OOPProPHET: A New Routing Method to Integrate the Delivery Predictability of ProPHET-Routing with OOP-Routing in Delay Tolerant Networks

Shih-Yang Yang
Kang-Ning Junior College of Medical Care and Management, Taipei, Taiwan
shihyang@knjc.edu.tw

Jiun-Ting Jiang
Dept of Computer Science and Information Engineering, Tamkang University, Taiwan
rjrj0510@gmail.com

Po-Zung Chen
Dept of Computer Science and Information Engineering, Tamkang University, Taiwan
pozung@mail.tku.edu.tw

***Abstract*—To balance the tradeoff of the routing throughput in DTNs, we try to propose a new method to integrate the delivery predictability of ProPHET-Routing with our previous OOP-Routing, called OOPProPHET-Routing. Also, we have used the NS2 network simulator to verify our proposed routing method is better than before, and we have achieved a good tradeoff between the delivery ratio and the numbers of forwarding packets. In particular, when the size of buffer-queue is not enough or when the numbers of forwarding packets of Epidemic-Routing is tremendously increasing, the delivery ratio of this routing can easily exceed Epidemic. In other case, the delivery ratio is close to Epidemic but the amounts of forwarding packets are less than one-tenth. Moreover, we vary the value of the threshold to examine the effect of adjustment.**

***Index Terms*—DTN, Routing, Epidemic, NS2, ProPHET, OOP**

I. INTRODUCTION

After the rapid growth of wireless networks, more and more charming routing methods have been developed in the Mobile Ad Hoc Networks. In these conventional routing, we need to assume we can find an end-to-end path. If the topologic of network is sparser or the speed of mobile node is faster, it often happens to the problems of intermittent connection.

In order to solve the communication problems, the most common solution is to use SCF (store, carry and forward) mechanism. The first routing method to conquer these problems is Epidemic Routing [3]. This routing uses the flooding method to replicate the message to all neighbor nodes which do not connect with the message; furthermore, it uses an anti-entropy session to reduce overhead. In the practical application, however, this is still insufficient because the numbers of forwarding packets are still too numerous. Plus, the size of buffer-queue is not adequate to cause the ratio of delivery packet decreasing. We need more efforts.

In contrast of the traditional MANET routing methods, most of the DTN-routings rely on replication to increase opportunities and use the information between the hop to make decisions, so it will be faced with greater challenges. If we can not have special devices or other special nodes, there are two categories routing methods in DTN.

The first is multi-copy, try to control the amount of replication [10], the second is single-copy, try to increase the node completely knowledge [11]. In addition, from the survey [8] has pointed out the knowledge about the network helps in deciding the best next hop. To develop the routing in the absolute zero knowledge network will be a greater challenge!

Therefore, we have two motivations. The first, we try to figure out a way to combine the advantages of single-copy and multi-copy routing to reach a high of delivery ratio in less replication. The second, we also found "history tends to repeat itself" in our lives. For example, peoples will go daily route to work or the wild animals will drink water in the same lake. So, we try to accumulate history information to increase the potential knowledge in zero knowledge networks.

Our goal is integrate the history information of ProPHET-Routing with our previous routing to produce a new routing and this new routing is more suitable for application in a small size of buffer-queue.

We organize the rest of the paper as follows: in Section 2, we discuss the related routing methods. In Section 3, we describe in detail the mechanisms of the OOPProPHET-Routing Scheme. In Section 4, we

doi:10.4304/jcp.8.7.1656-1663

prepare two scenarios, including Random-Waypoint-Mobility and Community-Mobility, furthermore, used NS2 to simulate and evaluate the performance of our OOPProPHET-Routing. Finally, we conclude this paper and provide some directions for future research in Section 5.

II. Related Work

A. DTN Problem and Motivation

For explaining the delay tolerance characteristics of the network, we prepare an example of evolution over time, like the intermittent links as in [6]. We concern about the Figure 1. The routing mission is to let any node can successfully send a message to any other node.

Suppose at time 8:55AM, there is a message should be send from node S to node D and the distance of communication is about from node S to node 1.

At 8:55AM, node S will smartly send the message to node 1. After 5 minutes, node 1 will send the message to node 2. At last, node 2 will send the message to node S, and will achieve this routing mission.

We can pay attention to this fact: there is no end-to-end path from node S to node D and there is a smartly decision to send the message on node S to node 1.

How smart to select node 1 as an intermediate node? Therefore, node 1 will send this message to the node D in the future. This is a big challenge in DTNs. This is the motivation that we want to face this challenge.

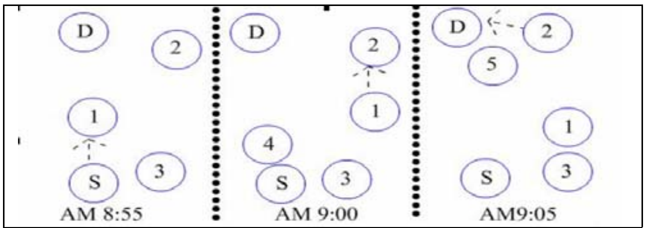


Figure 1. Example of the intermittent connect network

As with this problem, traditional routing protocols, such as AODV and DSR, failed to carry out the task because they cannot find an end-to-end path. Even when the path repair mechanism still fails due to the delay time is still too long. We must use replication mechanism to solve the problem.

