PAPER

A Priority Routing Protocol Based on Location and Moving Direction in Delay Tolerant Networks

Jian SHEN[†a)], ***Nonmember***, **Sangman MOH**[†b)], ***Member***, ***and*** **Ilyong CHUNG**[†c)], ***Nonmember***

SUMMARY Delay Tolerant Networks (DTNs) are a class of emerging networks that experience frequent and long-duration partitions. Delay is inevitable in DTNs, so ensuring the validity and reliability of the message transmission and making better use of buffer space are more important than concentrating on how to decrease the delay. In this paper, we present a novel routing protocol named *Location and Direction Aware Priority Routing (LDPR)* for DTNs, which utilizes the location and moving direction of nodes to deliver a message from source to destination. A node can get its location and moving direction information by receiving beacon packets periodically from anchor nodes and referring to received signal strength indicator (RSSI) for the beacon. LDPR contains two schemes named transmission scheme and drop scheme, which take advantage of the nodes' information of the location and moving direction to transmit the message and store the message into buffer space, respectively. Each message, in addition, is branded a certain priority according to the message's attributes (e.g. importance, validity, security and so on). The message priority decides the transmission order when delivering the message and the dropping sequence when the buffer is full. Simulation results show that the proposed LDPR protocol outperforms epidemic routing (EPI) protocol, prioritized epidemic routing (PREP) protocol, and DTN hierarchical routing (DHR) protocol in terms of packet delivery ratio, normalized routing overhead and average end-to-end delay. It is worth noting that LDPR doesn't need infinite buffer size to ensure the packet delivery ratio as in EPI. In particular, even though the buffer size is only 50, the packet delivery ratio of LDPR can still reach 93.9%, which can satisfy general communication demand. We expect LDPR to be of greater value than other existing solutions in highly disconnected and mobile networks.

key words: *delay tolerant networks (DTNs), location and direction aware priority routing (LDPR), priority, buffer size*

1. Introduction

DTNs are a practical class of emerging networks, which are occasionally connected networks comprised of one or more protocol families, and they experience frequent and long-duration partitions as well as long delays. Because there is no guarantee of end-to-end connectivity in DTNs, the routing protocols which have good performance in conventional networks are not suitable for DTNs, which are characterized by latency, bandwidth limitations, error probability, node longevity, or path stability [1].

The applications in DTNs must be delay tolerant in order to operate effectively in environments subject to significant delay or disruption [16]. In particular, our protocol can be applied in the following applications under different environments: (i) in the applications of terrestrial mobile networks where the networks may become unexpectedly partitioned due to node mobility or the networks may be expected to be partitioned in a periodic and predictable manner; (ii) in the applications of ad-hoc networks in battlefields and disaster areas where the networks may be expected to operated in hostile environments with disconnections due to mobility of the nodes or intentional jamming.

The simplest solution to the DTN routing problem is brute-force unconstrained replication or epidemic routing (EPI) [2]. A number of ideas have been explored to improve the efficiency of EPI, including prioritized epidemic routing (PREP) for opportunistic networks [3] and probabilistic routing in intermittently connected networks [4]. The use of network topology to estimate the transmission path and to increase the efficiency of routing has been studied in [5], [6].

In general, routing protocols in DTNs are classified into two categories based on which property is used to find the destination: flooding family and forwarding family. To find the destination, two different approaches of replication and knowledge are used. The replication is used in the flooding strategy in which different algorithms can be used to make multiple copies of a message and to manage those copies. The knowledge is used in the forwarding strategy in which different approaches can be used to obtain some network state information and then use it for making routing decisions [6].

Delay is inevitable in DTNs, thus, ensuring the validity and reliability of the message transmission and making better use of buffer space is more important than concentrating on how to decrease the delay. In this paper, we present a novel routing protocol for DTNs called Location and Direction Aware Priority Routing (LDPR). Just as the name implies, LDPR utilizes the location and moving direction information of nodes to deliver a message from source to destination. A node can get its location and moving direction information by receiving beacon packets periodically from anchor nodes and referring to received signal strength indicator (RSSI) [15] for the beacon. Two schemes, named transmission scheme and drop scheme, take advantage of the nodes' information of the location and moving direction in transmitting the message and in storing the message into buffer space, respectively. Each message, in addition, is branded a certain priority according to the message's attributes (e.g. importance, validity, security and so on). The

Manuscript received October 7, 2009.
Manuscript revised April 14, 2010.

[†]The authors are with the Department of Computer Engineering, Chosun University, Gwangju, 501–759 Republic of Korea.
a) E-mail: s_shenjian@126.com
b) E-mail: smmoh@chosun.ac.kr
c) E-mail: iyc@chosun.ac.kr

DOI: 10.1587/transinf.E93.D.2763

priority decides the transmission order when delivering the message and the dropping sequence when the buffer is full.

