www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

A Hybrid Routing Algorithm for Delay Tolerant Networks

Jianbo LI, Jixing XU, Lei YOU, Chenqu DAI, Jieheng WU
Information Engineering College of Qingdao University,
Ningxia Road 308, Qingdao 266071, China
Tel.: (+86)15020081468
E-mail: lijianboqdu@gmail.com

Received: 19 December 2013 /Accepted: 29 December 2013 /Published: 30 December 2013

Abstract: The unavailability of an end-to-end path poses great challenges in routing algorithms for delay tolerant networks (DTNs). Meanwhile, routing protocols for the traditional Ad-hoc or Mobile Ad hoc NETworks (MANETs) cannot work well due to the failure of assumption that network connections are available in most of time. In this article, we put forward a hybrid routing algorithm to combine the classical Spray&Wait and PRoPHETalgorithms. The proposed protocol essentially bases on an unsymmetrical Spray&Wait at the first stage, and then adaptively allocates the message copy count on demand by using the delivery probability used in PRoPHETalgorithm for the purpose of making forwarding decision more sensible. Finally, each node takes the real time inter-contact interval and average inter-contact interval with the message destination into consideration in order to dynamically adjust its message copy count, thus making timely and rational routing decision based on the real time network condition. Extensive simulations have been conducted to verify the effectiveness and efficiency of our algorithm and the results demonstrate that our algorithm achieves a higher delivery probability and lower average message latency under Random Walk and Random Waypoint mobility models, as compared to Spray&Wait, PRoPHET, Epidemic and FirstContact. *Copyright © 2013 IFSA.*

Keywords: Delay tolerant networks, SPRAY&Wait, PRoPHET, Hybrid routing, Adaptive routing.

1. Introduction

In recent years, delay tolerant network as a new emerging network has received extensive attentions. It is originally applied in military battlefield networks and emergency rescue networks. Now the concept of DTN has been applied in various fields, such as internet access service networks [1], Vehicular Ad Hoc Networks [2-4], Habitat Monitoring Networks [5], and underwater sensor networks [6], etc. These special application networks deployed in challenging environments are characterized by intermittent connectivity [7-9], frequent partitions, high node mobility and low node density, etc. So there may never be a complete end to end path between the sender and the receiver. In this case, the existing routing protocols based traditional TCP/IP are difficult to get satisfying routing performance. Consequently, the successful message transmissions in such networks face great challenges. Now, the routing design for delay tolerant networks has become the focus of wireless network research fields.

Delay Tolerant Network first originated from Interplanetary Networks (IPN [10]), which was used to solve the communication problems between planets. In 2003, Kevin Fall proposed the concept of DTN. Soon afterwards, the DTN Research Group (DTNRG) put forward the DTN network architecture, which introduced a bundle layer to cope with heterogeneous networks. And with the help of the bundle layer, DTN routing adopts the store-carry and forward strategy to relay messages hop by hop for

Article number P_RP_0036

the, thus it can be able to cope with frequent network topology partitions to some extent. But in most DTN application scenarios, it may be very difficult to capture the global network topology knowledge. In this case, DTN routing is still difficult to make the right routing selection for the purpose of getting a satisfying routing performance.

A simple and direct approach is to increase the number of message copies, which can increase the chance of encountering the final destination node, but also inevitably consume a large number of network resource. But in most DTN application scenarios, the buffer resource is extremely limited. So the desirable way is to get a good balance between the higher delivery rate and less consumption of resources.

In this paper, we first propose to limit the number of message copies for the purpose of getting a balance between higher message delivery ratio and less resource consumption. And then we take full use of the delivery probability between two nodes to distribute message copies reasonably. Besides, we also propose to dynamically adjust the number of message copies based on the real time network condition. Finally, a hybrid routing algorithm for delay tolerant networks is proposed to improve routing performance.

The rest of this paper is organized as follows. We discuss some related works in Section 2. In section 3, we give a detailed description of the proposed algorithm. The performance evaluations and comparisons among the proposed algorithm, First Contact, Spray&Wait, Epidemic and PRoPHET are presented in Section 4. Finally, section 5 summarizes this paper and gives future research directions.