In the DTN environment, in view of the number of replication, the routing methods can be classified into three categories;

The first is completely a replication method: the best-known method in multi-copy routing method [10] is Epidemic [3]. But, the method will generate too many forwarding messages and cause useful-packets may be dropped. Therefore, many scholars try to reduce unnecessary packet replication; even in 2010, scholars continually study the modified Epidemic [7].

The second category is part of the copy method: a representative method is the ProPHET [1], it attempts to maintain the history information, in order to reduce the amount of copies. Lately, there are some scholars continue to develop similar methods in [2] [12].

The third category is a single-copy method: in routing process, there is only one copy message [11]. These methods are very suitable for resource-constrained scenarios. In this casd, there is a common difficult challenge is to design an appropriate utility function while maintain a sufficiently high delivery ratio.

B. ProPHET-Routing

The routing method attempts to maintain information and use the property of transitivity to increase its delivery rate, named ProPHET, a Probabilistic ROuting Protocol using History of Encounters and Transitivity. The operation of ProPHET is similar as Epidemic Routing. When two nodes meet, they will exchange summary vectors which in this case also contain the delivery predictability information stored at the nodes. This information will be used to update the internal delivery predictability vector.

In order to calculate the delivery predictability, there are three mathematical equations:

In equation (1), where P represents a probability of encounter, and Pinit $\in$ [0,1] is an initialization constant, furthermore, Pinit is the ratio of increment when two nodes encountered. It is conducted (1) to gradually increase the metric of encounter. Equation (2), where $\gamma \in [0,1)$ is the aging constant, to gradually decrease the metric of encounter. Equation(3), where $\beta \in [0,1]$ is a scaling constant that decides how great impact the transitivity should have on delivery predictability.

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

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

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

About these parameters, some researches already improve in [4]. For example, lower the value of Pinit or avoid two nodes repeatedly contact problem which will increase the value of P. In this paper, we only use the primitive setting.

C. OOP-Routing

We have proposed a routing protocol in previous paper. The routing is based on one-broadcast one-copy persistent-storage in delay tolerant mobile ad hoc networks, called OOP-Routing [5].

A brief description of the characteristics of this routing is the number of delivery ratio close to Epidemic, but the overhead is much less.

The full name of OOP is One Broadcast, One Copy and Persistent Storage. Mean that, there are three steps in the OOP-Routing to process a message. These three steps are referred to as OB, OC and PS. We can refer to Fig. 2 ~ Fig. 5.

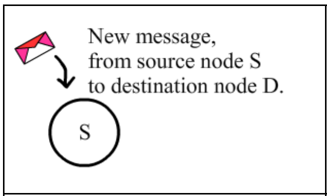


Figure 2. A new message is entering the system

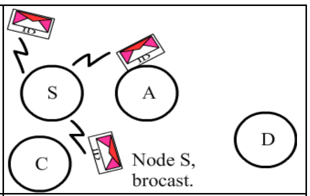


Figure 3. Step 1 (OB process) Just only one broadcast

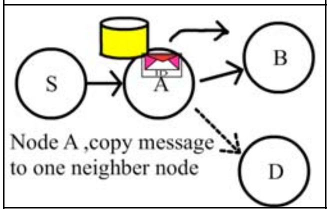


Figure 4. Step 2(OC process) Just only one copy

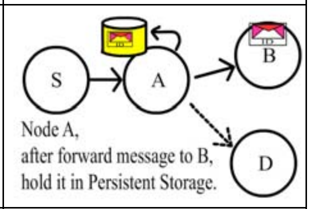


Figure 5. Step 2 (PS process) Hold on the message in PS

1) OB (One Broadcast) process:

When a new message is entering the system, referring to Fig. 2, the first phase will process for just only one broadcast shown in Fig. 3. Whether someone can receive the messages or not, the source node always stays in waiting state and waits neighbor nodes. If more than one node receives the message, these nodes can simultaneously process forwarding. As a result, we can get advantages of multi-copy.

2) OC (One Copy) process:

When a node holds a message, referring to Fig. 3, some neighbor nodes are linked up with it. The node can then select the best one node to copy. There is a simply selective rule of which method is randomly choosing only one to duplicate. Therefore, we can control the number of multi-copy packets are decreased. We can get the advantages of single-copy.

3) PS (Persistent Storage) process:

When node is forwarding the message to other nodes, the node must hold the message in Persistent Storage before the custody transfer, referring to Fig. 4. We use the Persistent Storage to increase the opportunity of sending directly to the destination node. We use this simple process, not only solve the HOL (head-of-line blocking) problem, but also propose an important idea to improve other single-copy routing.