Compared with other proposed routing protocols in DTNs, the most distinguished difference in LDPR is that some anchor nodes are deployed in a certain area to help to determine the location and moving direction information of the nodes by using RSSI. In this way, we can easily and accurately deliver the message from the source to the destination without completely depending on message replication or thinking about the network topology information. Obviously, LDPR belongs neither to the flooding family nor to the forwarding family. All the routing protocols in DTNs have the common objective of trying to increase the delivery ratio while decreasing resource consumption and latency. In this paper, LDPR can satisfy the requirement of the delivery ratio as much as possible and can make better use of buffer space. As is well known, buffer space, reliability and resource consumption are important issues for routing protocols in DTNs. Simulation results show that the proposed LDPR protocol outperforms EPI, PREP, and DTN hierarchical routing (DHR) protocol in terms of packet delivery ratio, normalized routing overhead and average end-to-end delay based on different buffer size and different radio transmission range. Moreover, LDPR doesn't need infinite buffer size to ensure the packet delivery ratio as in EPI. In particular, even though the buffer size is only 50, the packet delivery ratio of LDPR can still reach 93.9%, which can satisfy general communication demand.

The rest of this paper is organized as follows. In the next section, the related work on the delay tolerant routing protocols is briefly discussed. Location and Direction Aware Priority Routing Protocol is described in detail in Sect. 3. Simulations and results are presented in Sect. 4. Finally, the conclusions of this paper are covered in Sect. 5.

2. Related Work

Delay tolerant networks are a kind of application of Mobile Ad Hoc Networks (MANETs). The connection between the nodes, in DTNs, is intermittent and unstable because of node mobility. Some traditional routing protocols which performed very well in ad hoc networks may not perform well for DTNs, such as DSDV [7], DSR [8], and AODV [9]. Some researchers have made a lot of effort in designing new protocols in this special field. In general, the routing protocols in DTNs are classified into two categories based on which property is used to find the destination: flooding family and forwarding family. To find the destination, two different approaches of replication and knowledge are used. The replication is used in the flooding strategy and there are many algorithms to manage multiple copies of a message and to make those copies. The knowledge is used in the forwarding strategy. Some studies have been devoted to derive more efficient methods to obtain some network state information and then use it to make routing decisions [6], [10]. One of the earliest proposals for routing in delay tolerant networks is EPI [2]. In EPI, all the nodes can become the carriers. It ensures a high probability of message delivery. Moreover, a number of ideas have been explored to improve the efficiency of EPI, including PREP [3] and probabilistic routing [4]. The key idea of PREP is to impose a partial ordering on the message called bundle. In probabilistic routing, when a message arrives at a node which does not have an available contact with another node, it must be stored in the buffer until the node encounters another node. We should set a probability threshold on the nodes. It only permits that a node can receive the message when its delivery probability exceeds the threshold.

In addition, the use of network topology to estimate the transmission path and to increase the efficiency of routing has been studied in [5], [6], such as source routing, per-hop routing, per-contact routing and DHR. These protocols utilize the network topology information to effectively select the best path. Then, the message is forwarded from node to node along this path. In [5], [6], each node typically sends a single message along with the best path, so these protocols do not use replication.

2.1 Epidemic Routing

One of the earliest proposals for routing in delay tolerant networks is EPI [2]. Epidemic routing (EPI), as the name suggests, likes the pattern of pandemic virus transmitting. In EPI, all the nodes can become the carriers, which can take the message from one node to another. In this way, messages are quickly distributed through the networks due to node random mobility. Moreover, EPI relies upon carriers coming into contact with another in the network through node mobility. We assume that: (i) the sender does not know where the receiver is currently located or the best "route" to follow, (ii) the receiver may also be a roaming wireless host, and (iii) pairs of hosts (not necessarily the sender and receiver) periodically and randomly come into communication range of one another through node mobility [2]. Using EPI protocol, messages can be delivered successfully with a high probability. However, network resources are heavily consumed.

To explain the process of EPI, we give an example as depicted in Fig. 1. In EPI [2], each host stores a *summary vector* that indicates which entries in their local hash tables are set. When host A comes into transmission range of host B, an anti-entropy session is initiated. In the first step, A transmits its summary vector (called SV_A) to B. SV_A is a compact representation of all the messages which are

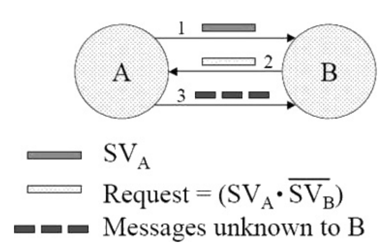


Fig. 1 The process of epidemic routing protocol.

Table 1 Comparison of the flooding families.

