

# Message Deletion and Mobility Patterns for Efficient Message Delivery in DTNs

(Work in Progress)

Sathita Kaveevivitchai  
The University of Tokyo  
wan@hongo.wide.ad.jp

Hideya Ochiai  
The University of Tokyo  
jo21xq@hongo.wide.ad.jp

Hiroshi Esaki  
The University of Tokyo  
hiroshi@wide.ad.jp

**Abstract**—This paper discusses the efficient message deletion mechanism, which will be able to apply regardless of applied routing protocol, in DTNs (Delay Tolerant Networks). In order to achieve an acceptable quality of message delivery probability, we will need to allow the multiple message replicas in the system. In this context, the capability and capacity of store-carry-and-forward at the nodes is crucial. Since the mobile nodes' storage capacity is not infinite, but is rather limited, an efficient resource management mechanism, i.e., message management algorithm, has to be applied. In this paper, we propose an approach, where we design the message deletion algorithm, while take into account the heuristic knowledge of actual human mobility. Through the preliminary numerical evaluation in this paper, we show that the proposed approach will achieve an effective network resource utilization and good/acceptable message delivery probability, while reducing the system overhead and the message delivery latency. We evaluate the system performance related to several key parameters. We found and showed that the introduction of message deletion mechanism into Epidemic, PRoPHET, and Spray and Wait routing algorithm could achieve better performance.

*Keywords; DTN, deletion mechanism, human mobility;*

## I. INTRODUCTION

The rapid growth of mobile personal devices has enabled users to communicate via infrastructure networks (cellular, etc.). These mobile devices referred to as nodes, generally have limited power and storage; therefore nodes often become on and off which consequently results in intermittent connectivity and the absent of end-to-end connectivity. In addition, their mobility also causes frequent topology changes, disruption, and network partitioned. These characteristics impose challenges on the traditional TCP/IP-based communication. The researches on Delay/Disruption Tolerant Networks (DTNs) [1] have emerged to enable communication in this new working environment.

In DTNs, communication can be achieved through store-carry-and forward mechanism [2]. There are two types of routing schemes that have been proposed for such network. The first one is the forwarding-based routing scheme, where only one copy of a message exists in the network. The second one is the replication-based routing scheme, where many copies of the message disperse throughout the network. In general, the replication-based routing scheme

could provide higher delivery rate, but at the same time it consumes larger network resources, storage capacities and bandwidth.

Due to the limited storage on mobile nodes, these message replicas in replication-based routing schemes may overwhelm nodes storage capacity and will finally affect the overall performances in DTN. It is therefore necessary to add an appropriate message management algorithm in order to optimize the performances of DTN routing schemes. There are many works regarding the buffer management algorithm in DTNs. To our point of view, we identify the following two discussion points: (i) The performance of DTN system tends to be strongly affected by the human mobility pattern and their characteristics; (ii) A message management policy to prevent/reduce the buffer overflow at the nodes is essential due to the replication of messages.

Since these mobile devices are normally carried by human (phone, GPS, etc.), the opportunities of transfer will strongly depend on the human mobility characteristics. Thus, we aim to make use of this human mobility knowledge in designing a message deletion scheme.

In this paper, we present our deletion mechanism, which will be able to apply regardless of the routing protocols in DTN. Then, we evaluate our new mechanism after applying into three different routing algorithms, i.e. Epidemic [3], Spray and Wait [4] and PRoPHET [5]. We study its effects on the delivery performance, average latency, and overhead of these routing algorithms. Epidemic routing, replicates messages to all encountered peers. Spray-and-Wait routing aims to limit number of message copies to  $n$ . PRoPHET routing will make a copy only to the nodes that have higher delivery probability. We compare our results to MaxProp [6], a routing algorithm which already includes an explicit deletion method of delivered messages. Our results show that the proposed deletion mechanism, by avoiding wasting storage and bandwidth resources, can significantly improve the delivery probability, reduces the average latency and the overhead with all the routing algorithms.