In a word, by the above procedures, the most significant feature of this routing can catch the both advantages of multi-copy routing and of single-copy routing. Thus, the performance will be better improved than Epidemic-Routing.

III. OOPProPHET-Routing Scheme

In the previous section, we have addressed the OOP-Routing, and we can find a challenging problem in OC-process. We look at the Fig. 6, suppose there are more than one neighbor nodes in source node S, such as node B, C and D. How do we choose the best node to forward the message for increase delivery ratio? Which one is the best node for forwarding?

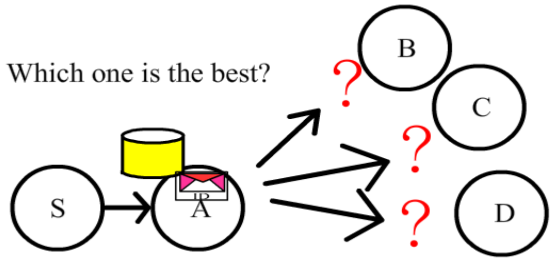


Figure 6. A challenge problem in Step 2 (OC-process).

In this paper, we focus on how to choose the better next node. We want to know the different characteristics of the routing in the different scenarios. Therefore, we build two scenarios, the first one is Random Waypoint Mobility and another is Community Mobility. Further, we add the metric of the delivery predictability of ProPHET-Routing to choose the better node to replication. We can then integrate the metric with OC-Process in OOP-Routing.

We have maintained the metric of delivery predictability in every node. However, how to decide the threshold of this metric can optimally improved the throughput?

We found that, not only the positive threshold but also the negative threshold can help packets to be sent out. We found when the value of threshold was -100%, and then we could get a degenerate routing into OOP-Routing. In the opposite side, the result is equivalent as the OOPProPHET-Routing and set the value of threshold is -100%.

If we select the node which value of threshold is above zero, indicating that, we just select the higher predictability opportunity node to replicate and give up the others node. Also, it represents that we will only use less quantity of nodes to assist routing. We try to verify the characteristics of this routing. Therefore, we focus on the value of threshold and have made a series of experiments.

IV. Simulation and Performance Evaluation

We used NS2 to simulate and to evaluate the performance of our OOPProPHET-Routing scheme.

A. Mobility Model

1) RWP Model:

The first scenario is similar to the Epidemic-Routing in [3]. And, we use the tools in NS2 to generate RWP (Random Waypoint [9]) mobility scenario.

The command is "./ns-2.28/indep-utils/cmu-scen-gen/setdest/setdest -v 2 -n 50 -s 1 -m 1 -M 20 -t 2020 -P 1 -p 3 -x 3000 -y 600 > scen".

We use "nam" tools to show the scenario graphs in Fig. 7.

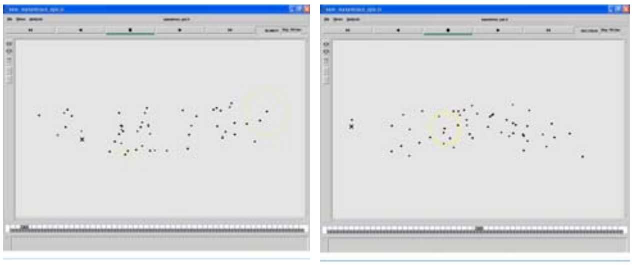


Figure 7. Scenario for RWP Model (The Epidemic-Routing is Running)

2) Community Model:

The second scenario is a community model, similar to ProPHET-Routing in [1]. We set the number of gathering places are two and random select half nodes to go to the gathering place on left side, the remaining half like the gathering place on the other side. As in Fig. 8, there are ten communities and two gathering places in 3000m X 600m area.

C8	C9	C10	Gather 2
C4	C5	C6	C7
Gather 1	C1	C2	C3

Figure 8. Communities (C1-C10) and two "gathering place"

Each node has one home community which it is more likely to visit than other places, and for each community there are five nodes as their home community. The mobility in this scenario indicates that nodes select a destination and a speed between 10 and 30 m/s, move there, pause there for 3 seconds. Furthermore, Table I shows the probabilities of different destination being chosen depending on the current location of a node. Table II shows the parameter settings of ProPHET. We can see the scenario graphs in Fig. 9.

TABLE I.
DESTINATION SELECTION PROBABILITIES

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

TABLE II.
PROPHET PARAMETER SETTINGS

Parameter	Pinit	β	γ
Value	0.75	0.25	0.98

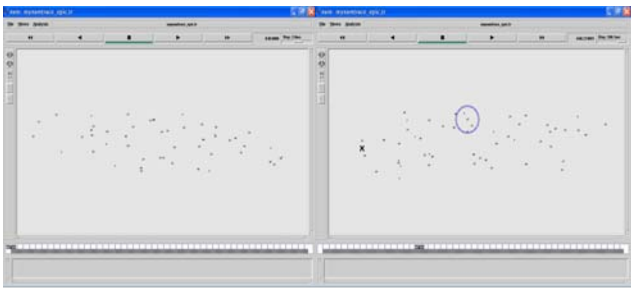


Figure 9. Scenario for Community Model (The Epidemic-Routing is Running)

B. Simulation Setup

In both mobility models, we randomly select one of the 45 nodes, and then combination two points as a flow pair, the total is 1980 pairs. Each pair during the simulation time sends a packet flow; the total is 1980 packets flows. These packets will start the simulation flow at 40 seconds after simulation starts, the last sent flow in 1620 seconds. Our experiment time is 2020 seconds, referring to Table III, Table IV.