	Hop count	Number of copies	Resource usage	Delivery ratio	Routing vector/table	Multipath support	Effectiveness	Latency
Direct contact	1	No	Low	Min	No	No	Bad	Long
Two-hop relay	2	n[(1)]	Low	Low	No	Yes	Bad	Long
Tree-based flooding	Many	$\sum_{i=0}^{n}\sum_{a=1}^{k} M_a$[(2)]	High	Low	No	Yes	Bad	Long
Epidemic routing	Many	Unlimited	Max	Max	Yes	Yes	Normal	Long
Prioritized epidemic routing	Many	Limited	Limited	Max	Yes	Yes	Good	Normal
Probabilistic routing	Many	Limited	Limited	Normal	Yes	No	Good	Normal
RUNES	Many	Limited	Limited	Normal	Yes	Maybe	Good	Long

[(1)] n is the number of the nodes in a network.
[(2)] n is the depth of a routing tree, k is the number of nodes at the same depth, and M_a is the number of copies of a message in node a.

buffered at A. Second, B performs a logical AND operation between the negation of its summary vector (given a symbol like $\neg SV_B$) and SV_A. We can easily conclude that the negation of B's summary vector represents the messages that B has never seen. Compared with A's summary vector, B needs to find the set difference between the messages buffered at A and the messages buffered locally at B. Then, B transmits a vector requesting these messages from A. In the third step, A transmits the requested messages to B. This process is repeated continuously when B comes into contact with a new neighbor. Given sufficient buffer space and time, these anti-entropy sessions guarantee the message can be eventually delivered to the destination.

The critical resource in EPI is the buffer. An intelligent buffer management scheme is employed to improve the delivery ratio over the simple FIFO scheme. The buffer policy in EPI is to drop packets that are the least likely to be delivered based on previous history. If node A has met B frequently, and B has met C frequently, then A is likely to deliver messages to C through B. Similar metrics are used in a number of epidemic protocol variants [3], [4], where the buffer policies take advantage of physical locality and the fact that movement is not completely random. However, these protocols still transmit many copies of each message, making them very expensive.

2.2 Flooding Family Routing Protocols in DTNs

In the flooding family, each node has a number of copies of each message and transmits them to a set of nodes (sometimes called relays). All the relays maintain the copies and store them in their buffer space until they connect with the next node. The earliest studies in the area of DTN routing fall into this family. Using message replication can increase the probability of message delivery. The basic protocols in this family do not need any information about the network. However, if some knowledge of the network is referred to as an additional routing metric, the flooding strategy can be significantly improved. Direct contact [6], [10], two-hop relay [6], [10], tree-based flooding [6], [10], EPI [2], PREP [3], probabilistic routing [4], and reconfigurable ubiquitous networked embedded systems (RUNES) routing protocols belong to the flooding family [10].

We evaluated the flooding family routing protocols in terms of various characteristics including important performance metrics. Hop count, the number of copies, resource usage, delivery ratio, routing vector/table, multipath support, effectiveness, and latency are studied in the comparative analysis. Table 1 summarizes the comparison results of the flooding family routing protocols.

From the comparison Table 1, some conclusive comments can be inferred: Prioritized epidemic routing (PREP) is the best of the flooding family routing protocols even though it has some drawbacks such as poor resource usage.

2.3 Forwarding Family Routing Protocols in DTNs

In the forwarding family, network topology information is effectively utilized to select the best path, and the message is then forwarded from node to node along the path. Note that the routing protocols in this family require some knowledge about the network. The nodes typically send a single message along the best path, so they do not use replication. Location-based routing [6], [10], source routing [6], [10], per-hop routing [13], per-contact routing [13], and DHR [14] belong to the forwarding family.

We evaluated the forwarding family routing protocols as well, where flexibility, resource consumption, information usage, routing vector/table, scalability, loop freedom, effectiveness, delivery ratio, and latency are studied and compared. Table 2 summarizes the comparison results of the forwarding family routing protocols.

From the comparison Table 2, DTN hierarchical routing (DHR) can be primarily chosen thanks to its many out-

Table 2 Comparison of the forwarding families.

	Flexibility	Resource consumption	Information usage	Routing vector/table	Scalability	Loop-free	Effective-ness	Delivery ratio	Latency
Location based routing	Bad	Little	Little	No	Bad	Yes	Bad	Min	Normal
Source routing	Bad	Normal	Normal	No	Bad	Yes	Bad	Low	Long
Per-hop routing	Bad	Normal	Normal	No	Bad	Yes	Bad	Low	Long
Per-contact routing	Good	Many	Many	Yes	Bad	No	Normal	Normal	Normal
DTN hierarchical routing	Good	Many	Many	Yes	Good	Yes	Good	Max	Normal

standing features although it has two negative characteristics of poor information aggregation and information compression.

3. Location and Direction Aware Priority Routing Protocol

3.1 Key Idea

Before describing Location and Direction Aware Priority Routing Protocol (LDPR) in detail, we briefly present the main idea. We make use of anchor nodes to estimate the location and moving direction information of the nodes. Depending on this information, we choose the best next hop to relay the message. During this process, priority is employed to decide which message should be delivered first among lots of messages wanting to be transferred. At the same time, when the buffer space in the relay node is full, priority is also taken advantage of to determine which message should be dropped or be transferred to other nodes that have available buffer space.