2. Related Work

After the DTN architecture was presented, a lot of researches have proposed many typical DTN routing algorithms.

The early DTN routing algorithms are based on single copy strategy. For example, Direct Delivery [11] and First Contact [12].They only deliver one single copy of the message in the entire network. The big advantages of this kind of routing algorithms are that they only need to consume little resource and can lower the network overhead ratio. But due to the frequent network topology partitions, they are difficult to get satisfying message delivery ratios.

In order to improve message delivery ratio, many multi-copy based routing algorithms have been proposed. Epidemic [13] is the typical algorithm, which tries to replicate all carried message to all encountered nodes without evaluating these nodes. It can lower the message delivery latency to some extent, but it also leads to a high network overhead ratio. Besides, it will create a lot of redundant message copies, so it is still difficult to get a satisfying routing performance when the buffer resource is insufficient. Based on this condition, Spraty&Wait [14] and Spray&Focus [15] make improvements by limiting the number of message copies. Only in spray period can they replicate L message copies to L different intermediate relay nodes. In wait period or focus period, they do not replicate message copies any more. By limiting the number of message copies, they can control the network overhead ratio and reduce the consumption of network resource. But they also do not evaluate the encountered nodes when distributing the message copies. So their message distribution strategies are not sensible enough.

PROPHET [16] a typical routing algorithm based on history utility, which takes full use of the regular mobility patterns of nodes and uses the captured encounter history information and transitive property to predict and update the delivery probability between two nodes. And then it takes into account the delivery probability when selecting next hop relay nodes. By evaluating every encountered node, it only selects the neighbor with a bigger delivery probability as the next hop relay node. So in this case, its routing selection is more accurate than that of Epidemic, thus improving the message delivery ratio.

In [17], several knowledge oracle based routing protocols are proposed. By abstracting a weighted graph from the captured global network topology information, they apply different modified Dijkstra algorithms on the graph to seek the shortest path. However the potential dependence on the pre-known knowledge oracle highly constrained the practicality of these protocols.

[18-19] propose some typical routing protocols based fixed infrastructures. And [20-22] put forward to some mobile infrastructures assistant routing schemes. These routing schemes can improve network routing performance by introducing the extra infrastructures. Besides, [23-25] present some coding based routing schemes, which can compensate the degraded routing performance to some extent.

Most DTN routing protocols more or less rely on extra assistant information or some assumption of the network topology so as to enhance the routing performance, while sacrificing the practicality and simplicity.

3. Routing Framework

Inmost DTN application scenarios, the buffer resource is extremely limited. So in this case, controlling the number of message copies is a desirable scheme to improve routing performance. But the classical Spray&Waitis not sensible enough when distributing message copies. So in this article, by taking full use of the delivery probability used in PRoPHET, we reasonably allocate message copies and finally put forward a hybrid routing algorithm to implement unsymmetrical spray routing. Besides, in order to make timely and rational routing decision based on the real time network condition, we also dynamically adjust message copy count on relay nodes by taking the real time inter-contact interval and average inter-contact interval with the message destination into consideration.

3.1. Delivery Predictability Calculation

In order to distribute message copies more sensible, we propose to allocate message copies according to delivery predictability. Here, we use the three proposed equations in PROPHET [16] to predict and update the probability of encountering a certain node.

When two nodes a and b encounter, the equation (1) is used to predict the delivery predictability that node a has for node b, where $P_{init} \in [0,1]$ is an initialization constant.

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

When two nodes do not encounter each other for a period of time, the delivery predictability should be reduced in the process. The equation(2) is the aging function, where γ^k is the aging constant, and k is the number of time units that have elapsed since the last time the predictability was aged.

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

The delivery predictability between two nodes also has the transitive property. So if node a encounters node b very frequently, and node b also encounters node c very frequently, then node a may also be a good forwarder to relay messages destined for node c.