This paper is organized as follows. Section II explains our deletion mechanism while we evaluate its performances in Section III. We present the related work in Section IV. We conclude this work in Section V and present the future work.

## II. DELETION MECHANISM AND MOBILITY PATTERNS

The objective of this research is to design an efficient message deletion mechanism for DTNs. This mechanism

must be efficient in terms of: (i) Ability to remove copies of delivered messages in timely manner; (ii) resources consumption (bandwidth, storage). To this end, we intend to make use of the knowledge of human mobility patterns in designing an effective ACK distribution mechanism.

The design can be separated into two parts: (i) Methods for distributing acknowledgment messages (ACK); (ii) The use of Helper nodes to relay ACK messages.

#### A. ACK distribution methods to delete replicas

ACK messages can be distributed in two different policies, passive or active. In passive distribution policy; ACK of any message ‘m’ will not express its knowledge to any encountering node unless the encountering node tries to send a copy of message ‘m’ to it. In this case, ACK message will be slowly distributed and copies of the delivery message will be deleted accordingly. For the active distribution policy; a node holding ACK of any message tries to express its knowledge to its encounters. The extreme case of this active distribution is to express its knowledge to every encounter (broadcast-like). Other active methods that fall in between can be named as multicast type or publish-subscribe policies, which target only express knowledge to the nodes in concern.

At our first step we have implemented a broadcast-like ACK distribution scheme where every node holds a list of known acked messages (active policy). When two nodes become in contact, they will exchange this list. If a message in their buffer matches an acked message in the list, they will delete this message from their buffer. The deletion algorithm for each node is presented in Fig. 1. Our future step is to design the multicast type or publish-subscriber type of ACK distribution in order to reduce the resource consumption while maintaining high performance.

#### B. Helper nodes (H-nodes) characteristics

As previously discussed, communications in Delay Tolerant Networks are often facilitated by the mobility of devices. Therefore understanding this mobility patterns can benefit in designing algorithms for DTNs. Another benefit is to obtain more realistic scenarios for future network simulations. We aim to use this knowledge to enhance the distribution of ACK messages in our deletion mechanism.

Static H-node: Some nodes, which are located at highly visited locations, will have higher chance to be in contact with other nodes. This implies higher possibility to distribute ACK messages to numerous distinguish nodes. For instance, in [7], it has been shown that there do exist locations, which are natural hotspots where people swarm. This finding coincides with the study of Barabasi et. al [8]. The fractal waypoints model in the study of Lee et. al also suggests similar characteristic [9]. In addition, we were able to observe the same phenomenon in the real GPS traces of human from New York City [10]. Based on this characteristic we can locate our H-nodes, which serve as the ACK messages distributors. Consequently, the deletion of the copies of delivered messages can be done more quickly; hence it can reduce the transmission, save bandwidth and storage consumption. Besides the geographical locations of hotspots, other characteristics are also important, such as average pause time, or the speed of nodes. The circumspect designed ACK distribution method will improve utilization of network resources.

Dynamic H-nodes: From [11] we observed that by adding trams and cars into simulation scenarios, it creates many more short-term contacts and thus improve communication in DTNs. As they move faster, they increase contact opportunities among nodes. It can be assumed that these high mobility nodes can also be used for distributing ACK messages. However, the short contact time is an issue since this contact time is also essential to route the message from a source to a destination. The use of dynamic H-nodes will be more deeply investigated in our future work.

### III. PERFORMANCE EVALUATION

#### A. Simulation Setup

To evaluate our mechanism we have simulated our algorithm in the Opportunistic Networking Environment (ONE) simulator [12]. This simulator provides DTN nodes characteristics, which can be easily customized such as heterogeneous node groups. It also supports Map-based movement through Geographic Information System (GIS) such as OpenJUMP [13], which enable us to simulate in more realistic environment.

