

Research Article

An Improved PROPHET Routing Protocol in Delay Tolerant Network

Seung Deok Han and Yun Won Chung

School of Electronic Engineering, Soongsil University, 511 Sangdo-dong, Dongjak-gu, Seoul 156-743, Republic of Korea

Correspondence should be addressed to Yun Won Chung; ywchung@ssu.ac.kr

Received 20 August 2014; Accepted 5 October 2014

Academic Editor: Haythem Bani Salameh

Copyright © 2015 S. D. Han and Y. W. Chung. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In delay tolerant network (DTN), an end-to-end path is not guaranteed and packets are delivered from a source node to a destination node via store-carry-forward based routing. In DTN, a source node or an intermediate node stores packets in buffer and carries them while it moves around. These packets are forwarded to other nodes based on predefined criteria and finally are delivered to a destination node via multiple hops. In this paper, we improve the dissemination speed of PROPHET (probability routing protocol using history of encounters and transitivity) protocol by employing epidemic protocol for disseminating message m , if forwarding counter and hop counter values are smaller than or equal to the threshold values. The performance of the proposed protocol was analyzed from the aspect of delivery probability, average delay, and overhead ratio. Numerical results show that the proposed protocol can improve the delivery probability, average delay, and overhead ratio of PROPHET protocol by appropriately selecting the threshold forwarding counter and threshold hop counter values.

1. Introduction

In traditional data networks, such as Internet, at least one continuous end-to-end path is guaranteed between a source node and a destination node, and packets are delivered from a source node to a destination node through one of the available paths. In delay tolerant network (DTN), however, an end-to-end path is not guaranteed and packets are delivered from a source node to a destination node via store-carry-forward based routing [1–7]. In DTN, a source node or an intermediate node stores packets in buffer and carries them while it moves around. These packets are forwarded to other nodes based on predefined criteria and finally are delivered to a destination node via multiple hops. A lot of attention has been paid to DTN for possible uses in disconnected network environments, especially for extreme cases such as inter-planetary communications [2] and disaster scenarios [7].

One of the representative DTN routing protocols is epidemic routing protocol [8, 9]. As the name implies, a source node forwards a message to all the neighbor nodes whenever it contacts neighbor nodes, like “epidemic.” This simple routing protocol is very powerful even when the buffer

size of nodes is sufficient. However, if the buffer size is not sufficient, especially as in mobile nodes, epidemic protocol generates significant message overhead and the performance degrades.

In order to solve the message overhead problem of the epidemic protocol, several schemes have been proposed, such as Spray and Wait protocol [10, 11] and PROPHET protocol [12, 13]. In these protocols, the total number of message copies present in a network is limited by a certain number or message forwarding is carried out only when a certain condition is met.

The Spray and Wait protocol consists of Spray phase and Wait phase and the total number of message copies present in a network is limited by L . In Spray phase, L message copies are disseminated to other nodes until there is no node which has more than one message copy. Then, the protocol transitions into Wait phase and the message copy is delivered to a destination node only. In order to enhance the dissemination speed of message copies of basic Spray and Wait protocol, binary Spray and Wait protocol [11] was proposed, where half of stored message copies are distributed to another node. In Spray and Wait protocol, however, message copies are

distributed blindly without considering the characteristics of receiving nodes.

In PROPHET (probability routing protocol using history of encounters and transitivity) protocol, delivery predictability between two nodes is calculated based on contact history between them, where higher delivery predictability implies a higher probability of future contacts between them. In PROPHET protocol, a message is copied to a contact node only when the delivery predictability to a destination node of the contact node is larger than that of the transmitting node. By doing this, PROPHET protocol achieves good delivery probability as well as satisfying low overhead.

In PROPHET protocol, however, we note that the dissemination speed of a message is relatively low, since a message is copied only when a delivery predictability condition is met, and this results in longer average delay and low delivery probability when the buffer size is sufficient. In this paper, we improve the dissemination speed of PROPHET protocol by employing epidemic protocol for disseminating message m , if the conditions for the forwarding counter and the hop counter of message m are met. Then, we show that the proposed routing protocol achieves higher delivery probability and lower average delay than PROPHET protocol.

