bring a message to its destination, this problem can be mitigated [18].

III. 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. We would like to make use of these observations and this information to improve routing performance by doing *probabilistic routing* and thus, we 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. The operation of PROPHET is similar to that of Epidemic Routing. When two nodes meet, they exchange summary vectors which in this case also contain the delivery predictability information stored at the nodes. This information is used to update the internal delivery predictability vector as described below, and then 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. III-B).

A. Delivery predictability calculation

The calculation of the delivery predictabilities have 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 elapsed 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.

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

B. 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 little system resources, but the probability of delivering a message is probably lower, and the incurred delay high.

In the evaluations in this paper, we have chosen a rather simple forwarding strategy – when two nodes meet, a message is transferred to the other node if the delivery predictability of the destination of the message is higher at the other node.

IV. SIMULATION SETUP

To aid in the evaluation of the protocol, we have developed a simple simulator. The simulator focuses on the operation of the routing protocols, and does not simulate the details of the underlying layers.

When doing an evaluation of a protocol or system, it is very important that the models used in the evaluation are realistic. 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. One mobility model that has been commonly used in evaluations of ad hoc routing protocols is the random way-point mobility model [19]. In this model, nodes randomly choose a destination and a speed and move there. Upon arrival at the destination, the node pause for a while and then chooses a new destination. This model is however likely to be unrealistic since normal users do not run around completely randomly, but rather have some set goal with their movements. Thus, it is desirable to model the mobility in some better way to better reflect reality.

Therefore, we have used two different scenarios to evaluate the protocols. Firstly, as a reference, we have used a scenario that is very similar to the one used by Vahdat and Becker in their evaluation of Epidemic Routing in [10]. This scenario consists of a $1500 \times 300\ m$ area where 50 nodes are randomly placed. These nodes move according to the random way-point mobility model [19] 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.
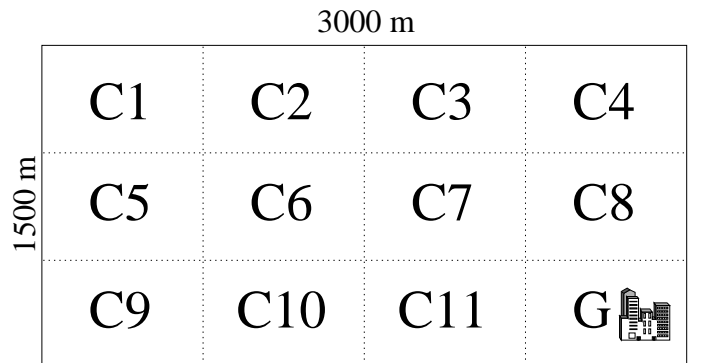
Fig. 2. Community model

However, since part of the motivation for this work was the observation that totally random mobility is not likely to be a real-

istic model, we wanted to design a more realistic scenario. This resulted in the design of a scenario that we call the "community model". This scenario consists of a $3000 \times 1500m$ area as shown in Fig. 2. 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 five 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 between 10 and 30 m/s, 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 for it to go to other places), and if it is away from home, it is very likely that it will return home. Table I 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, 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 I

DESTINATION SELECTION PROBABILITIES

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

The traffic in this scenario is also different from the random mobility scenario. 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 cease 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.

In our evaluation of the two protocols, we have focused on comparing their performance 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 long 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.

TABLE II

PARAMETER SETTINGS

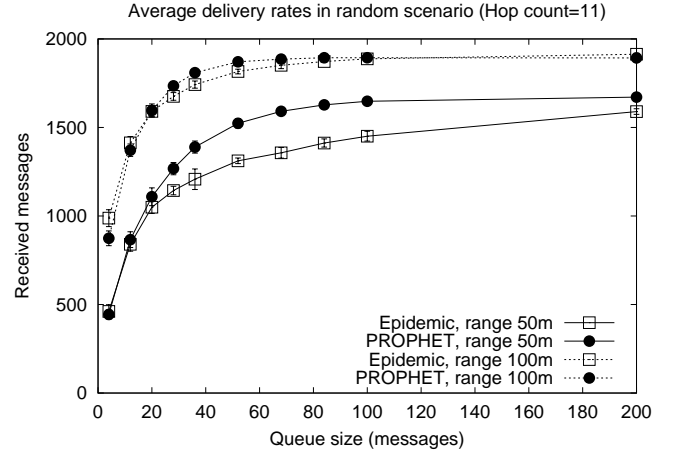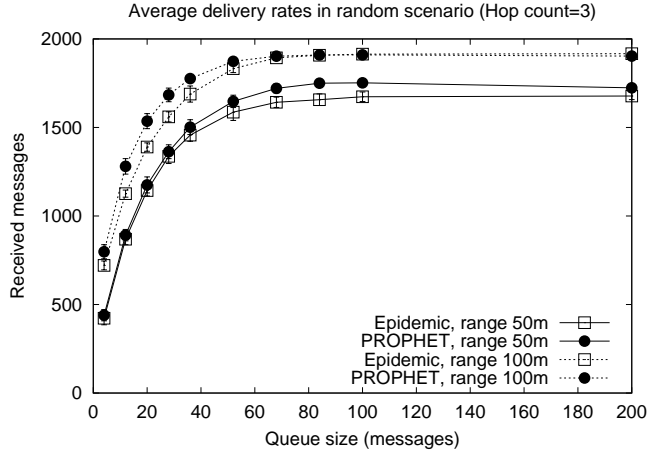| Parameter	Value
P_{init}	0.75
β	0.25
γ	0.98