TABLE III.
SIMULATION MODEL PARAMETERS

Number of mobile nodes	50
Topology area	3000m X 600m
Transmission Range	10m~80m (default Dist=40m)
Node speed	0.1~20m/sec Uniform Distr.
Message length	1KB
Hop limit	3, 6, 11,15,20,30
Simulation time	2020sec
The size of Buffer-Queue	10,15,20,30,40,50,99,198 ,297,396,495,594,990,1980,2200
ThV (Threshold)	-100,-10,-5, 2, 5, 10, 100 (%)

TABLE IV.
SIMULATION NETWORK PARAMETER

Communication Channel type	WirelessChannel
Radio Propagation Model	Two Ray Ground
Network Interface type	WirelessPHY
MAC type	802.11
Interface Queue type	DropTail/PriQueue
Antenna model	OmniAntenna
Antenna Height	1.5m
max packet in ifq (ifq)	200

Our scheme is compared with Epidemic-Routing (abbreviated as Epi) and Direct-Transmission-Routing (abbreviated as Dir). We abbreviate the value of the threshold with "ThV" and use this abbreviation to represent our OOPProPHET-Routing in the later of this paper. We catch the experimental result in the IFQ is 200, the transmission distance is 40m. The size of buffer-queue of message is varying from 0.5%~111%. We use some metrics, number of received messages (abbreviated as Recv), delay time, number of forwarding packets. We study these metrics and figure out the characteristics of our routing.

C. Result

For each metric and scenario, there are two group graphs in Fig. 10 (a) (b) (c) (d) and Fig. 10 (a) (b) (c) (d). Each of these graphs contains curves for nine routings including Epidemic, Direct and OOPProPHET-Routing with the value of ThV which varies from -100% to +100%.

From Fig10 (a), Fig11 (a), we found that the Epidemic-Routing is the Upper-Bound, and Direct-Routing is Low-Bound. Furthermore, Fig. 11 (a), in the Community Model, we can find that: when the ThV is increasing from -100% to +100%, the numbers of received packets are gradually increasing.

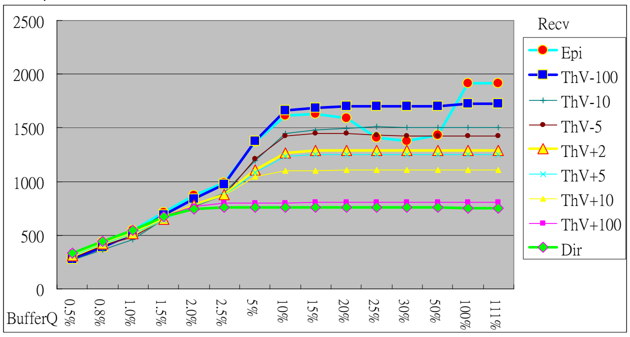
We split three parts to focus on Epidemic-Routing: the first part, when the size of buffer-queue is equal or larger than 100%, The Epidemic is the best. The second, when the size of buffer-queue is not large enough, such as 15%, and the hop-limit is bigger, such as 11, there are too many of the amount of forwarding packets in Epidemic-Routing; therefore, the delivery ratio is rapidly decreasing. The third, when the buffer-queue size is too small, such as less 1 %, the "anomaly" occurs in Fig. 10 (a). The numbers of received packets in Direct-Routing, which is no any forwarding-packet to be forward, has biggest delivery ratio than other routing.

In Fig. 10 (b) and Fig. 11 (b), the Y-axis is logarithmic coordinates to show the numbers of forwarding packets are rapidly increasing, and we can find the forwarding packets of OOPProPHET-Routing are less one-tenth than Epidemic-Routing.

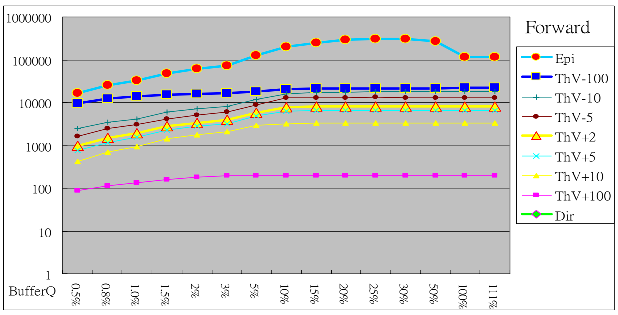
In Fig. 10 (c) and Fig. 11 (c), delay time of Direct-Routing is longest but Epidemic-Routing is shortest.