The remainder of this paper is organized as follows. In Section 2, related works are surveyed. In Section 3, the detailed algorithm of the proposed routing protocol is described. In Section 4, numerical examples are presented using simulation from the aspect of delivery probability, average delay, and overhead. Finally, conclusions and further works are drawn in Section 5.

2. Related Works

In related works, works on epidemic and PROPHET protocols, which constitute the proposed routing protocol, are surveyed in detail.

2.1. Epidemic Protocol. Epidemic protocol is basically a flooding-based protocol, and, thus, a node with messages copies them to any other contact nodes if they do not have them already. To do this, two nodes firstly exchange summary vectors which contain the list of messages they have when they contact each other. Then, each node checks the list of messages which it does not have yet and requests the messages from the other node.

If buffer size is infinite, epidemic routing protocol can achieve optimal delivery probability and average delay. Since buffer size is finite and epidemic protocol generates significant copies of a message, enhanced schemes to manage limited buffer and battery energy have been proposed. As an example, in an energy efficient n -epidemic routing protocol [14], a node copies messages to other nodes only when the number of neighbor contact nodes reaches a predefined threshold value, that is, n , in order to save energy.

In [15], the authors studied the performance of four categories of epidemic routing protocol in detail, that is, P-Q epidemic, epidemic with time-to-live (TTL), epidemic with encounter counter (EC), and epidemic with immunity table using trace-file and random waypoint mobility models.

Then the authors proposed three enhanced schemes such as dynamic TTL, EC + TTL, and cumulative immunity and showed that the enhanced schemes can improve delivery probability and buffer occupancy level can be reduced significantly for cumulative immunity scheme.

2.2. PROPHET Protocol. PROPHET protocol uses nonrandom mobility and contact patterns in real application scenarios to copy messages to other nodes in order to improve the routing performance [13]. That is, the PROPHET protocol is based on the fact that if a node has visited a location or contacted with a node frequently, the probability of visiting the location and contacting the node is higher. To achieve this, "delivery predictability" is defined at a node for every other contacted node. The delivery predictability of node A to node B is denoted by $P_{A,B}$ and the range of delivery predictability value is defined as $0 \leq P_{A,B} \leq 1$. If node A with a message to a destination node D contacts with node B, node A and node B exchange summary vectors and delivery predictability. Then, node A compares $P_{A,D}$ and $P_{B,D}$. If $P_{B,D} > P_{A,D}$, the message to destination node D is copied to node B. Otherwise, the message is not copied to node B.

In PROPHET+ [16], deliverability is defined as a weighted sum of buffer size, battery power, location, popularity, and the delivery predictability. If a node meets another node, it queries deliverability value of another node. Then, if the deliverability value of another node is higher than a predefined threshold value, a source node sends a message to another node. If there are multiple nodes which are in simultaneous contact with the source node, the message is sent to the node with the highest deliverability value. In PROPHET+, it was shown that the proposed PROPHET+ performs well from the aspect of delivery probability and average delay, by appropriately choosing the weight factor for buffer size, battery power, location, popularity, and the delivery predictability.

In [17], a policy of history of message's movement was newly considered and a new probabilistic routing protocol based on history of message was proposed. In the proposed protocol, message's hop count as well as delivery predictability is considered to determine next hop node, where the history of message's traversed path is defined as a sequence of contacted nodes.

In distance-based PROPHET [18], distance between two nodes was additionally used to compute delivery predictability. That is, each node having a message checks distance from neighboring nodes and chooses a node located in a smaller distance as a forwarder since a node can have higher transmission rate to a nearer node and, thus, can increase the delivery probability and decrease delivery delay. In [19], the authors extended the results in [18] by considering community mobility model, in addition to random waypoint mobility model in [18].