The network should be initialized first to successfully deliver the message. All the nodes are deployed in a given area. Two kinds of nodes exist in the network. One is the anchor node, the other is the general node. Anchor nodes are equipped with GPS while general nodes are not. General nodes have the same radio transmission range and move randomly, while anchor nodes can assist in computing the location and moving direction information of general nodes by using RSSI [15]. In order to preferably route data from source to destination, we pre-determine some properties about the anchor nodes:

- All the anchor nodes have enough energy and capability of storing.
- Radio transmission range of the anchor node is large enough to cover the whole scale of the network.
- Location of the anchor nodes can be exactly obtained by GPS or other assisting methods.
- All the anchor nodes can move randomly around the network.

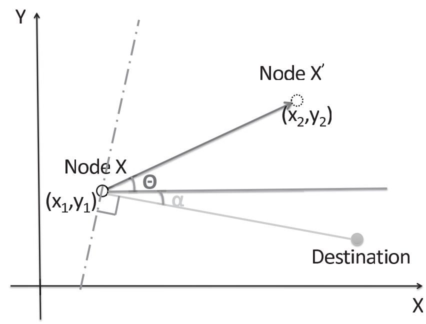


Fig. 2 An example of calculating node's moving direction by using location information.

General nodes obtain the location information by making use of RSSI. As we know, in RSSI, one general node wanting to estimate its location should at least cooperate with three anchor nodes so as to calculate the location by trilateration. Moreover, by utilizing the location information at different times, the general node can easily calculate its moving direction information. For example, as seen in Fig. 2, the location of node X at time T_1 and T_2 are (x_1, y_1) and x_2, y_2, respectively. Then, the moving direction of node X is $\theta = arctan\frac{y_2-y_1}{x_2-x_1}$. Using the location information, node X can further calculate its moving speed by the following equation: $S_X = \frac{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}}{T_2-T_1}$. Observed from Fig. 2, we define that the message transmission direction is α. If $|\theta - \alpha| \leq 45°$, then we think the node's moving direction is the same as the message's transmission direction.

Every node stores its own location and moving direction information. In order to minimize the communication overhead, all the information will not be exchanged with each other unless they are required from other nodes. Moreover, when an anchor node is situated in the transmission range of a certain general node, the information of this general node can be stored in this anchor node. For a simple description, we can also say that this anchor node lists this general node. After some time interval, the anchor node should update its list so as to re-obtain the latest location and moving direction information of the general nodes, whose radio

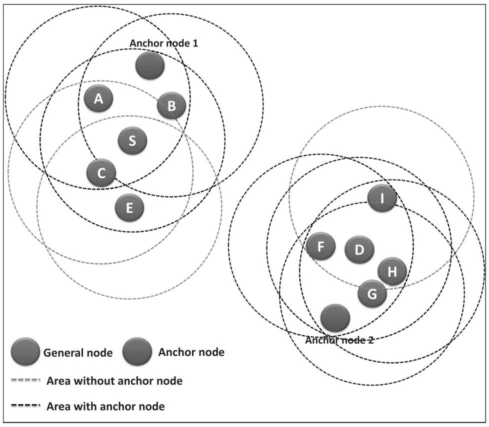


Fig. 3 An example of achieving location information of the destination node.

transmission ranges cover this anchor node.

In order to understand the information achieving operation of the LDPR protocol, consider the following scenario depicted in Fig. 3. We assume that there is a message wanting to be transmitted from the source (S) to the destination (D). If there is an anchor node in the transmission range of node S, S can request the anchor node to find the destination's location. Otherwise, node S will wait until an anchor node appears in its transmission range. After this anchor node broadcasts the ID of the destination node, all the other anchor nodes will check their lists to find the destination node. If one anchor node finds the destination node, the source can obtain the information about it. As shown in Fig. 3, anchor node *1* is in the transmission range of nodes A, B and S. Hence, the location information of nodes A, B and S can be stored in this anchor node. We can also say anchor node *1* lists nodes A, B and S. In a similar way, anchor node *2* lists nodes F, D, H, and G. Therefore, source node S can request anchor node *1* to check its list or to broadcast the request to check other anchor nodes whether or not including the destination node D. Anchor node *2* transmits the location information to anchor node *1* after receiving the request and checking its list. Source node S, finally, obtains the location information about node D by relaying on anchor node *1*. In the worst case, if there is no anchor node in the transmission range of destination node D, then D will wait until it can be listed in a certain anchor node due to all the general nodes and the anchor nodes being mobile.

