Redundancy Elimination in DTN via ACK Mechanism

Xiqing Zhang
Ocean University of China
No 238 Songling Road
Laoshan Qingdao China
zhangxiqing2005@163.com

Yuan Feng*
Ocean University of China
No 238 Songling Road
Laoshan Qingdao China
fengyuan@ouc.edu.cn

Feng Hong
Ocean University of China
No 238 Songling Road
Laoshan Qingdao China
hongfeng @ouc.edu.cn

Zhongwen Guo
Ocean University of China
No 238 Songling Road
Laoshan Qingdao China
guozhw@ouc.edu.cn

ABSTRACT

The traditional routing protocols for delay tolerant networks (DTN) usually take the strategy of spreading multiple copies of one message to the networks. When one copy reaches destination, the transmission of other copies not only waste the bandwidth but also deprive other messages of the opportunities for transmission. This paper brings up a mechanism to eliminate the redundant copies. By adding an acknowledge field to the packet header to delete redundant copies, it can degrade the network overhead while improve the delivery ratio. Simulation results confirm that the proposed method can improve the performance of epidemic and Spray and Wait routing protocol.

Categories and Subject Descriptors

C.2.2 [**Computer Systems Organization**]: Network Protocols – *Routing protocols*

General Terms

Design

Keywords

DTN, redundancy elimination, ACK

1. INTRODUCTION

Delay tolerant networks (DTNs) [1], which are characterized by intermittent connection, high transfer delay and error rate, attract increasing interest these years. Initially, it is proposed for Inter Planetary Network (IPN), and now it covers many new networks, such as wildlife tracking networks, packet switched networks and military networks. DTN is characterized by the following points: 1) Intermittent connection, the topology is dynamic and changing; 2) Long or variable delay; 3) Asymmetric data rates.

.

Since in the DTN model there may not exist an end-to-end path between a source and a destination, conventional ad-hoc network routing schemes, such as DSR [2] and AODV [3], etc, would fail. However, this does not mean that packets can never be delivered in such networks. Over time, different links come up and down due to node mobility. It implies that a message could be sent over an existing link, got buffered at the next hop until the next link in the path comes up, until it reaches its destination.

Using such kind of store-carry-forward method, researchers have proposed many routing protocols, such as Epidemic routing [4], PRoPHET [5] and Spray-and-Wait/Focus [6], etc. All these protocols transmit messages by spreading certain amount copies of the messages to the networks. As a result, it is unavoidable that there are some unnecessary transmissions. For example, one copy of a message may have reached its destination, but its counterparts in other nodes still reside in the networks. The nodes carrying these copies have no knowledge about the global result and may still transmit these copies. It not only wastes the bandwidth resource but also may deprive other messages of the valuable opportunities for transmission.

This paper presents a mechanism to eliminate these redundant copies by adding an acknowledge field to the packet header when sending messages. The acknowledge field can indicate which messages have reached their destinations. The intermediate nodes receiving the message can delete the copies of messages in their own buffer that have reached destinations according to the acknowledge field, while the intermediate nodes can also update the acknowledge field when transmit this message next time for better performance. We used ONE [7] to verify the proposed method. Simulation results show that our proposal can degrade the network overhead while improve the delivery ratio of routing protocols of both epidemic and Spray and Wait.

2. RELATED WORK

A large number of routing protocols for DTN have been proposed in the past. In many of them, all nodes are assumed to be mobile. Depending on the number of copies of a single message that may coexist in the network, one can define two major categories of hop-by-hop routing schemes, namely single-copy routing scheme and multiple-copy routing scheme [8].

*corresposding author

MOBIMEDIA 2015, May 25-27, Chengdu, People's Republic of China

DOI 10.4108/icst.mobimedia.2015.259046

In single-copy routing scheme there's only a single custodian for each message. When the current custodian forwards the copy to an appropriate next hop, this becomes the message's new custodian, and so on until the message reaches its destination.

Different from single-copy routing scheme, multiple-copy routing scheme may generate multiple copies of the same message which can be routed independently for increased robustness. The multiple-copy schemes can be further classified as unlimited-copy schemes or limited-copy schemes. The unlimited-copy schemes mean that protocols don't control the number of copies of the same message. The classical routing protocol of this scheme is Epidemic routing. In [4] the concept of epidemic algorithms is applied to intermittently connected mobile networks. The proposed epidemic routing scheme is essentially flooding. There are also some other schemes of this form [9,10].