3. An Improved PROPHET Routing Protocol

The proposed improved PROPHET routing protocol is a hybrid of epidemic protocol and PROPHET protocol. The main idea of the proposed protocol is to accelerate the dissemination of messages in the early phase of message delivery,

```

At the generation of message  $m$ ,  $n_m$  and  $h_m$  are initialized as 0
if Node  $i$  contacts node  $j$  then
  Node  $i$  and Node  $j$  exchange summary vectors
  Node  $i$  decides candidate message set  $M$  to transmit
  for all messages  $\in M$  do
    if  $n_m \leq N_m$  then
      if  $h_m \leq H_m$  then
         $n_m++$ 
        Node  $i$  delivers message  $m$  using Epidemic routing with  $n_m$  and  $h_m++$ 
      else
        Node  $i$  delivers message  $m$  using PROPHET routing with  $n_m$  and  $h_m++$ 
      end if
    else
      Node  $i$  delivers message  $m$  using PROPHET routing with  $n_m$  and  $h_m++$ 
    end if
  end for
end if

```

ALGORITHM 1: Detailed algorithm of the proposed scheme.

by employing epidemic protocol. On the other hand, the proposed protocol restricts dissemination in later phase since it only copies messages to other nodes only when a delivery predictability condition is met.

Before going into the detailed operation, new notations are defined as follows.

- (i) n_m is forwarding counter of message m at the current node, which is defined as the total number of messages copied to other nodes for the message m along the forwarding path from a source node to the current node. At the generation of message m , the value of n_m is initialized at 0. If the current node is a source node, n_m for a message m means that the source node forwarded n_m copies to other nodes. If the current node is not a source node, the initial value of n_m at the current node was set to the current n_m value of the previous transmitting node and n_m was increased by one for each message copy to another node from the current node.
- (ii) N_m is threshold forwarding counter of message m .
- (iii) h_m is hop counter of message m at the current node, which is defined as the total number of hops that a message m has traversed along the forwarding path from a source node to the current node. If the current node is a source node, h_m for a message m is initialized at 0.
- (iv) H_m is threshold hop counter of message m .

Algorithm 1 shows the detailed algorithm of the proposed protocol. At the generation of message m , both the values of n_m and h_m are initialized as 0. When node i contacts node j , they exchange summary vectors which contain message list stored in each node. Then node i decides candidate message set M to transmit to node j . After selecting a message $m \in M$, node i checks the relationship between n_m and N_m . If $n_m > N_m$, PROPHET routing protocol is used to deliver the message

m . Otherwise, node i checks again the relationship between h_m and H_m . If $h_m > H_m$, PROPHET routing protocol is used to deliver the message m , too. Then, message m is delivered using PROPHET routing protocol and if it is delivered to another node, n_m of the receiving node is initialized at the current value of n_m of the transmitting node and h_m of the receiving node is increased by one (h_m++). Otherwise, that is to say, if $n_m \leq N_m$ and $h_m \leq H_m$, node i uses epidemic routing protocol to deliver the message m . Then n_m of the transmitting node is increased by one (n_m++) and the increased n_m and h_m++ information is delivered to the receiving node too. The loop of delivering a message in M repeats until all the messages are delivered based on either epidemic or PROPHET routing protocol.

Figure 1 shows an example scenario of the proposed protocol in DTN environment, where $H_m = 5$ and $N_m = 10$. Suppose that node i has four messages, that is, m_1 , m_2 , m_3 , and m_4 , and node j has one message m_4 . If they are within the contact of each other, they exchange summary vectors which contain message list stored in each node and delivery predictability information. In the considered scenario in Figure 1, node i determines that messages m_1 , m_2 , and m_3 should be delivered to node j . Then, node i delivers m_1 and m_2 using epidemic protocol since $m_i \leq M_i$ and $h_i \leq H_i$ for $i = 1, 2$, and the values of n_1 and n_2 are increased by one in both nodes i and j . Also, the values of h_1 and h_2 in node j are increased by one from those in node i . Node i delivers m_3 using PROPHET protocol since $n_3 > N_3$. Since delivery predictability to destination node C in node j is larger than that in node i , the message m_3 is delivered to node j using PROPHET protocol and h_3 in node j is increased from that in node i .