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

3.2. Message Copies Distribution

In most DTN application scenarios, nodes do not move around completely randomly. On the contrary, the two nodes which encountered each other very frequently in the past are more likely to encounter each other again in the near future. So in this case, we take into account the delivery probability when distributing message copies. The intermediate relay node with a bigger delivery probability is more likely to finish message transmission successfully, so the node should be distributed more message copies. When two nodes encounter, we re-allocate the number of message copies they should carry according to the total number of message copies they carry now. The detailed process are shown in equations (4) and (5), where m_d is the message for node d, $L_{a_{old}}(m_d)$ is the number of message copies that node a is carrying, and $L_{a_{new}}(m_d)$ is the number of message copies that we re-allocate to node a.

$$L_{a_{new}}(m_d) = \frac{P_{(a,d)}}{P_{(a,d)} + P_{(b,d)}} \times (L_{a_{old}}(m_d) + L_{b_{old}}(m_d)) \quad (4)$$

$$L_{b_{new}}(m_d) = (L_{a_{old}}(m_d) + L_{b_{old}}(m_d) - L_{a_{new}}(m_d)) \quad (5)$$

3.3. Inter-Contact Interval

In order to timely and rationally adjust the number of message copies based on the real time network condition, every node needs to record the two time interval $\tau_{(a,b)}$ and $m_{(a,b)}$, where $\tau_{(a,b)}$ represents the time that have elapsed since the last time they encountered, and $m_{(a,b)}$ denotes the average encounter time interval which is updated according to the equation (6). Their detailed update processes are shown in Fig. 1.

Algorithm 1. Update τ and *m* value

Require:
When connection between *a* and *b* is up
Ensure:
1. = currentTime - lastUpdateTime;
2. lastUpdateTime currentTime;
3. $m_{(a,b)_{new}} \leftarrow am_{(a,b)_{old}} + (1-\alpha)$

Fig. 1. Update τ and *m* value.

$$m_{(a,b)_{new}} = \alpha m_{(a,b)_{old}} + (1-\alpha)\tau_{(a,b)} \quad (6)$$

Finally after updating the two encounter time interval, we can dynamically adjust the number of message copies according to the following equation (7).

$$L_{a_{new}}(m_d) = L_{a_{old}}(m_d) \times \frac{\tau_{(a,d)}}{m_{(a,d)}} \quad (7)$$

3.4. Detailed Routing Algorithm

Based on the above schemes, we can finally implement the proposed routing algorithm as shown in Fig. 2. Here, we define the message set of node *a* as the follows.

$$M(a) = \{m_i \mid 1 \le i \le |M(a)|\} \quad (8)$$

As shown in algorithm 2, when two nodes encounter, line 1 first exchanges some essential information which are used to update the numbers of message copies on the two nodes. Lines 4-5 recalculate the numbers of message copies that the two nodes should respectively carry according to the total number of message copies they are carrying and the delivery probabilities for destination node. Then line 6 adjusts the number of message copies based on the real time network condition. Lines 7-10 update the number of message copies in the buffer according to the calculated $L_{a_{new}}(m_k)$. Lines 11-16 forward message copies to the encountered node.

Algorithm 2. The proposed routing protocol
Require:
When node *a* encounters node *b*
Ensure:
1. exchange *P*, *m* and τ with each other;
2. **for** message $m_k \in N$ **do**
3. $d \leftarrow$ destination node of m_k;
4. compute $L_{a_{new}}(m_k)$ with equation (4);
5. compute $L_{b_{new}}(m_k)$ with equation (5);
6. update $L_{a_{new}}(m_k)$ with equation (7);
7. buffer.get(m_k).update($L_{a_{new}}(m_k)$);
8. **if** $L_{a_{new}}(m_k) = 0$ **then**
9. buffer.deleteMessage(m_k);
10. **end if**
11. **if** $L_{b_{new}}(m_k)$ && $L_{b_{old}}(m_k)$ **then**
12. add m_k into forwardList;
13. forwardList.get(m_k).update($L_{b_{new}}(m_k)$);
14. forwardList.sort(ascending, TTL);
15. forwardList.sendTo(*b*);
16. **end if**
17. **end for**

Fig. 2. The proposed routing protocol.