In LDPR, all of the messages wanting to be transferred must be attached with the priority information, which should be set based on the following factors:

- The validity of the message.
- The security of the message.
- Transmission speed request.
- The value of the information.
- The cost of the message.
- The distance to the destination and the direction to the destination.

The arranging sequence of these factors is abided by the priority level.

LDPR consists of two schemes: a transmission scheme that is enabled to transmit messages in compliance with their priority, and a drop scheme for managing and utilizing the buffer space. Each of them is described below.

3.2 Transmission Scheme

All the messages must be arranged in the buffer space of the nodes according to their priority. The message which has the highest priority will be arranged at the top of the buffer space, at the same time, it will be first transmitted if the best next hop is determined.

At the beginning of transmission, all the information of the destination should be known by the source. Now we suppose that there is a message wanting to be transmitted from the source node (S) to the destination node (D). Thus, the location and moving direction information of the destination node should be obtained first. The process how to obtain this information is described in Sect. 3.1.

After node S gets the information of destination D, the second step is to determine the best next hop for the transmission. Each general node can obtain all its neighbor nodes' location and moving direction information by directly communicating with its neighbors. In the beginning, node S broadcasts a "destination location" request. As shown in Fig. 4. If node D is in the transmission range of S, then node D replies to S before node S directly transmits the message to node D. Otherwise, if node D is not in this range, then no node replies to S. After that, node S broadcasts the "moving direction" request to all its neighbors, which is illustrated in Fig. 5.

It is important to note that the decision of message's transmission depends on whether the moving direction of the next-hop node is the same as the message's transmission direction. After receiving the "moving direction" request from node S, as seen in Fig. 6, each node in the transmission range of node S needs to compare its moving direction with the message's transmission direction. Only the node possessing the same moving direction as the message's transmission direction can reply to node S. Note here that this node is referred to as the best next-hop node. The location, moving direction and moving speed information of the best next-hop node should be included in the reply message so that node S can estimate whether the moving direction of the best next-hop node is indeed the same as the message's transmission direction. After receiving the reply from the best next-hop node, node S can determine the message's transmission and transmit the message to the best next hop. In Fig. 6, we assume that node A's moving direction is the same as the message's transmission direction. Hence, only node A replies to node S when A receives the "moving direction" request from node S. Then, the message is transmitted to node A immediately. That is to say, A becomes the best next hop. In another situation depicted in Fig. 7, there are two nodes (A and C) both having the same moving direction as the message's transmission direction, then both of

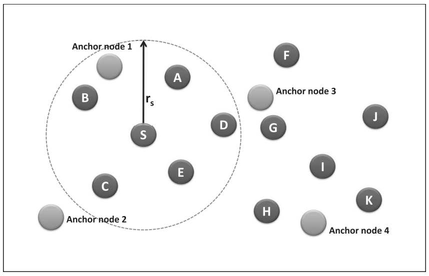


Fig. 4 Node S broadcasts "destination location" request.

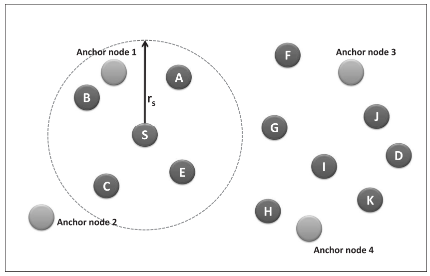


Fig. 5 Node S broadcasts "moving direction" request.

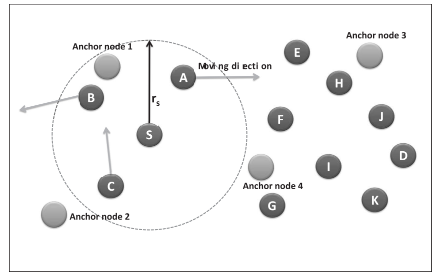


Fig. 6 Node A's moving direction is same as the message's transmission direction.

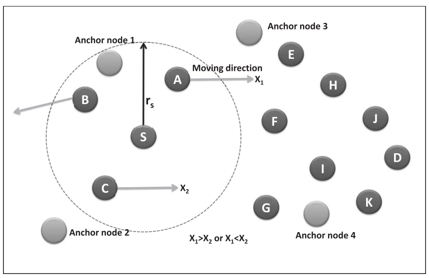


Fig. 7 Nodes A and C both have the same moving direction with the message's transmission direction.

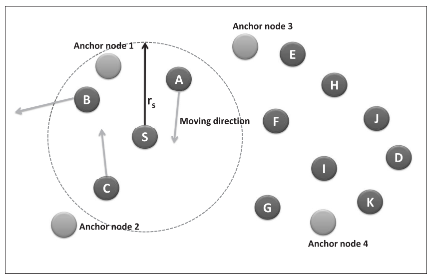


Fig. 8 All the nodes in the transmission range of nodes S are different from the message's transmission direction.