From Fig. 10 (d), in RWP Model, the mobility is random waypoint we can not guess the track of nodes in the future. Therefore, we get the result similar as OOP-Routing but the less value of ThV, the more numbers of next nodes. In Fig. 11 (d), in Community Model, we have the higher probability to guess the track. When the value of ThV is -5%, we can find the delivery ratio is close to another delivery ratio when the value of ThV is -100%. Therefore, we can select fewer nodes to achieve the similar delivery ratio.

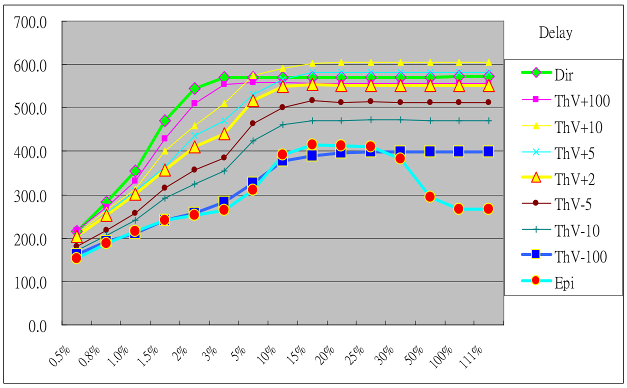
1) RWP Model



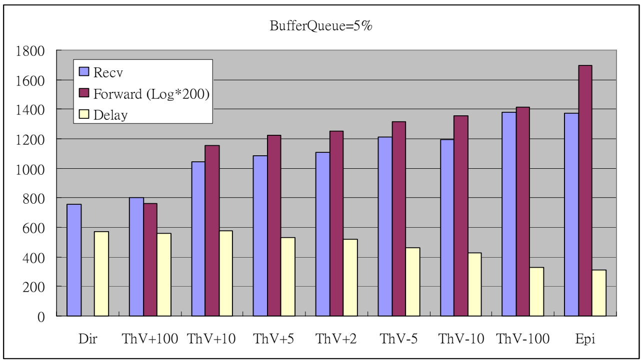
(a) The numbers of received packets.



(b) The numbers of forwarding packets.



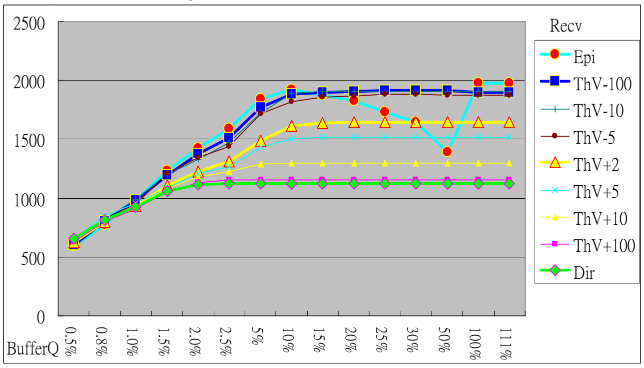
(c) The average number of delay time.



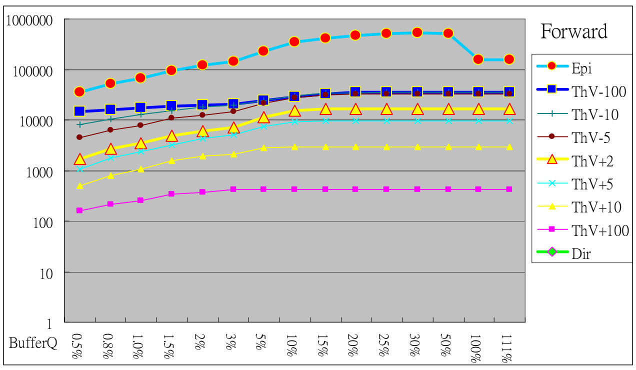
(d) Compare three metrics in buffer-queue size is 5%.

Figure 10. Simulation results in RWP Model (Dist=40m, 11Hops)

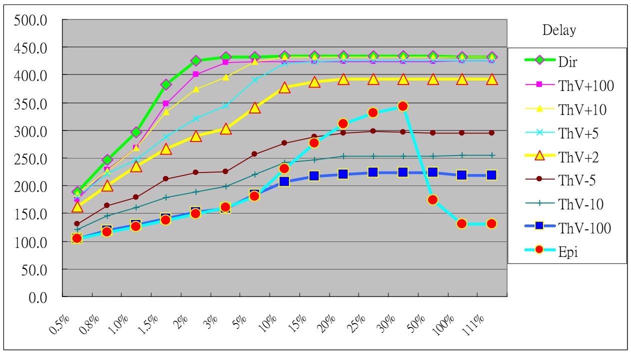
2) Community Model



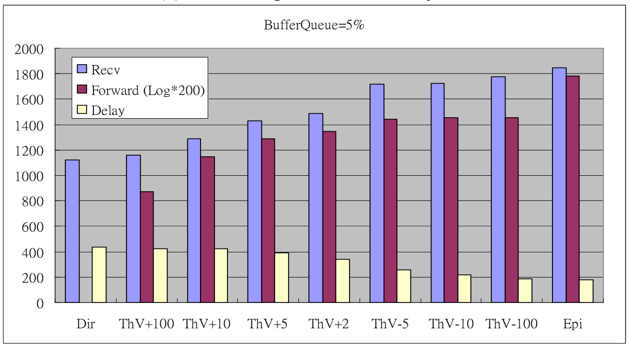
(a) The numbers of received packets.