We ran simulations for each scenario several times, varying the queue size at the nodes (the number of messages 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 II shows the values for parameters kept fixed in our simulations (preliminary simulations indicated those values were reasonable choices for the parameters).
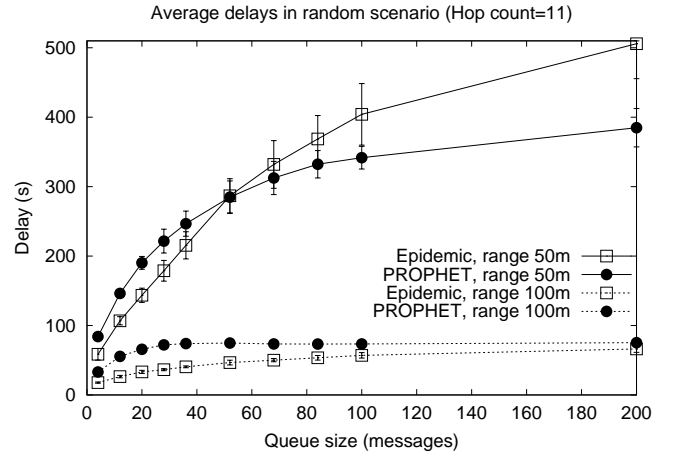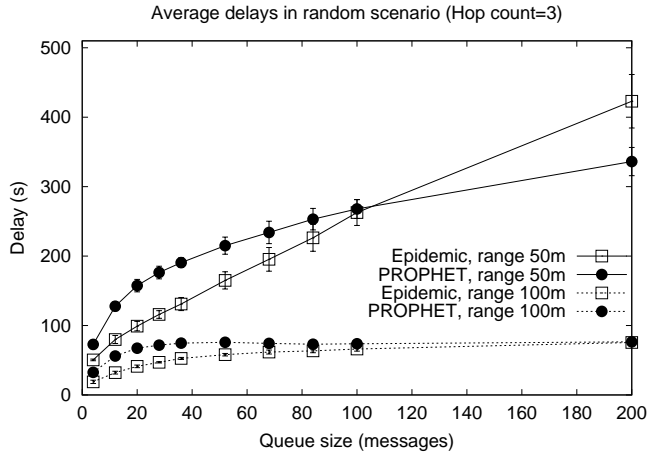
V. RESULTS

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. On the x-axis in each graph, the queue size can be found.

Figure 3 shows the results from the simulations in the random mobility scenario, and Fig. 4 from the community model simulations. First, we investigate the delivery rates of the protocols in the different scenarios, shown in Fig. 3a) and Fig. 4a). 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 have greater importance.
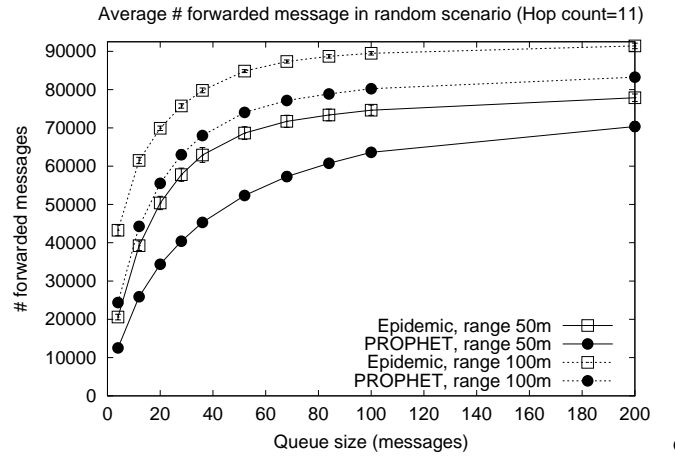
Looking at the delivery delay graphs (Fig. 3b) and Fig. 4b)), it seems like increasing the queue size, also increases the delay for messages. However, the phenomenon seen is probably not mainly that the delay increases for messages that would be delivered even at a smaller queue size (even though large 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