4. Simulation

In this section, we use the most popular network simulator ONE (the Opportunistic Network Environment evaluator) to implement our proposed routing protocol. And extensive simulations have been conducted respectively based on Random Walk mobility model and Random Waypoint mobility model. We compare the routing performance of our proposed algorithm, First Contact, Spray and Wait, Epidemic and PROPHET in terms of message delivery ratio, network overhead ratio, average delivery latency and average hop count. We mostly focus on their different routing performance indifferent buffer sizes and message time-to-live. The detailed simulation settings are shown in Table 1.

Table 1. Simulation settings.

Parameter	Default value
Area size	1000 m x 1000 m
Number of nodes	120
Tickets	18
Initial topology	Uniform
Transmit radius	100 m
Message size	500 K
Message interval	40 s
Transmit speed	250 Kbps
Moving speed	0.5 (m/s)
Node buffer size	5 M
Time-To-Live(TTL)	300 min
Simulation time	5 hours
Movement model	RandomWalk and RandomWaypoint

4.1. Vary Buffer Size in Random Walk Model

Fig. 3 shows the different simulation results of varying node's buffer size in the Random Walk model. As shown in the figure, the proposed algorithm can get obvious advantages in terms of delivery probability and average latency, which can verify that the proposed routing scheme can greatly improve the transmission performance of the entire network.

First Contact only delivers one single message copy in the entire network, thus it needs more time to finish message transmission. As shown in the figure, the average latency of First Contact is the biggest among the five routing algorithms, and its average hop count is also the most compared to others. But its network overhead ratio is significantly lower than those of Epidemic and Prophet. So in a conclusion, First Contact is applicable to the application scenario whose load capacity is low and buffer resource is extremely limited.

Epidemic does not take any measures to control message redundancy, so it will inevitably create numerous redundant messages in whole network and consume more network resource. When node's buffer resource is insufficient, Epidemic will drop a large number of message packets, so its message delivery probability is the lowest in this case. Besides, these redundant messages also increase the network overhead ratio. However, Epidemic can reduce the average latency and average hop count compared to First Contact.

Spray and Wait limits the number of message copies, so its overhead ratio can be kept at a very low level. And its average hop count is also the lowest because that it does not forward message in the wait period. But it does not evaluate nodes when distributing message copies, so its message delivery probability is lower than that of our proposed algorithm and its average latency is also bigger than that of the proposed algorithm.

Prophet uses the delivery probability to evaluate every encountered node, so it can great improve the message delivery probability and control overhead ratio compared to Epidemic. Besides, it can also improve the accuracy of routing selection by using the deliver probability as the metric, thus reducing the cost of message transmission.

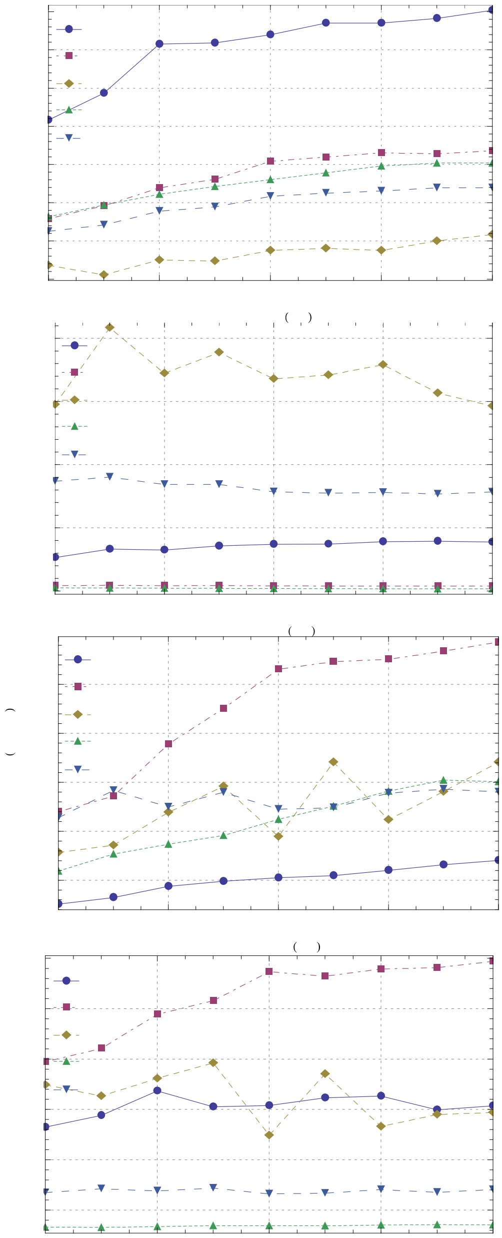

Fig. 3. Deliveryprobability, overheadratio, averagelatency, average hop count vs. buffer size in Random Walk model.