(b) The numbers of forwarding packets.



(c) The average number of delay time.



(d) Compare three metrics in buffer-queue size is 5%.

Figure 11. Simulation results in Community Model (Dist=40m, 11Hops)

According to the results, about Epidemic-Routing, following the increasing buffer-queue size, the life time of each message packet is also extended, so the numbers of received messages, delay time is increasing, and the amount of forwarding packets alarming increase significantly. The received ratio of Epidemic is high bound and the delay time is low bound on whole routing methods.

About Direct-Routing, the received ratio is low bound and the delay is high bound on whole routing methods, and the number of forwarding packet is zero.

About OOPProPHET-Routing, when the sizes of buffer-queue are more than 10%, we can get almost the same result compared with 100%. Therefore, OOPProPHET-Routing is unaffected by the lack of the buffer-queue. In addition, we can find “ThV-100” is close to the Epidemic, “ThV+100” is close to Dir, and “ThV+2” is in the middle of the two. In Community model, we can select the value of ThV is -5% to get the better tradeoff then other value.

In comparison to the differences in two mobility models, we collect the experimental results in Fig. 12. In RWP Model, there are more nodes to assist forward the packets in small value of ThV. We select the better value of ThV is -100%. In Community Model, if we choice the value of ThV is -5%, we can get the delivery ratio is very close to ThV-100. Why do we use the negative value of ThV? The reason is about next node which is better than this node or is only a mall less than this node both can improve the performance of routing.

Therefore, about the value of ThV, we have the better choices: we can set the value of ThV is -5%. At this situation, the node which has messages want to sent choose the value of ThV of next node will be better or less -5% than itself; therefore, the numbers of next nodes will trend to be better and will not soon become too scare. We can achieve a good better tradeoff of performance in routing.

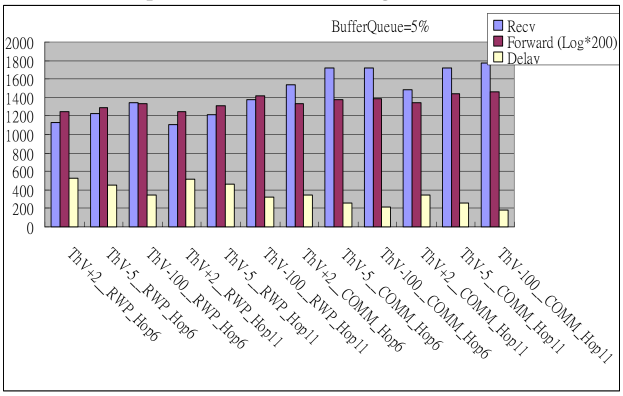


Figure 12. Compare the differences in two mobility models, when the hop-limit is 6 and 11.

Moreover, we can find that, when the value of ThV varies from +100 to -100, the received ratio is gradually increased and the amount of forwarding packets is gradually reduced. We can use ThV to adjust the delivery success ratio and simply quality control in DTNs.

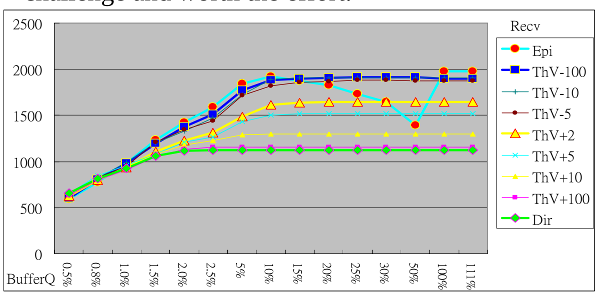
3) Compare the effects of hop limit and aging interval

At last, we also discuss the effects of hop limit and interval of aging in Community mobility Model.

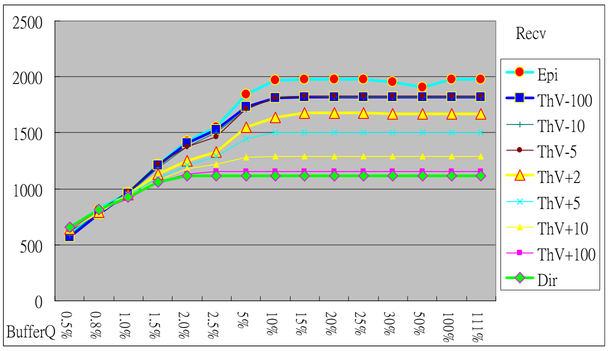
a) Hop limit in Community Model

In Fig. 13, in Community Model, we compare three different hops limit: 3, 6 and 11. We can find that: more less hops, the result of OOPProPHET-Routing is more like Direct-Routing. On the contrary, when the hop-limit is expanding, the result of OOPProPHET-Routing is more close to Epidemic-Routing.