Average delivery rates in random scenario (Hop count=3)

Average delivery rates in random scenario (Hop count=11)

a)

Average delays in random scenario (Hop count=3)

Average delays in random scenario (Hop count=11)

b)

Average # forwarded message in random scenario (Hop count=3)

Average # forwarded message in random scenario (Hop count=11)
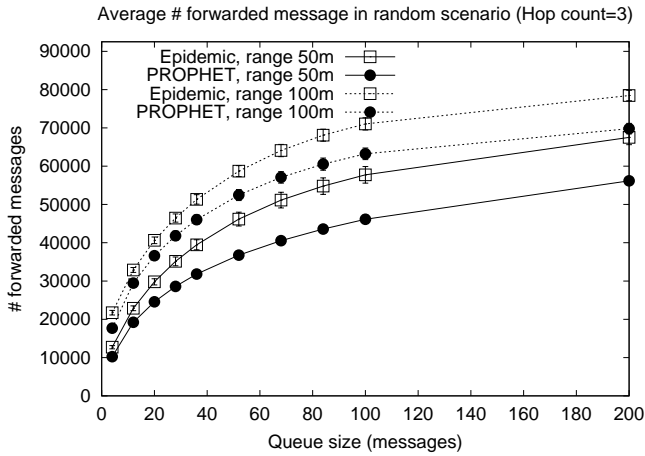
c)

Fig. 3. Simulation results from random mobility scenario

Average delivery rates in community scenario (Hop count=3)

Average delivery rates in community scenario (Hop count=11)

a)

Average delays in community scenario (Hop count=3)

Average delays in community scenario (Hop count=11)

b)

Average # forwarded message in community scenario (Hop count=3)

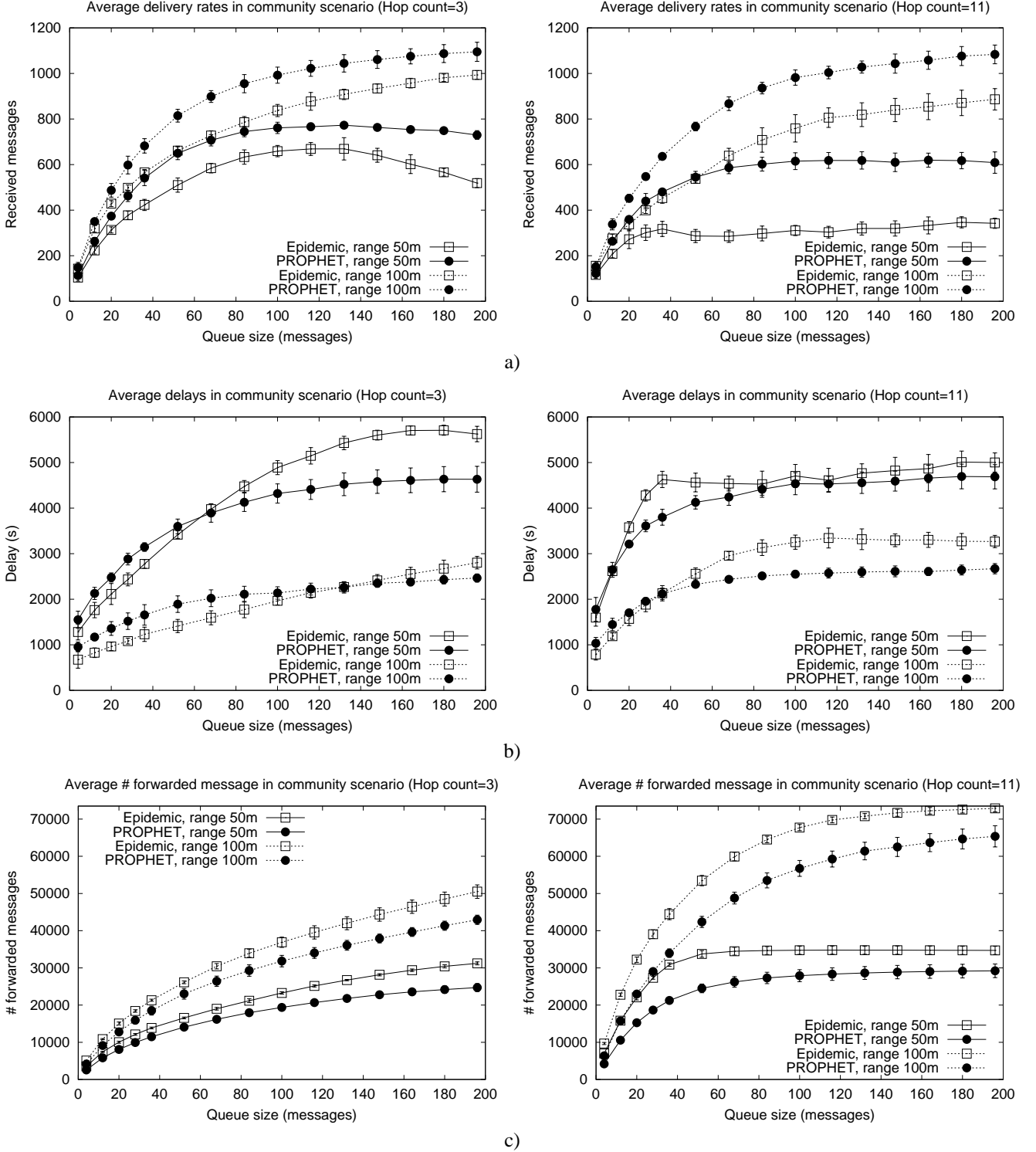Average # forwarded message in community scenario (Hop count=11)