4. Numerical Examples

In this paper, we carried out simulation for the proposed protocol using the opportunistic network environment (ONE) simulator developed by Helsinki University [20, 21]. In

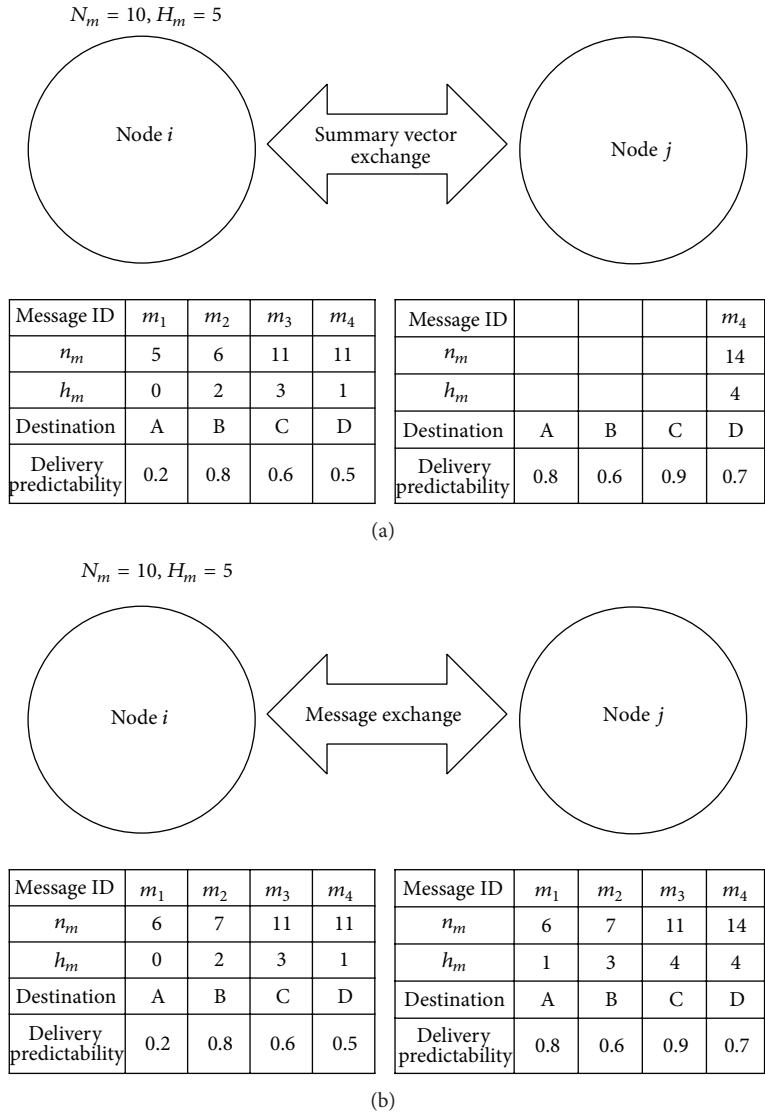


FIGURE 1: Example scenario of the proposed protocol ((a) summary vector exchange, (b) message exchange).

numerical examples, we analyzed the performance of the proposed protocol, from the aspect of delivery probability, average delay, and overhead ratio, for varying the values of threshold forwarding counter and threshold hop counter. The delivery probability is defined as the number of successfully delivered messages divided by the number of created messages. The average delay is the average value of delay for successfully delivered messages. The overhead ratio is defined by $(NR - ND)/ND$, where NR is the number of relayed messages and ND is the number of successfully delivered messages. The parameter values are assumed, as in Table 1.