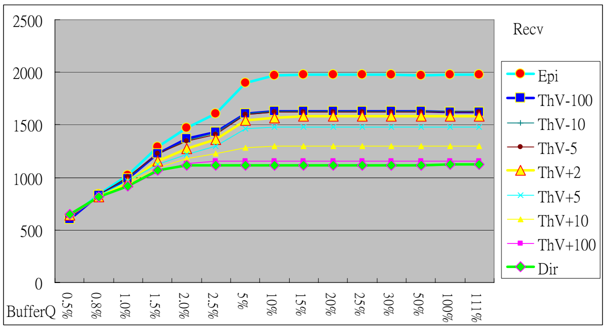
If the value of hops-limit becomes higher and the diameter of network is larger, the repeated replication packets will generate bigger effect. Especially for the Epidemic, because there are many of the forwarding packets were generated, the buffer-queue size is not enough to continually maintain packet replications. Therefore, this routing must drop many useful packets and may not have the highest received ratio. Trying to reduce unnecessary duplication packet will be a challenge and worth the effort.



(a)Hop Limit=11



(b)Hop Limit=6

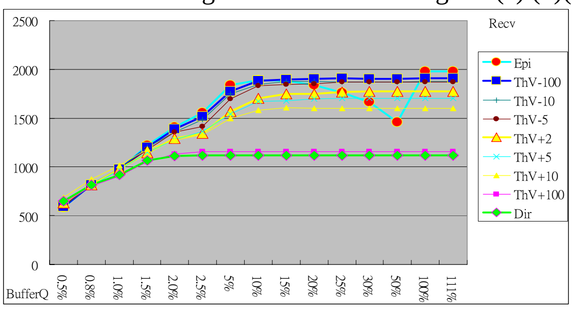


(c)Hop Limit=3

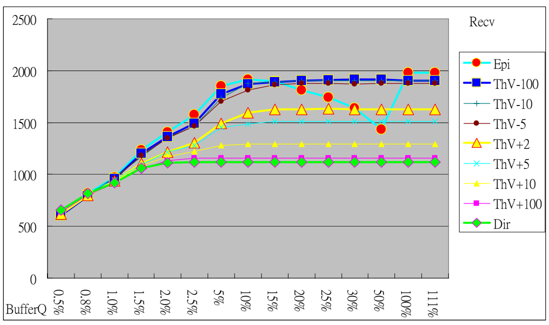
Figure 13. Compare three different hop limit: 3, 6 and 11 in Community mobility Model

b) Interval of Aging in Community Model

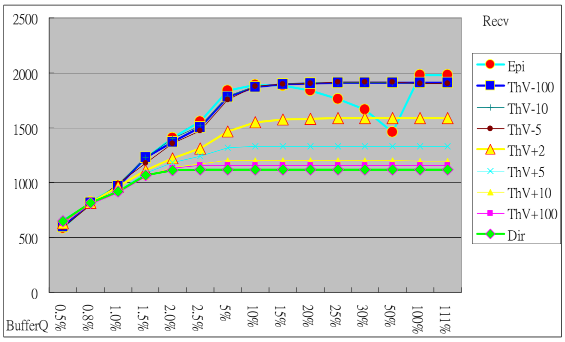
The default of Aging Interval is set to five seconds and hop limit is eleven. We adjust this parameter is one sec and ten seconds to get the results as in Fig. 14 (a) (b)(c).



(a)Aging Interval=1 sec (Hop =11))



(b)Aging Interval=5 sec (Hop =11))



(c)Aging Interval=10 sec (Hop =11)

Figure 14. Compare three different Aging Interval: 1, 5, 10 sec in Community mobility Model

Compare the three figures, we found that: except “ThV+100” curve, more less value of Aging Interval then more close to the curve of Epidemic Routing. Conversely, except “ThV-100” bigger value of Aging Interval then more close to the curve of Direct-Transmission Routing.

The influence of adjusting Aging Interval on delivered ratio is similar as the influence of adjusting ThV.

In the general case, we can select the value of Aging Interval to response the average time of the interval of encounter. Then, we can just care the value of ThV in OOPProPHET and the original Routing will be work as the routing with adjustment characteristics.

D. Discussion in OOPProPHET-Routing

We discuss the degradation of this routing method: the first case, we set the value of ThV is 100%, the routing will be similar to Direct-Routing and one broadcast. The second case, we set the value of ThV is -100%, the routing will nearly be same as OOP-Routing.