In our simulations, we have used shortest path map-based movement model on the Helsinki city terrain 4500 x 3400 m with simulation length equivalent to 12 hours. Messages are generated by uniformly selected random sources, towards random destinations. The size of messages is also randomly chosen between 500KB-1MB. Our nodes exhibited Bluetooth properties: 250KB/s of bandwidth within 10m radio range. The buffer management policy used in our simulation is FIFO drop, default policy in the ONE simulator. We have simulated both homogeneous node properties and heterogeneous node properties where nodes are divided into groups with different speed, buffer size, and movement models, shown in Table 1.

```

1: //Notation
2: N: This node, P: Peer (node in contact)
3:
4: //Exchanging ackedMessageIds
5: if connection is up
6:   N.ackedMessageIds.addAll(P.ackedMessageIds)
7:   P.ackedMessageIds.addAll(N.ackedMessageIds)
8: //remove all message that messageId is in ackedMessageIds
9:   N.deleteAckedMessages()
10:  P.deleteAckedMessages()

```

Figure 1. Deletion algorithm

TABLE I. NODES SETTING IN SIMULATIONS

Group	#Nodes	Speed (m/s)	Buffer (MB)	Movement Model
1: Pedestrians	40	0.5-1.5	5	ShortestPathMapbased
2: Cars	40	2.7-13.9	5	ShortestPathMapbased
3: Pedestrians	40	0.5-1.5	5	ShortestPathMapbased
4: Trams (1)	2	7-10	50	MapRouteMovement
5: Trams (2)	2	7-10	50	MapRouteMovement
6: Trams (3)	2	7-10	50	MapRouteMovement

The results show the same trend so in this paper we chose to present only the results from heterogeneous setting scenario.

### B. Performance Evaluation

In order to evaluate our deletion mechanism, we have studied following parameters, which are essential to DTNs performance.

#### 1) Buffer size

We have simulated 4 scenarios where buffer sizes are 5, 10, 15, and 20MB for pedestrians and cars, 50, 100, 150, 200MB for trams. The results for all the considered routing algorithms, Epidemic, PRoPHET, and Spray and Wait, show that when increasing buffer space from 5MB for pedestrians and cars, 50MB for trams to 10MB and 100MB respectively, it increases the delivery performance by 43.3%. As we increase them to 15MB and 150MB, the delivery performance increases by 52% when compare to 5MB scenario. For 20MB and 200MB, the delivery performance increases by 60%. These results confirmed that buffer constraint is a cause of low delivery performance. However, there is a limit in the benefit while increasing the buffer size.

#### 2) Time-To-Live (TTL)

One of the proposed message deletion mechanisms is TTL (Time-To-Live) [14], where each message have restricted lifetime. Yet, the choice of an appropriate TTL value is difficult in DTNs. Moreover, different routing algorithms also respond to any TTL value differently. In our experiment, we have varied TTL values from 60 minutes to 240 minutes and compared them with the infinite TTL (Fig. 2). Epidemic and PRoPHET performed better when short TTL is introduced. The best case is at TTL equals to 60 minutes, which provides 30% improvement in delivery performance for Epidemic and PRoPHET when compare to the infinite TTL. Since these two routing algorithms have similar characteristics where number of copies of messages increase with time. Therefore introducing TTL helps limiting number of copies of messages in the network. In addition to the improvement of delivery performance, in these two routing algorithms TTL also helps in reducing the average latency and overhead. In contrast, Spray and Wait performs at its best when TTL is infinite, (Fig. 2). Since the delivery probability of this routing algorithm relies largely on the mobility of nodes, which allows direct contact between intermediate nodes (carrying the message) and the destination. Hence, the longer they hold the message, the

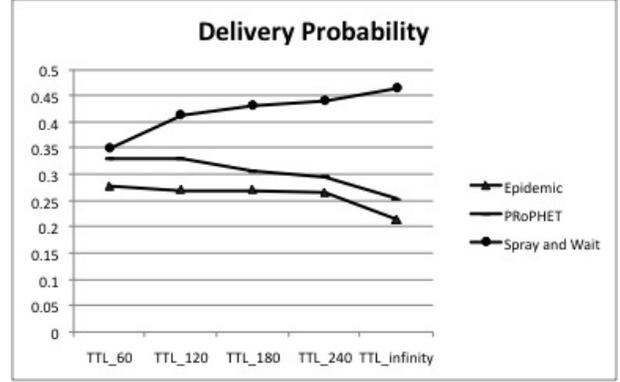


Figure 2. Delivery Probability of Epidemic, PRoPHET, and Spray and Wait routing protocols with different TTL values 60, 120, 180, 240 minutes and infinity.

better chance to meet the destination and successfully deliver the message. Introducing TTL to Spray and Wait can reduce average latency but not overhead.