4.1. The Effect of Threshold Forwarding Counter N_m . Figure 2 shows the delivery probability for varying the values of N_m , where $H_m = 0, 10$, and 20 . The delivery probability of the proposed protocol when $H_m = 0$ and 10 is higher than that of PROPHET and epidemic protocols, which are constant since they do not depend on the values of N_m , for most values of

N_m . This is because the proposed protocol employs epidemic protocol to spread the message copies quickly until the forwarding counter value reaches threshold forwarding counter value and employs PROPHET protocol to deliver message copies to better nodes with higher delivery predictability to destination nodes after the forwarding counter value reaches threshold forwarding counter value. However, the delivery probability of the proposed protocol when $H_m = 20$ is lower than that of PROPHET protocol, since too many message copies result in message drop in buffer and, thus, reduce delivery probability compared to PROPHET protocol. The delivery probability of the proposed protocol when $H_m = 20$ is, however, still higher than that of epidemic protocol, since the number of message copies is still smaller than that of epidemic protocol. The delivery probability of the proposed protocol saturates as the value of N_m becomes large since the effect of increasing the value of N_m for large values of N_m is negligible.

TABLE 1: Parameter values.

Parameter	Value
Area size (m ²)	4,500 × 3,400
Simulation times (s)	21,600
Transmission range (m)	10
Transmission rate (kbps)	250
Movement model	Tram: MapRouteMovement Car, pedestrian: ShortestPathMapBasedMovement
Speed of nodes (m/s)	Tram: $U[5, 14]$ Car: $U[2.7 : 13.9]$ Pedestrian: $U[0.5 : 1.5]$
Buffer size (bytes)	Car, pedestrian: 5 M Tram: 50 M
Message generation intervals (s)	$U[25, 35]$
Message size (kbytes)	$U[500, 1,000]$
Number of nodes	Tram: 6 Car: 40 Pedestrian: 80

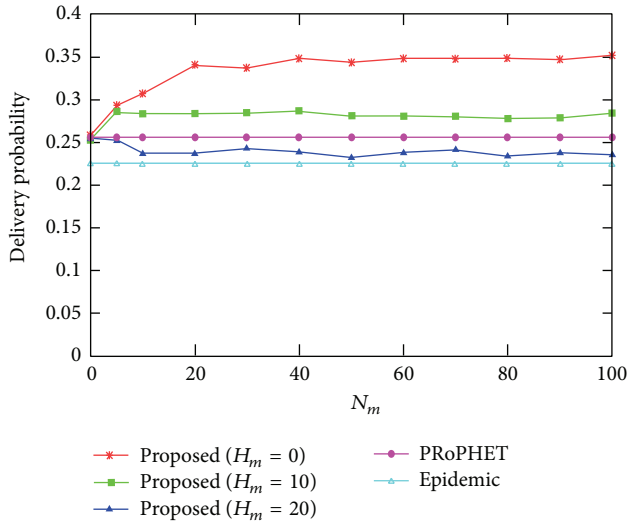


FIGURE 2: Delivery probability for varying the value of N_m .

Figure 3 shows the average delay for varying the values of N_m , where $H_m = 0, 10$, and 20 . The average delay of the proposed protocol is smaller than that of PRoPHET and epidemic protocols for most values of N_m , since the proposed protocol reduces the average delay of PRoPHET protocol by employing epidemic protocol in source node, but the packet is delivered more quickly than epidemic protocol since the proposed protocol uses delivery predictability for better delivery and uses more message copies for fast delivery compared to PRoPHET protocol. Similar to Figure 2, the effect of N_m on the average delay becomes negligible as the value of N_m becomes large.

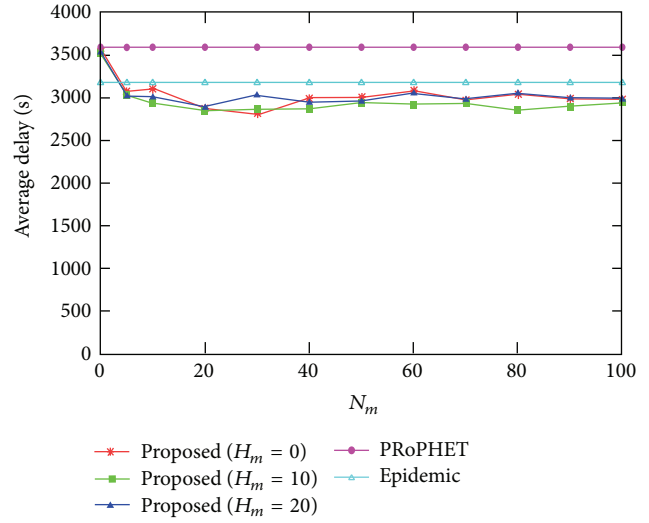


FIGURE 3: Average delay for varying the value of N_m .