When the size of buffer-queue is less than 1%, we can find the "anomaly" phenomenon. The reason is that most of the useful messages will be dropped by the forwarding messages. If we have sufficient buffer-queue size and sufficient network bandwidth to accept a large amount of forwarding packets, the Epidemic-Routing is the best. Nonetheless, the flooding problem in Epidemic is a big problem! Therefore, a novel routing could maintain a similar amount of delivery messages and decrease the amount of forwarding packets at the same time, then the routing has a chance to better than Epidemic in real scenarios. In our experimental results, we can find the performance of this routing exceeding Epidemic. The main reasons are two: the first reason is the numbers of the forwarding packets just only less one-tenth. The second reason is the metric of ProPHET, if the metric can catch the mobility, we can select higher chance nodes to send and can enhance the utilization of the limited bandwidth. At last, we set the value of ThV is -5%, we can get the better tradeoff which between the delivery ratio and the numbers of forwarding packets in OOPProPHET-Routing. In the general case, we can select the value of Aging Interval to response the average time of the interval of encounter. Then, we can just care the value of ThV in OOPProPHET and the original Routing will be work as the routing with adjustment characteristics.

V. Conclusions & Future Work

In this paper, we proposed a new routing protocol at DTMNs, namely OOPProPHET-Routing. We integrate our OOP-Routing [5] and the probabilistic metric of ProPHET-Routing [1] to balance the tradeoff of the Routing throughput in DTNs. In RWP Model, the received ratio is close to Epidemic and the numbers of forwarding packets less than one-tenth. Furthermore, in Community Model, not only we can get the better balance than Epidemic, but also can adjust the value of ThV to enhance the utilization of the limited bandwidth. An important feature of our routing is that, we need the requirements of size of buffer-queue are far less than Epidemic Routing, but the received ratio is nearly close. Therefore, we can use this routing with other coding techniques to serve the quality control of the communication in DTNs. For a different model, we can carefully to calculate the metric of delivery predictability we can then adjust the value of the ThV to adapt different scenarios.

References

[1] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," SIGMOBILE Mobile Comput. Commun. Rev., vol. 7, no. 3, 2003.

[2] A. Lindgren, A.Doria, O. Schelen, "Probabilistic routing in intermittently connected networks," in: Proceedings of SAPIR 2004, pp. 239 – 254.

[3] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," Tech. Rep. CS-200006, Department of C.S., Duke University, Durham, NC, 2000.

[4] Anders F. Lindgren, Avri Doria, Elwyn Davies, Samo Grasic, "Probabilistic Routing Protocol for Intermittently Connected Networks," draft-irtf-dtnrg-prophet-09, [Online] Available:http://tools.ietf.org/html/draft-irtf-dtnrg-prophet-09, April 3, 2011.

[5] Jiun-Ting Jiang, Po-Zung Chen, "OOP: A Novel Routing Protocol based on One-Broadcast One-Copy Persistent-Storage in Delay Tolerant Mobile Ad Hoc Networks," NCS 2011, Workshop-2.MW, p216-227.

[6] Kevin Fall, "A delay-tolerant network architecture for challenged internets," ACM, SIGCOMM, August 2003.

[7] P. Mundur, M. Seligman, G. Lee, "Epidemic routing with immunity in Delay Tolerant Networks," Military Communications Conference, 2008.

[8] R. J. D'Souza, Johny Jose, "Routing Approaches in Delay Tolerant Networks: A survey", International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 17, 2010.

[9] T.Camp, J.Boleng, and V.Davies, "A survey of mobility models for ad hoc network research," WCMC: Research, Trends and Applications, vol.2, no.5, pp.483 – 502, 2002.

[10] T.Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: The single-copy case," IEEE Trans. Networking, vol. 16, no. 1, Feb. 2008.

[11] T.Spyropoulos, K. Psounis, C.S. Raghavendra, ETH Zurich, ZurichEfficient, "Efficient routing in intermittently connected mobile networks: the multiple-copy case,"Networking, IEEE/ACM Transactions on, Feb. 2008.

[12] Ting-Kai Huang, Chia-Keng Lee, Ling-Jyh Chen, "PRoPHET+: An Adaptive PRoPHET-Based Routing Protocol for Opportunistic Network," 24th IEEE International Conference on Advanced Information Networking and Applications 2010.

Shih-Yang Yang received his Ph.D. degree in Computer Science and Information Engineering from Tamkang University, Taiwan, in January 2008. Since January 2008, he is an Associate Professor with the Department of Media Art at Kang-Ning Junior College of Medical Care and Management (Taipei, Taiwan). His research interests include parallel & distributed systems, web technology network and animation education.

Jiun-Ting Jiang received his Master's degree in Computer Science and Information Engineering from Tamkang University, Taiwan, in 2000. He is working toward the Ph.D. degree in the Department of Computer Science and Information Engineering in Tamkang University, Taiwan. His research domains include distributed systems, Ad-Hoc wireless networks and delay-tolerant communication for mobile networks.

Po-Zung Chen received his Ph.D. degree in Computer Science from the University of Iowa in December 1989. From November 1989 to May 1990, he was a visiting Assistant Professor at Michigan Technological University (Houghton, Michigan). Since August 1990, he is an Associate Professor with the Department of Computer Science and Information Engineering at Tamkang University (Taipei, Taiwan). His research interests include object-oriented distributed programming, parallel & distributed systems and simulation & modeling.