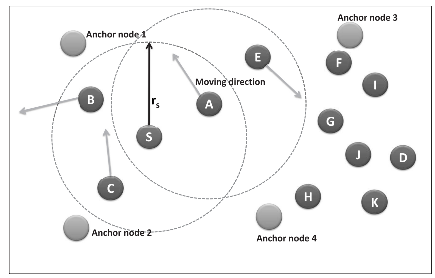


Fig. 9 Node E's moving direction is same as the message's transmission direction.

them need to reply to S. Furthermore, the moving speeds of nodes A and C are compared by node S in order to determine which node should accept and relay the message. The node with the highest speed can only become the best next hop. Therefore, noted from Fig. 7, node A will become the best next hop and the message will be transmitted from S to A since node A's speed is faster than node C's.

In the worst case, all the moving directions of the nodes in the transmission range of node S are different from the message's transmission direction, then no node replies to node S. Therefore, node S will wait until it can find some node whose moving direction is the same as the message's transmission direction. This scenario is explained in Fig. 8.

However, there exists one problem by using this transmission scheme, which is depicted in Fig. 9. Even though all the moving directions of the next-hop nodes are different from the message transmission direction, the message may still be delivered to the destination successfully. According to Fig. 9, nodes A, B, and C are the neighbor nodes of node S since all of them are in the transmission range of node S, while node E is not the neighbor of node S but the neighbor of node A. On the other hand, all the moving directions of nodes A, B, and C are different from the message's transmission direction while the moving direction of node E is the same as the message's transmission direction. Applying the transmission scheme of LDPR, no node can reply to node S. Therefore, node S should wait until it can find a neighbor node whose moving direction is the same as the message's transmission direction. However, if the message can be transmitted from node S to A first and further transmitted from A to E, then it may still be delivered to the destination successfully. Nevertheless, this situation increases the complexity of routing, where the transmitting node needs to collect all the information of locations and moving directions from the two-hop away nodes. Hence, in the presented protocol, for simply solving this problem, we only consider the message delivery with respect to the moving directions information of neighbor nodes only one-hop away from the transmitting node. It is referred to as one-hop situation.

Finally, the best next hop can be decided by this transmission scheme. Thus, the message is able to be delivered to and stored in this intermediate node, which will continue to determine the best next hop by the same transmission scheme until the message successfully arrives at the destination node.

3.3 Drop Scheme

Once the source node determines the first next hop to the destination, it will transmit the message to this first intermediate node. However, is there available buffer space in this node or not? How can we optimize the buffer space? In LDPR, the drop scheme solves these problems.

For explaining the drop scheme in detail, a clear example is given in Fig. 10. We assume that node S is the source and node D is the destination. Node A is supposed to be the best next hop determined by the transmission scheme described above. Nodes B, C, E, and F are in the transmission range of node A but not in the range of node S. There is a message which intends to be transmitted from node S to node A.

First, node S sends a "transmission" request to node A. After receiving the request, node A checks its buffer space to determine whether or not buffer space is available. Node A

```
1: If (p_new < all of the p_old)    {A sends the "refuse" reply to S;      Return;}
2: // A refuses this transmission and wait until A's buffer is available again.
3: If (p_new > some of the p_old)    // one message with the lowest priority in node A is dropped or allocated to other node.
4: {
5:          A broadcasts the "buffer available" request to all its neighbors;
6:          If ( there is one neighbor node X whose buffer space is available and the available size is more than 1/2 of its total size)
7:          {
8:                    Node X sends reply to node A;
9:                    The message with the lowest priority in node A is sent to node X;
10:                   Node A sends "permit" reply to node S;
11:                   The new message is sent from node S to node A;      Return;
12:         }
13:         If ( there are more than one neighbor node whose buffer space is available and the available size is more than 1/2 of its total size)
14:         // We assume that  node B and node E satisfy this condition.
15:         {
16:                   Node A broadcasts the "moving direction" request to them;
17:                   If (there is one node Y whose moving direction is same with the message's direction)
18:                   // We assume that node E satisfy this condition.
19:                   {
20:                             Node Y sends reply to node A;
21:                             The message with the lowest priority in node A is sent to node Y;
22:                             Node A sends "permit" reply to node S;
23:                             The new message is sent from node S to node A;      Return;
24:                   }
25:                   If (all of them have the same moving direction with the message's direction)
26:                   {
27:                             The message with the lowest priority in node A is sent to one of them randomly;
28:                             Node A sends "permit" reply to node S;
29:                             The new message is sent from node S to node A;      Return;
30:                   }
31:         }
32:         If (all the neighbors' buffer spaces are not available or the available buffer space are all less than 1/2 of buffer space)
33          {
34:                   The message with the lowest priority in node A will be dropped;
35:                   Node A sends "permit" reply to node S;
36:                   The new message is sent from node S to node A;      Return;
37:         }
38: }
```

Fig. 11 Priority comparing algorithm in the drop scheme.