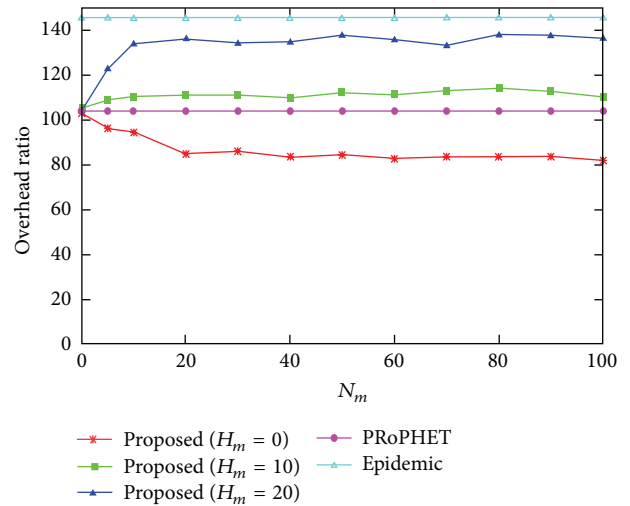


FIGURE 4: Overhead ratio for varying the value of N_m .

Figure 4 shows the overhead ratio for varying the values of N_m , where $H_m = 0$. The overhead ratio of the proposed protocol when $H_m = 0$ is smaller than that of both PRoPHET and epidemic protocols, since the effect of the increased number of delivered messages of the proposed protocol is higher than that of the increased number of relayed messages of the proposed protocol for $H_m = 0$. However, the overhead ratio of the proposed protocol when $H_m = 10$ and 20 is higher than that of PRoPHET protocol since the effect of the increased number of relayed messages of the proposed protocol is higher than that of the increased number of delivered messages of the proposed protocol. Similar to Figures 2 and 3, the effect of N_m on the overhead ratio becomes negligible as the value of N_m becomes large.

4.2. *The Effect of Threshold Hop Counter H_m .* Figure 5 shows the delivery probability for varying the values of H_m , where $N_m = 5, 10, 20$, and 40 . The delivery probability of the

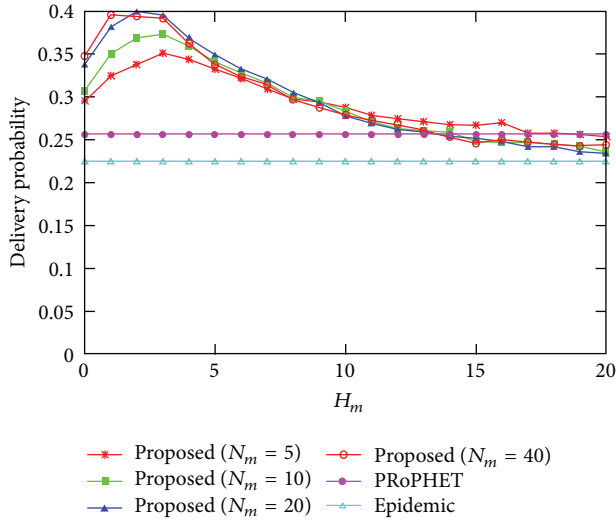


FIGURE 5: Delivery probability for varying the value of H_m .

proposed protocol is higher than that of epidemic protocol always. Also, the delivery probability of the proposed protocol is higher than that of PRoPHET protocol for most values of H_m ; that is, $H_m \leq 15$, and it is slightly lower than that of PRoPHET protocol for large values of H_m , since too many message copies result in message drop in buffer and thus, reduces delivery probability of the proposed protocol. The delivery probability of the proposed protocol increases as H_m increases from $H_m = 0$ and decreases as H_m increases from $H_m = 2$ or 3 in the considered parameter setting. This is because increasing the spreading of message copies from $H_m = 0$ to $H_m = 2$ has a positive effect on increasing delivery probability due to higher message copies but the increasing the spreading of message copies too high, that is, from $H_m = 2$ or 3, results in higher buffer occupancy and this results in message drop. The delivery probability of the proposed scheme when the values of H_m are small increases as N_m increases but it is saturated for large values of H_m .