To reduce the transmission overhead of unlimited-copy scheme, limited-copy schemes were proposed. Spay and Wait routing [11] is a representative one. There are two phases in Spray and Wait. First, spray phase, for every message originating at a source node, L message copies are initially spread (forwarded by the source and possibly other nodes receiving a copy) to L distinct "relays". Second, wait phase, if the destination is not found in the spraying phase, each of the L nodes carrying a message copy performs direct transmission (i.e. will forward the message only to its destination). To improve the performance, Spray and Focus [6] was proposed, in which the wait phase was changed into focus phase that means the copies try getting close to the destination rather than wait.

There is no doubt that both the limited-copy schemes and unlimited-copy schemes have redundancy. The initial aim of spreading many copies is to increase the delivery ratio. But once a copy reaches the destination, the transmission of other copies will cause senseless waste of bandwidth and buffer space of nodes. At present, there are three ways to delete these redundant copies:

[1] Set TTL (Time To Live) for every message. The deficiency is that it is hard to set the value for the messages that don't have time limitation.

[2] The destination node broadcasts an acknowledge message to the network after it received a message. Because of dynamic topology of DTN, it is hard to design a broadcast mechanism.

[3] The node drops the copies when its buffer is fully occupied. Some dropping strategies have been proposed in [12, 13, 14]. Although these strategies have certain effects, they are always blind and passive.

Unlike the measures above, we bring up a new mechanism to eliminate the redundancy in an active and directional manner.

3. DETAILED DESIGN

3.1 Network Model

In our design the DTN model is assumed to have the following characteristics:

[1] All nodes in the network are mobile and they move independently.

[2] Every node has limited-sized buffer to carry messages. The bandwidth and communication range are also limited.

[3] Two nodes can communicate when they are in the communication range of each other.

[4] Every node may send messages to all other nodes in the network.

3.2 Detailed Design

The basic design idea of this paper is that nodes delete redundant copies while transmit messages. So every node needs to keep record of the messages that it has received recently. The records can be kept in a received-message list. The element of this list is the ID of received message and the received-time. It won't occupy too much buffer space. When the list is full, it can be updated in the principle of FIFO (first in first out).

3.2.1 Source Node

When the source node sends message, it will add an acknowledge field to the packet header. The acknowledge field is constituted by a certain amount of entries. One entry includes a received-message's ID and its received-time. The structure of one entry can be seen in Figure 1. The acknowledge field can be seen as a collection of entry like this.

Message ID	**Received time**

Figure 1. One entry of Acknowledge field

When sending message to one neighbor, the source node will scan the received- message list to find **n** (**n** is the number of entry in acknowledge field) messages that received most recently and fill the messages' ID and received-time in the acknowledge field. After encapsulation, the source node transmits the packet to its neighbor.

3.2.2 Intermediate Node

If the relay node is a intermediate node for the message (not the final recipient), after having received the message, the intermediate node resolves the acknowledge field and updates its buffer space. Specifically, the intermediate node extracts all the entries of the acknowledge field one by one, and then check its own buffer space. If it finds one message in its buffer is identical with the ID in the entry, which means that the message has reached its destination, the intermediate node will drop this copy. After all the entries having been checked, the message will be put into the node's buffer space, waiting to be transmitted again. The process of deleting redundant copies in intermediate node can be seen in Figure 2.

When the received message is transferred next time by intermediate node, the message's acknowledge field can be updated. That is because, first the acknowledge field may not be full, and second the messages indicated by acknowledge have reached destination for a long time. So some entries in acknowledge field can be replaced by new message received more recently by the intermediate node. The substitution strategy can be various, but one key criterion is the received-time, that is why we add received-time to the acknowledge field.

3.2.3 Destination Node

When the message is finally received by its destination node, the destination node will also drop the redundant copies in its buffer like the intermediate node. After this, the destination node will push the ID of the message and the message's received-time to its

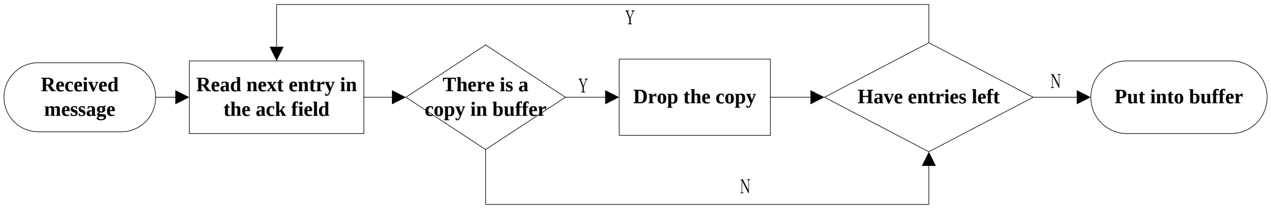

Figure 2. Flow chart of deleting redundant copies

received-message list. When the destination node sends another message, it can fill the ID of the message just received in the new message's acknowledge field.