Our proposed routing algorithm dynamically distributes message copies according the delivery probability, which is more sensible and can greatly improve message delivery probability. So as shown in the figure, it can get the highest message delivery probability and the lowest average latency. Besides, the overhead ratio of our proposed algorithm is much lower than those of Epidemic and Prophet.

Finally from the whole figure, we can make a conclusion that our proposed routing algorithm can outperform the other four algorithms in terms of message delivery probability and delivery latency in the Random Walk model based network, thus providing a higher message delivery probability.

4.2. Vary Message time-to-live in Random Walk Model

Fig. 4 describes the different simulation results of varying message's TTL in the Random Walk model. As shown in figure, our proposed routing algorithm can still get the highest message delivery probability and the lowest average delivery latency among the five algorithms, which proves once again that our proposed algorithm can greatly improve network routing performance.

When message TTL is more than 250, the message delivery probabilities of the five algorithms can remain stable, but the delivery probability of our proposed algorithm is obviously higher than that of the other four algorithms. Besides, Epidemic still gets the lowest message delivery probability in this case.

When increasing different message TTL constantly, the overhead ratio of our proposed algorithm can be kept a low level compared to Epidemic and Prophet.

Finally from the whole figure, we can see that our proposed routing algorithm can outperform the other four algorithms in different message TTL in Random Walk model based network, thus providing a good routing performance.

4.3. Vary Buffer Size in Random Waypoint Model

Fig. 9 describes the different simulation results of varying buffer size in the Random Waypoint model. As shown in the figure, our proposed routing algorithm can still outperform the other four algorithms in terms of message delivery probability and average delivery latency, and Epidemic is still unacceptable

In a Random Waypoint model based network, node mobility can help to improve message delivery probability. So in this case, the message delivery probabilities of our proposed algorithm, First Contact and Spray and Wait are much higher than that of Epidemic and Prophet. First Contact still gets the most average hop count and the biggest average delivery latency, but also it gets a very low network overhead ratio. In addition, when node mobility is enhanced, our proposed routing algorithm can still get the highest message delivery probability and the lowest delivery latency, which can prove that our proposed routing algorithm can also improve routing performance in the Random Waypoint model based network.

Finally from the whole Fig. 5, we can see that our proposed routing algorithm can efficiently improve routing performance, thus providing a higher message delivery probability.