Figure 6 shows the average delay for varying the values of H_m , where $N_m = 5, 10, 20$, and 40. Similar to Figure 3, the average delay of the proposed protocol is smaller than that of PRoPHET and epidemic protocols for most values of H_m , based on a similar rationale as in Figure 3. From Figure 6, it can be shown that the effect of different values of N_m on the average is not significant.

Figure 7 shows the overhead ratio for varying the values of H_m , where $N_m = 5, 10, 20$, and 40. The overhead ratio of the proposed protocol is always smaller than that of epidemic protocol, since the proposed protocol generates smaller message copies. The overhead ratio of the proposed protocol is smaller than that of PRoPHET protocol for small values of H_m ; that is, $H_m \leq 8$, since the effect of the increased number of delivered messages of the proposed protocol is higher than that of the increased number of relayed messages of the proposed protocol for $N_m = 5, 10, 20$, and 40. However, the overhead ratio of the proposed protocol is higher than that of PRoPHET for large values of H_m , since the effect of the increased number of relayed messages of the proposed

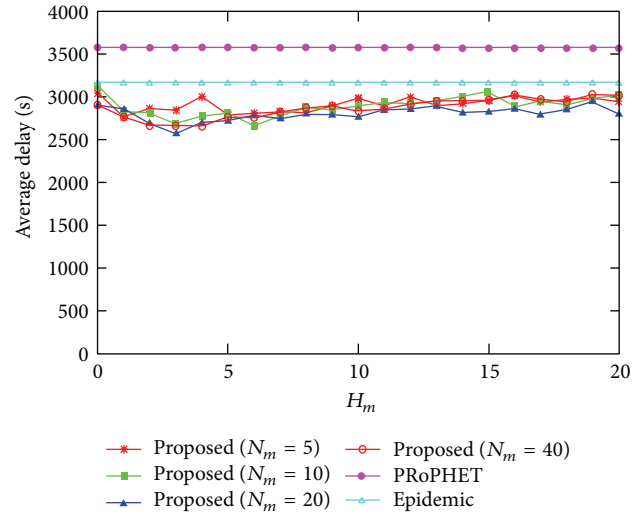


FIGURE 6: Average delay for varying the value of H_m .

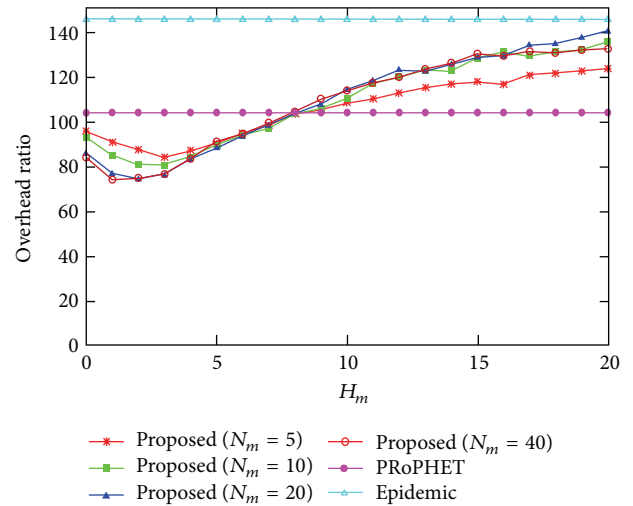


FIGURE 7: Overhead ratio for varying the value of H_m .

protocol is higher than that of the increased number of delivered messages of the proposed protocol with $N_m = 5, 10, 20$, and 40.