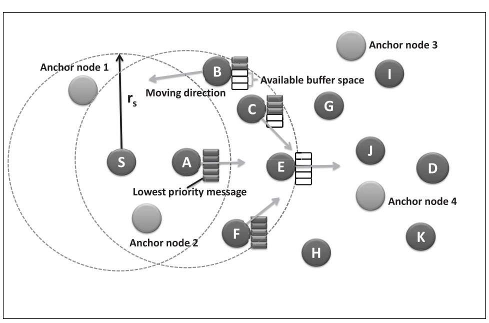


Fig. 10 An example in the drop scheme.

replies to node S and permits the transmission only if node A's buffer space is not full. Then node S sends the new message to node A. However, if node A's buffer space is full, node A still replies to node S and only permits to accept priority information of this message sent from node S. Then node A compares this priority information with others which are already stored in node A's buffer space.

Figure 11 explains the detailed steps of the priority comparing algorithm in drop scheme. Here, p_{new} represents the priority of the new message needed to be delivered to the destination; on the contrary, p_{old} delegates the priority of the old message stored in node A. If p_{new} is lower than all the p_{old}s, then node A will refuse this new message to be transmitted from node S to node A or permit it until node A's buffer is available again. In this condition, hence, node A sends the "refuse" reply to node S (line 1-2 in Fig. 11). However, if p_{new} is higher than some of the p_{old}s, then one message with the lowest priority in node A will be dropped or be allocated to other available buffer space in other nodes in order to make space for storing this new message (line 3). First, node A broadcasts the "buffer available" request to all its neighbors to judge which node's buffer space is available. If there is one neighbor node whose buffer space is available and the available size is more than $1/2$ of its total size, then this node replies to node A and allows accepting the lowest priority message sent from node A. In this case, the new message can be transferred from node S to node A successfully (line 5-12). In our example according to Fig. 10, we suppose that nodes B and C's buffers are available. Yet, only B's available size is more than $1/2$ of its total size. In addition, node E has an empty buffer space that can be completely available. To the contrary, node F's buffer is completely not available, namely the buffer of node F is full. Hence, nodes B and E reply to node A, while nodes F and C keep silent. This is easy to be explained because only node B or E having enough available buffer space can accept this lowest priority message. Secondly, node A broadcasts the "moving direction" request to nodes B and E. We assume that node B's moving direction is the opposite of the message's direction while node E's is the same as the message's direction. Therefore, indubitability, node E replies to node A and permits to accept the lowest priority message sent from node A. In this case, node A can make room for the new message from node S. Finally, node A replies to S and permits the new message to be delivered (line 13-24). Of course, there exists another case. When nodes B and E have the same moving direction as the message's direction, the message with the lowest priority in node A will be randomly sent to one of them (line 25-31).

In the worst case, if all the neighbors' buffer spaces are not available or their available buffer spaces are all less than $1/2$ of their buffer spaces, then the message with the lowest priority in node A will be dropped so that node A can accept the new message (line 32-37).

After the message is successfully transmitted from node S to node A, according to all the messages' priority, node A continues deciding the best next hop of the highest priority message until this message arrives at the destination by using the same transmission and drop scheme in LDPR as described above.

4. Performance Evaluation

4.1 Simulation Environment

We implemented LDPR by using the ns-2 simulator. The version of ns-2 used in our simulation is ns-2.33. The implementation of our proposed routing protocol is based on the Monarch [11] extensions to ns-2. Monarch extends ns with radio propagation that models signal capture and collision. The simulator also models node mobility, allowing for experimentation with ad hoc routing protocols that must cope with frequently changing network topology. Finally, Monarch implements the IEEE 802.11 [12] Medium Access Control (MAC) protocol.

Unless otherwise noted, our simulations are run with the following parameters. We model 20, 50, 100 and 150 mobile nodes (including 10% anchor nodes) moving in a square area of 1000 m x 1000 m. Each node picks a random spot in the square and moves there with a speed uniformly distributed between 0~5 meters/sec. The radio transmission range is assumed to be from 10 to 250 meters and a two-ray ground reflection propagation channel is considered. The buffer size varies from 10 to 1000. The parameters for the simulation are given in Table 3 in detail. Most other parameters use ns-2 defaults. Nodes are generated randomly in an area and move according to the well-known Random waypoint mobility model.

Table 3 Parameters used in the simulation.