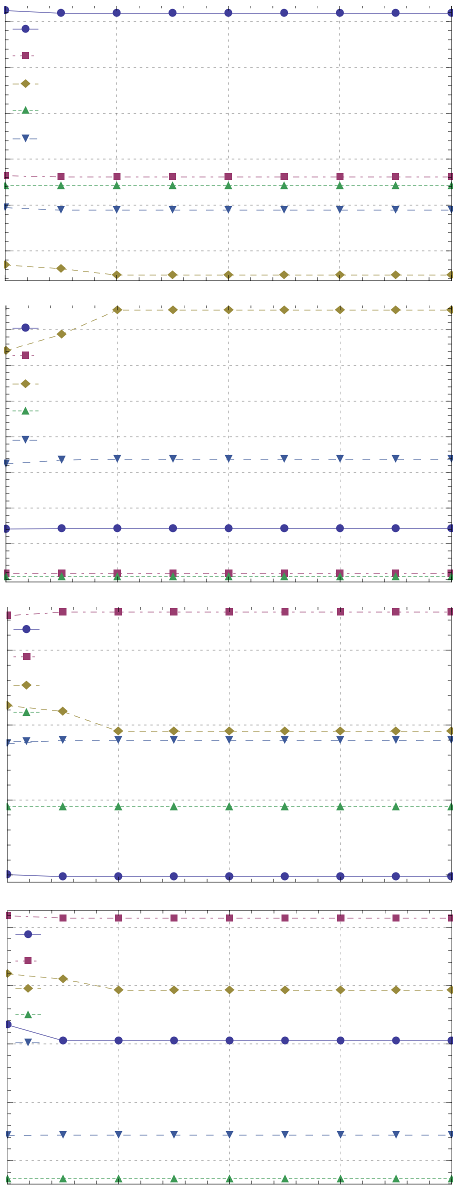

Fig. 4. Delivery probability, overhead ratio, average latency, average hop count vs. TTL in Random Walk model.

4.4. Vary Message time-to-live in Random Waypoint Model

Fig. 6 shows the different simulation results of varying message TTL in the Random Waypoint model. As shown in the figure, our proposed algorithm still get the highest message delivery probability and the lowest average delivery latency, and Epidemic is still unacceptable in terms of message delivery probability and network overhead ratio.

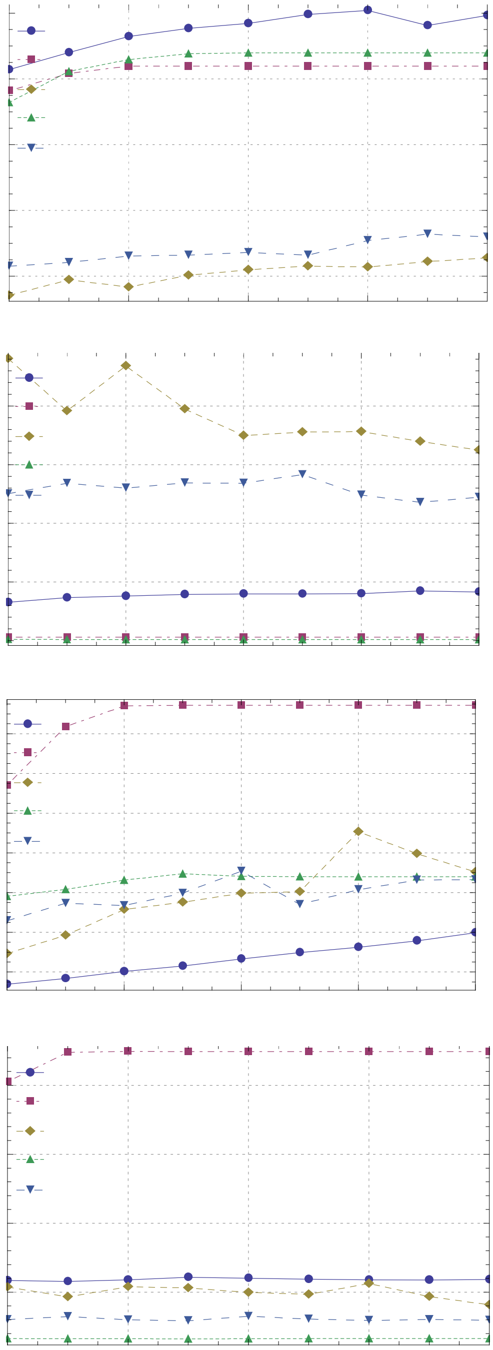

Fig. 5. Deliveryprobability, overheadratio, averagelatency, average hop count vs. buffer size in Random Waypoint model.

When varying message TTL, the message delivery probabilities of our proposed algorithm, First Contact and Spray and Wait are still bigger than that of Epidemic and Prophet, and their network overhead ratios are also lower than that of Epidemic and Prophet. This can indicate that node mobility can improve routing performance. So in this case, we recommend choosing our proposed routing algorithm, First Contact or Spray and Wait.