#### 3) Feedback policy

We have compared our deletion mechanism (broadcast-like ACK message) with the “default scenario”, where no deletion mechanism is used and with “One hop delete delivered” mechanism provided by the ONE simulator, where destination generates ACK message when receiving message (m) and responds to the immediate sender. Then the immediate sender deletes m from its buffer. The results are shown in Figure 3. From the graph, with broadcast-like ACK messages mechanism, we can achieve 100% improvement on delivery performance for Epidemic, 97.5% and 8% for PRoPHET and Spray and Wait respectively. Overhead can be reduced by 61% in Epidemic, 63.5% in PRoPHET and 10% in Spray and Wait. When compare its results with MaxProp, Spray and Wait outperforms MaxProp in term of latency and overhead. While the “one hop delete delivered” mechanism shows a small improvement on delivery probability and reduced overhead, our method shows significant improvement in delivering probability and reducing overhead, especially for PRoPHET and Epidemic. Since these two algorithms do not limit the number of message replicas, their buffers are more likely to be overwhelmed and benefit from our deletion mechanism. On the contrary, our algorithm shows less improvement on Spray and Wait, since it already has a mechanism to limit message replicas. From these results, we can imply that copies of delivered messages, which are in the network, waste a lot of network resources.

## IV. RELATED WORK

To the best of our knowledge, there has not been any work regarding standalone deletion mechanism in DTNs. Indeed, storage management in DTNs was generally integrated with some routing algorithms such as Time-To-Live (TTL) in Epidemic, hop-based TTL in Spray and Wait, anti-packet in MaxProp, passive-cure in potential-based entropy adaptive

routing (PEAR) [15]. Other works are focusing on buffer management policies. We believe that each proposed routing scheme has its own strengths. A standalone deletion mechanism will therefore help to enhance the performance of these routing schemes. To cope with the problem of buffer overflow/limited storage, one has to consider both preventive measure and procedure to handle the situation once occurs: (i) Deletion mechanism is designed to control the buffer space usage by only allowing message with valid status, valid time-to-live (TTL), or has not yet been delivered, in the buffer otherwise it will delete the message; (ii) buffer management policies will only take action when node's buffer becomes full. It will decide, which packet has to be dropped. Deletion mechanisms that have already been proposed are Time-to-live (TTL); global TTL, local TTL, hop-based TTL, and time-based TTL [14]. The message will be deleted from the network when its lifetime is over. In DTNs delay can be extensively large, so it is difficult to find an appropriate TTL value. Small TTL can result in low delivery rate, where large TTL can result in exhausted storage space by numerous copies of delivered messages. Another deletion mechanism is anti-packet or passive cure [16], where destination creates the ACK message once it has received the message. This ACK will be forwarded to only the nodes that are still trying to forward the delivered packet in passive mode described in section III. This method has reduced bandwidth overhead but results in long propagation delay, which eventually results in long-held of copies of delivered messages.

## V. CONCLUSION AND FUTURE WORK

In this work, we show that the well design of a message deletion mechanism can enhance the performance of DTN. Indeed, the inherent limited resources of DTNs will be better utilized. By efficiently deleting copies of delivered messages in the network, we can reduce storage consumption and avoid the nodes from wasting their bandwidth by distributing messages that have already been delivered.

Currently, we are taking into account the human mobility to improve our deletion mechanism. Our future work will also include the use, of multicast or publish-subscribe ACK distribution method in order to reduce overhead used in deleting useless messages. In addition, we will implement

the mobility pattern based on our studies into the simulation tool for our evaluation. We aim to simulate our deletion mechanism in the simulated environment that closely relate to real world before expand our work into practice.