5. Conclusions and Further Works

In this paper, we improved the dissemination speed of PRoPHET protocol by employing epidemic protocol if forwarding counter and hop counter values are smaller than or equal to the threshold values. Then, the performance of the proposed protocol was analyzed from the aspect of delivery probability, average delay, and overhead ratio for varying the values of threshold forwarding counter and threshold hop counter using ONE simulator. Numerical results show that the proposed protocol can improve the delivery probability, average delay, and overhead ratio of PRoPHET protocol, by appropriately selecting the threshold forwarding counter and threshold hop counter values. As further works, we

will propose an adaptive selection of threshold forwarding counter and threshold hop counter based on measured network parameter values to maintain good performance for varying network environments always.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '03)*, pp. 27–34, ACM Press, New York, NY, USA, 2003.
- [2] S. Burleigh, A. Hooke, L. Torgerson et al., "Delay-tolerant networking: An approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.
- [3] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 1, pp. 24–37, 2006.
- [4] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervelló-Pastor, "From delay-tolerant networks to vehicular delay-tolerant networks," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 1166–1182, 2012.
- [5] M. J. Khabbaz, C. M. Assi, and W. F. Fawaz, "Disruption-tolerant networking: a comprehensive survey on recent developments and persisting challenges," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 607–640, 2012.
- [6] Y. Cao and Z. Sun, "Routing in delay/disruption tolerant networks: a taxonomy, survey and challenges," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 2, pp. 654–677, 2013.
- [7] A. M. Campillo, J. Crowcroft, E. Yoneki, and R. Martí, "Evaluating opportunistic networks in disaster scenarios," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 870–880, 2013.
- [8] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep. CS-200006, Duke University, Department of Computer Science, Durham, NC, USA, 2000.
- [9] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance modeling of epidemic routing," *Computer Networks*, vol. 51, no. 10, pp. 2867–2891, 2007.
- [10] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the ACM SIGCOMM 2005 Workshop on Delay-Tolerant Networking (WDTN '05)*, pp. 252–259, August 2005.
- [11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77–90, 2008.
- [12] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.
- [13] A. Lindgren, A. Doria, E. Davies, and S. Grasic, "Probabilistic routing protocol for intermittently connected networks," IETF RFC 6683, August 2012.
- [14] X. Lu and P. Hui, "An energy-efficient n-epidemic routing protocol for delay tolerant networks," in *Proceedings of the 5th IEEE International Conference on Networking, Architecture and Storage (NAS '10)*, pp. 341–347, IEEE, Macau, China, July 2010.
- [15] Z. Feng and K.-W. Chin, "A unified study of epidemic routing protocols and their enhancements," in *Proceedings of the 26th IEEE International Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW '12)*, pp. 1484–1493, Shanghai, China, May 2012.
- [16] T.-K. Huang, C.-K. Lee, and L.-J. Chen, "PROPHET+: an adaptive PROPHET-based routing protocol for opportunistic network," in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA '10)*, pp. 112–119, April 2010.
- [17] F. C. Lee and C. K. Yeo, "Probabilistic routing based on history of messages in delay tolerant networks," in *Proceedings of the IEEE Vehicular Technology Conference (VTC Fall '11)*, pp. 1–6, San Francisco, Calif, USA, September 2011.
- [18] P. Sok and K. Kim, "Distance-based PROPHET routing protocol in disruption tolerant network," in *Proceedings of the International Conference on ICT Convergence (ICTC '13)*, pp. 159–164, Jeju, Republic of Korea, October 2013.
- [19] P. Sok, S. Tan, and K. Kim, "PROPHET routing protocol based on neighbor node distance using a community mobility model in delay tolerant networks," in *Proceedings of the IEEE International Conference on High Performance Computing and Communications*, pp. 1233–1240, November 2013.
- [20] "The opportunistic network environment simulator," <http://www.netlab.tkk.fi/tutkimus/dtn/theone/>.
- [21] A. Keranen, J. Ott, and T. Karkkainen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, vol. 55, pp. 1–10, March 2009.