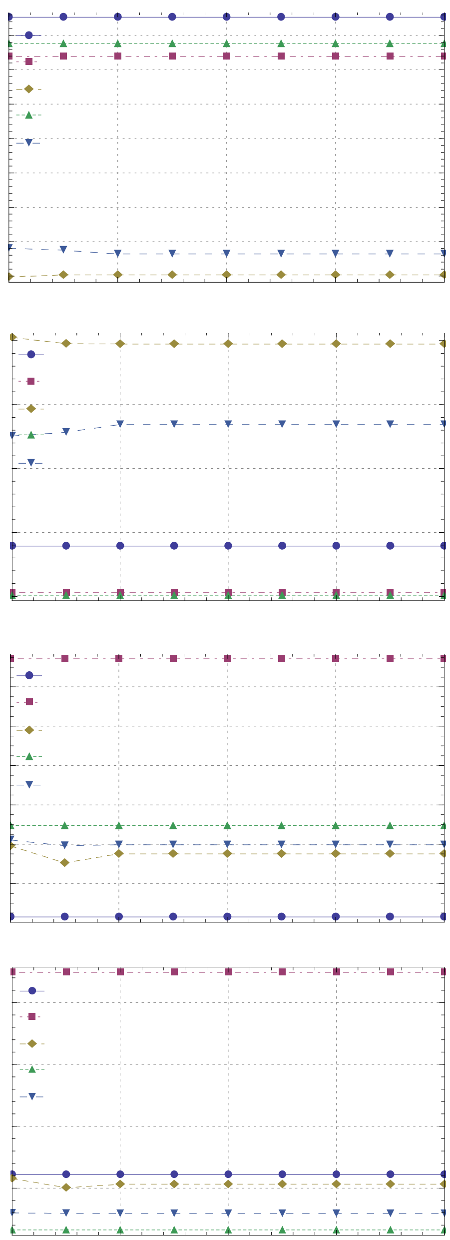

Fig. 6. Delivery probability, overhead ratio, average latency, average hop count vs. message TTL in Random Waypoint model.

5. Conclusion and Future Work

In most DTNs, the buffer resource is extremely limited. So in this case, uncontrolled flooding schemes are unacceptable. On the contrary, controlling the number of message copies tends to be more acceptable. And it is more sensible that we distribute these message copies according to the delivery probability, thus implementing efficient unsymmetrical spray routing. Besides, based on the real time network condition, we can dynamically adjust the number of message copies to get a better routing performance.

Extensive simulations have been conducted in both Random Walk model based network and Random Waypoint model based network, and the results demonstrate that our proposed routing algorithm can outperform First Contact, Spary and Wait, Epidemic and Prophet in terms of message delivery probability and average delivery latency, thus providing a better routing performance.

Our future work will focus on the social networks. In recent years, DTN model is increasingly used in civilian areas, and the social networks are closely related with the public life. So DTN model in such network tends to have broad application prospects. Our future work will focus on this, and we will also try to find the opportunity to deployment our proposed routing algorithm in this paper.

Acknowledgements

This research is supported in part by Foundation research project of Qingdao Science and Technology Plan under Grant No. 12-1-4-2-(14)-jch and Natural Science Foundation of Shandong Province under Grant No. ZR2013FQ022.

Reference

[1]. M. Demmer, K. Fall, DTLSR: delay tolerant routing for developing regions, in *Proceedings of the Workshop on Networked Systems for Developing Regions*, 2007, Article No. 5.

[2]. H. Wu, R. Fujimoto, R. Guensler, M. Hunter, MDDV: Mobilitycentric data dissemination algorithm for vehicular networks, in *Proceedings of the ACM SIGCOMM Workshop on Vehicular Ad Hoc Networks (VANET)*, 2004.

[3]. P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, C. Cervello-Pastor, From delay-tolerant networks to vehicular delay-tolerant networks, *IEEE Communications Survive Tutorials,* Vol. 14, No. 4, 2012, pp. 1166–1182.

[4]. V. N. G. J. Soares, J. J. P. C. Rodrigues, F. Farahmand, GeoSpray: A geographic routing protocol for vehicular delay-tolerant networks, *Information Fusion*, November 2011, pp. 1–12.

[5]. Alberto Cerpa, Jeremy Elson, Deborah Estrin, Lewis Girod, Michael Hamilton, Jerry Zhao, Habitat monitoring: application driver for wireless communications technology, in *Proceedings of the ACM SIGCOMM Workshop on Data Communications*, April 2001.