4. SIMULATION

4.1 Simulation Environment Setting

To verify our thought, we use The ONE[7] to simulate it. The simulation scene is in size of 4500m*3400m. There are 6 groups of nodes. The concrete parameters of every group are presented in TABLE I.

Table 1: Parameters setting

group type	pedestrian group 1,2	car group 3	tram group 4,5,6
number of nodes/group	40	40	2
speed range m/s	[0.5,1.5]	[2.7,13.9]	[7,10]
transmit range /m	10	10	50
movement model	random movement	shortest path	map route

We don't list the size of nodes' buffer in TABLE I, because it is a key parameter in simulation. The role of eliminating redundant copies lies on the number of message one node can carry. Therefore, we take the buffer size as a variable in simulation. The base is that: pedestrians and cars' buffer size is 5M, and trams' buffer size is 50M. We magnified it two or more times in simulation.

The message's size in our simulation is in [500kB, 1MB].Message generating interval is [25s, 35s]. There is no TTL for every message, so the copies of message are dropped owing to either that the buffer is full or our redundancy eliminating scheme.

We implemented the scheme on two typical routing protocols: Epidemic and SNW(spray and wait). The length of acknowledge field is 10, that is to say, every packet can denote 10 received-messages. The substitution strategy is that, in intermediate node, the earlier received-messages are replaced by newer received-messages. In SNW, the initial number of copies is 16 and the spraying method is binary spray.

As we mentioned above, we magnified buffer size 2, 3,4 times respectively. The simulation time for each condition is 96 hours.

4.2 Result Analysis

The main evaluation criteria are two: delivery ratio and overhead ratio. The delivery ratio is percentage of messages that reach their destinations in all messages. The overhead ratio is expense of transmitting one message successfully. It is calculated by the times of relays divided into the number of messages transmitted to destination.

We compare the performance of two routing protocols with acknowledge field and usual protocols without acknowledge field.

4.2.1 Delivery Ratio

Because of limited resources in real DTN environment, for example, limited buffer space, limited bandwidth, etc, not all the messages can reach destinations. The delivery ratio indicates how many percents of messages that can be routed to destinations. It is the efficiency of protocols. It is calculated by formula (1).

$$\text{delivery ratio} = \frac{\text{number of messages reached destinations}}{\text{number of all the message}} * 100\% \quad (1)$$

As shown in Figure 3, the scheme presented in this paper can boost the delivery ratio in both Epidemic and SNW protocol. That is because the scheme can delete the redundant copies in time.

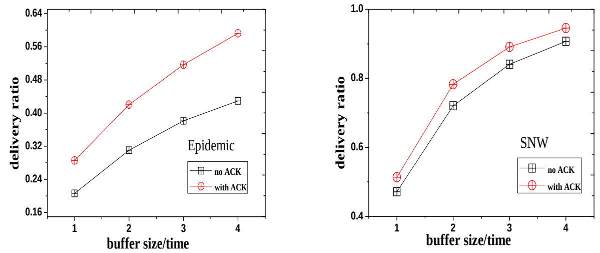

Figure 3. Comparison of delivery ratio

First, deleting redundant copies can save valuable buffer space. Because the size of buffer space of every node is limited, so the quantity of messages they can carry is also limited. When buffer space is fully occupied, the node has to drop some messages to make room for new message. Without acknowledge mechanism, the node may drop the messages that haven't reached destinations. After adding acknowledge mechanism, the node can drop the messages that have reached destinations, which make node have free space almost any time, avoiding dropping messages that haven't reached destinations.

Second, deleting redundant copies can save the opportunities for transmission. In DTN environment, because of mobility, the connection is intermittent. It may last for only a few seconds, even can't afford to transmit an integral message. The acknowledge mechanism removes the redundant messages, so these messages won't be transmitted. The opportunities for transmission can be used to transmit messages that haven't reach destinations.