## REFERENCES

- [1] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," Proc. of ACM SIGCOMM 2003.
- [2] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," Proc. of ACM SIGCOMM, Aug. 2004.
- [3] A. Vahdat, D. Becker, "Epidemic routing for partially-connected ad hoc networks", Technical Report CS-200006, Duke University.
- [4] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," Proc. ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN) 2005.
- [5] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," ACM SIGMOBILE Mobile Computing and Communications Review, 7(3): July. 2003.
- [6] J. Burgess, B. Gallagher, D. Jensen, and B. Neil Levine, "MaxProp: Routing for vehicle-based disruption-tolerant networks," Proc. IEEE INFOCOM, April. 2003.
- [7] F. Bai, N. Sadagopan, and A. Helmy, "Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks," Proc. IEEE INFOCOM, 2003.
- [8] A. Barabasi, M. Gonzalez, C. A. Hidalgo, "Understanding individual human mobility patterns" Nature, vol. 453, Jun. 2008, pp. 779-782.
- [9] K. Lee, S. Hong, S.J. Kim, I. Rhee, S. Chong, "SLAW: A Mobility Model for Human Walks," INFOCOM, Brazil 2009.
- [10] "CRAWDAD metadata: ncsu/mobilitymodels (v.2009-07-23)," <http://crawdad.cs.dartmouth.edu/meta.php?name=ncsu/mobilitymodels, 2009>.
- [11] A. Keranen, J. Ott, "Increasing Reality for DTN Protocol Simulations," Helsinki University of Technology. Networking Laboratory, July 2007.
- [12] A. Keranen, J. Ott, T. Karkkainen, "The ONE Simulator for DTN Protocol Evaluation," SIMUTOOL, 2009.
- [13] "OpenJUMP – The free Java based and open source Geographic Information System for the World," <http://openjump.org>, 2007.
- [14] W.H. Yuen, and H. Schulzrinne, "Message Replication and Deletion in Delay Tolerant Networks under Hop-based and Time-based TTL Schemes". Columbia University.
- [15] H. Ochiai, and H. Esaki, "Mobility Entropy and Message Routing in Community-Structured Delay Tolerant Networks," AINTEC, November. 2008.
- [16] K. A. Harras, Challenged Networks Protocol and Architecture Challenges in Delay and Disruption Tolerant Networks, pp. 31-62.

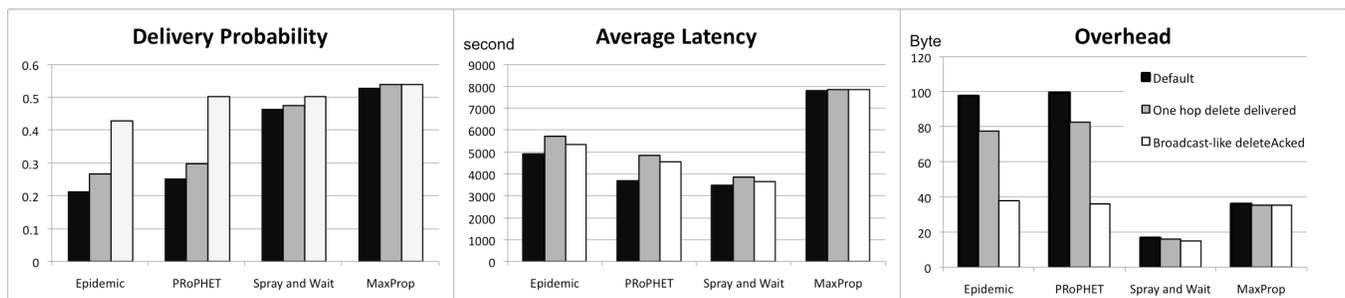


Figure 3. Performance of Epidemic, PRoPHET, Spray and Wait and MaxProp routing protocols in 3 scenarios. (Black) Default scenario with no deletion mechanism. (Gray) with the "one hop delete delivered" mechanism. (White) with the "broadcast-like ACK messages" mechanism.