[6]. Z. Guo, B. Wang, J.-H. Cui, Generic prediction assisted single-copy routing in underwater delay tolerant sensor networks, *Ad Hoc Networks*, Vol. 11, Issue 3, January 2013, pp. 1–14.

[7]. K. Fall, A delay-tolerant network architecture for challenged internets, in *Proceedings of the International Conference SIGOMM'03*, August 2003.

[8]. M. Chuah, L. Cheng, B. Davison, Enhanced disruption, fault tolerant network architecture for

bundle delivery, in *Proceedings of the IEEE International Conference Globlecom*, 2005.

[9]. S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, Delay-tolerant networking: An approach to interplanetary Internet, *IEEE Communications Magazine*, Vol. 41, Issue 6, 2003, pp. 128-136.

[10]. M. Grossglauser, D. Tse, Mobility increases the capacity of ad hoc wireless networks, *IEEE/ACM Transactions on Networks*, Vol. 10, No. 4, 2002, pp. 477-486.

[11]. J. Legnay, T. Friedman, V. Conan. DTN muting in a mobility pattern space, in *Proceedings of the ACM SIGCOMM Workshop on Delay Tolerant Networking*, *ACM Press*, New York, 2005, pp. 276.283.

[12]. A. Vahdat, D. Becker, Epidemic routing for partially-connected adhoc networks, Technical Report Cs-2000-06, *Duke University*, 2000.

[13]. T. Spyropkoulos, K. Psounis, C. Raghavendra, Efficient routingin intermittently connected mobile networks: The multiple-copy case, *IEEE/ACM Transactions of Networks*, Vol. 16, No. 1, 2008, pp. 77–90.

[14]. T. Spyropoulos, K. Psounis, C. S. Raghavendra, Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility, in *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom'07*, 2007, pp. 79–85.

[15]. A. Lindgren, A. Doria, O. Schelen, Probabilistic routing in intermittently connected networks, *ACM SIGMOBILE Mobile Computers Communications Review*, Vol. 7, 2003, pp. 19–20.

[16]. S. Jain, K. Fall, R. Patra, Routing in a delay tolerant networks, in *Proceedings of the ACM SIGCOMM 2004*, October 2004, pp. 145-158.

[17]. W. Zhao, Y. Chen, M. Ammar, M. Corner, B. Levine, E. Zegura, Capacity enhancement using Throwboxes in DTNs, in *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS'2006)*, IEEE Communications Society, Vancouver, 2006, pp. 31-40.

[18]. N. Banerjee, An energy-efficient architecture for DTN Throwboxes, in *Proceedings of the INFOCOM'07,* IEEE Communications Society, Anchorage, 2007, pp. 776-784.

[19]. W. Zhao, M. Ammar, E. Zegura, A message ferrying approach for data delivery in sparse mobile ad hoc networks, in *Proceedings of the International Conference MOBIHOC'04*, 2004, pp. 187-198.

[20]. B. Tariq, M. Ammar, E. Zegura, Message ferry route design for sparse ad hoc networks with mobile nodes, in *Proceedings of the International Conference MOBIHOC'06*, 2006, pp. 37-48.

[21]. W. Zhao, M. Ammar, E. Zegura, Controlling the mobility of multiple data transports ferries in a delay-tolerant network, in *Proceedings of the International Conference INFOCOM'05*, 2005, pp. 1407-1418.

[22]. Y. Wang, S. Jain, M. Martonosi, K. Fall, Erasur-coding based routing for opportunistic networks, in *edings of the International Conference SIGCOMM'05*, 2005, pp. 229-236.

[23]. Y. Liao, K. Tan, Z. S. Zhang, L. X. Gao, Estimation based erasure-coding routing in delay tolerant networks, in *Proceedings of the International Wireless Communications and Mobile Computing Conference*, IEEE Communication Society, Vancouver, 2006, pp. 557-562.

[24]. J. Widmer, J. Y. Le Boudec, Network coding for efficient communication in extreme networks, in *Proceedings of the International Conference ACM SIGCOMM'05*, Philadelphia, 2005, pp. 284-291.

[25]. A. Keränen, J. Ott, T. Kärkkäinen, The ONE simulator for DTN protocol evaluation, in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, 2009, Article No. 55.