4.2.2 Overhead ratio

The overhead ratio is the utility presentation of protocols. In this paper, it is the average number of relays needed to transmit a message to destination. The smaller it is, the better the protocol is. It is calculated by the formula (2).It is noteworthy that number of all relays includes the relays that don't result in messages' being transmitted to destinations.

$$\text{overhead ratio} = \frac{\text{number of all relays}}{\text{number of messages reached destinations}} \quad (2)$$

Figure 4 shows the decrease of overhead ratio when using the acknowledge mechanism in both Epidemic and SNW protocol.

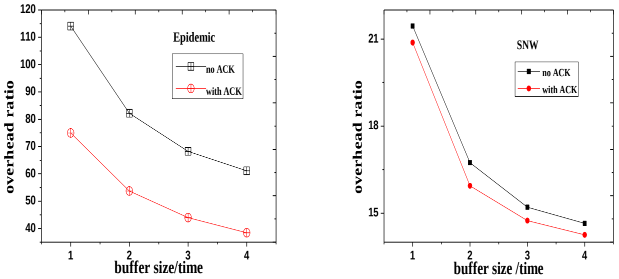

Figure 4. Comparison of overhead ratio

The reasons are that: first, as discussed above, acknowledge mechanism boosts the delivery ratio, so the number of messages that have reached destinations increases. This means the denominator increases. Second, Acknowledge mechanism deletes redundant copies, so they won't cause useless relays. The numerator decreased. Numerator's decrease and denominator's increase make the result of overhead ratio become less.

5. CONCLUSION

To solve the problem of redundant copies in routing protocols of ICMN, we proposed an ACK mechanism, that is adding an acknowledge field in data packet header. The acknowledge field can indicate which messages have been received by their final recipients. The node that received messages with acknowledge field can delete the redundant copies in its buffer that the acknowledge field shows. The Simulation reveals this scheme can improve networks' performance. It boosts the delivery ratio while decreases the overhead ratio.

6. ACKNOWLEDGE

This research is partially supported by NSF program of China under granted number of 61379128 and 61379127.

7. REFERENCES

[1] Fall K. A delay-tolerant network architecture for challenged Internets. In: Proc. of the 2003 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications. Kalsruhe: ACM，2003: 27-34 .

[2] D. B. Johnson, D. A. Maltz, and J. Broch. Ad hoc networking, chapter 5 - DSR: the dynamic source routing protocol for multihop wireless adhoc networks Addison-Wesley, 2001, 139–172.

[3] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, *Ad-hoc on-demand distance vector routing*. IETF MANET DRAFT, 2003.

[4] Vahdat A, Becker D, *Epidemic routing for partially connected ad hoc networks*, Technical Report, CS-2000-06, Duke University, 2000.

[5] Lindgren A, Doria A, Schelén O, Probabilistic routing in intermittently connected networks, ACM SIGMOBILE Mobile Computing and Communications Review, 7(3), 2003. 19−20.

[6] Spyropoulos T, Psounis K, Raghavendra CS, *Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility*. Proc. of the IEEE PerCom Workshop on Intermittently Connected Mobile Ad Hoc Networks. 2007.

[7] The Opportunistic Network Environment simulator. http://www.netlab.tkk.fi/tutkimus/dtn/theone/

[8] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, *Single-copy routing in intermittently connected mobile networks*. In Proc. of IEEE Secon'04, 2004.

[9] A. Beaufour, M. Leopold, and P. Bonnet. *Smart-tag based data dissemination*, In Proc. ACM International Workshop on Wireless Sensor Networks and Applications(WSNA'02), 2002.

[10] N. Glance, D. Snowdon, and J.-L. Meunier.,*Pollen: using people as a communication medium*，Computer Networks, 35(4):pp.429–442, Mar. 2001

[11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, *Spray and wait: Efficient routing in intermittently connected mobile n networks*, In Proceedings of ACMSIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

[12] LI Yun, ZHAO Ying, LIU Zhan-jun, et al, N-Drop: congestion control strategy under epidemic routing in DTN, Proc of International Conference on Wireless communications and Mobile Computing, New York: ACM，2009. 457-460.

[13] KRIFA A, BARAKA C, SPYROPOULOS T, et al, *Optimal buffer management policies for delay tolerant networks*, Proc of the 5th Annual IEEE Communications Society Conference, 2008. 260-268.

[14] WANG Gui-zhu, WANG Bing-ting, ZHANG Jia-yong, *Adaptive congestion control strategy based on binary spray and wait routing in DTN*. Application Research of Computers. 27(11), Nov. 2010. 4237–4241.