c)

Fig. 4. Simulation results from community scenario

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. Both PROPHET and Epidemic Routing have similar delays in both scenarios, but as queue sizes grow large, PROPHET seems to have shorter delays.

Finally, looking at the graphs in Fig. 3c) and Fig. 4c), it can be clearly seen that PROPHET has a lower communication overhead and sends less 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 send 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. This is 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.

VI. 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. However, 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 remain to be done in the future. We may have taken a first step, but many more are required before we have walked a mile. Some of the issues to work on in the near future are outlined below.

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, and for example drop 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. However, it is still interesting to investigate other forwarding strategies to see if performance can be enhanced further.

VII. 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 PROPHET, a probabilistic protocol for routing in intermittently connected networks that is more sophisticated than previous protocols, using history of node encounters and transitivity 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 sometimes 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 a lower communication overhead, and better performance than existing protocols.

References

[1] Elizabeth M. Royer and Chai-Keong Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications Magazine*, vol. 6, no. 2, pp. 46–55, Apr. 1999.

[2] 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.

[3] 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.

[4] Glenn E. Prescott, Steven A. Smith, and Karen Moe, "Real-time information system technology challenges for NASA's earth science enterprise," in *Proceedings of The 20th IEEE Real-Time Systems Symposium, Phoenix, Arizona*, Dec 1999.

[5] 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*, vol. 18, no. 11, pp. 1882–1893, 2001, 18 (11).

[6] 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)*, June 2003, pp. 233–244.

[7] 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.

[8] Anthony D. Joseph, Joshua A. Tauber, and M. Frans Kaashoek, "Mobile computing with the rover toolkit," *IEEE Transactions on Computers*, vol. 46, no. 3, pp. 337–352, 1997.

[9] Vinton G. Cerf, Scott C. Burleigh, Adrian J. Hooke, Leigh Torgerson, Robert C. Durst, Keith L. Scott, Kevin Fall, and Howard S. Weiss, "Delay-tolerant network architecture," Internet Draft draft-irtf-dtnrg-arch-02.txt (work in progress), March 2003.

[10] Amin Vahdat and David Becker, "Epidemic routing for partially connected ad hoc networks," Technical Report CS-200006, Duke University, April 2000.

[11] Natalie Glance, Dave Snowdon, and Jean-Luc Meunier, "Pollen: using people as a communication medium," *Computer Networks*, vol. 35, no. 4, pp. 429–442, March 2001.

[12] 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)*, aug 2001, pp. 21–27.

[13] 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.

[14] 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.

[15] Alan Demers, Dan Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry, "Epidemic algorithms for replicated database maintenance," in *Proceedings of the ACM Symposium on Principles of Distributed Computing*, 1987, pp. 1–12.

[16] Werner Vogels, Robbert van Renesse, and Ken Birman, "The power of epidemics: Robust communication for large-scale distributed systems," in *Proceedings of First Workshop on Hot Topics in Networks (HotNets-I), 28-29 October 2002, Princeton, New Jersey, USA*, oct 2002.

[17] Qun Li and Daniela Rus, "Communication in disconnected ad-hoc networks using message relay," *Journal of Parallel and Distributed Computing*, 2003.

[18] Matthias Grossglauser and David Tse, "Mobility increases the capacity of ad-hoc wireless networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, Aug 2002.

[19] Tracy Camp, Jeff Boleng, and Vanessa Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.