Parameter	Value
Number of node	20, 50, 100, 150
Mobility model	Random way point
Mac	IEEE 802.11 DCF
Traffic source	CBR for UDP-based traffic
Node speed	0 ~ 5 m/s
Propagation model	Two-ray ground reflection
Simulation time	1000 seconds
Data transmission rate	2 Mbps
Radio transmission range	10, 25, 50, 100, 250 meters
Buffer size	10, 50, 100, 500, 1000
Pause time	0, 20, 50, 100, 300, 600, 900 seconds
Packet outgoing rate	1, 2, 4, 8, 16 packets/sec
Number of sessions	2, 6, 10, 14, 18

Due to space restrictions, we have focused on comparing the performance of the protocols with regards to the following metrics. First of all, we are interested in the packet delivery ratio, i.e. how many packets are delivered to the destination. The definition of packet delivery ratio is given in Eq. (1).

$$\textit{Packet delivery ratio} = \frac{\textit{Number of delivered packets}}{\textit{Number of generated packets}} \tag{1}$$

Second, we study the normalized routing overhead of the whole network. This indicates the system resource utilization and consumption. The equation of normalized routing overhead is described in Eq. (2).

$$\textit{Normalized routing overhead} = \frac{\textit{Number of routing packet transmission}}{\textit{Number of data packet transmission}} \tag{2}$$

Finally, even though the applications assumed in this paper are relatively delay-tolerant, it is still of interest to consider the average end-to-end delay of packet delivery to find out how much time it takes for a message to be delivered. The calculation of average end-to-end delay is shown in Eq. (3).

$$\begin{aligned}&\textit{Average end-to-end delay}\\ &\quad = \textit{average value of}\\ &\qquad (\textit{delivered packet's timestamp} -\\ &\qquad \textit{generated packet's timestamp})\end{aligned} \tag{3}$$

4.2 Result and Discussion

Our simulation includes two parts. First of all, we present a comparative simulation analysis among the proposed LDPR, EPI [2], PREP [3] and DHR [14] with respect to

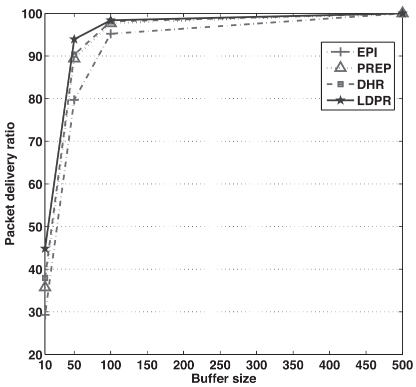


Fig. 12 Packet delivery ratio versus buffer size.

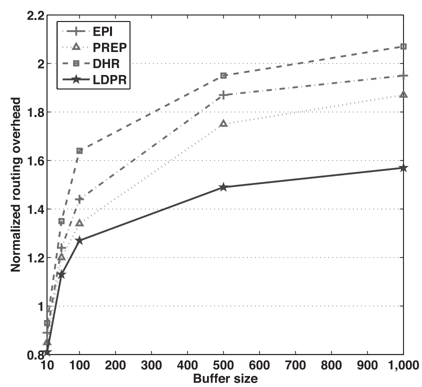


Fig. 13 Normalized routing overhead versus buffer size.

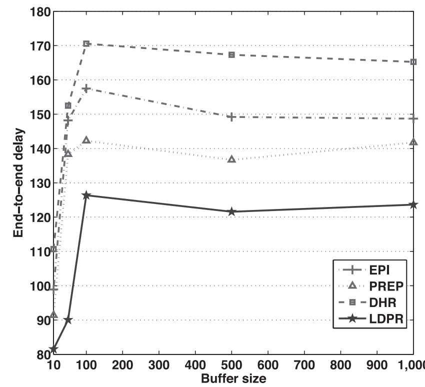


Fig. 14 End-to-end delay versus buffer size.

packet delivery ratio, normalized routing overhead and average end-to-end delay, where EPI is the earliest and well-known routing protocol in DTNs, PREP is the best routing protocol in the flooding family, and DHR is the most outstanding routing protocol in the forwarding family. The number of nodes is set to be 50 and some other simulation parameters such as pause time, packet outing rate and number of sessions are only assumed to be 100 sec., 4 packets/sec. and 6, respectively. We concentrate on comparing LDPR with the other three routing protocols in terms of different buffer size and radio transmission range. The buffer size is changed from 10 to 1000 and the radio transmission range is considered from 10 to 250 meters. We do simulations for each scenario. While the buffer size is varied during a simulation, the radio transmission range is fixed as 50 m. Conversely, the buffer size is set as 500 when radio propagation is changed.

In the second part, we only focus on reporting the comparative simulation results about packet delivery ratio of LDPR itself with respect to different node density and distinct buffer size. The number of nodes varies from 20 to 150 and the range of variation of buffer size is the same as in part one. The default value of the number of nodes and the buffer size are 50 and 50. At the same time, the pause time, the packet outgoing rate (transmission rate), and the number of sessions are varied in a meaningful range (i.e., the pause time from 0 to 900 sec., the packet outgoing rate from 1 to 16 packets/sec., and the number of sessions from 2 to 18 are applied). While one simulation factor is varied during a simulation, the others are fixed as follows: the pause time is 100 sec., the packet outgoing rate is 4 packets/sec., and the number of sessions is 6.

4.2.1 Comparison with EPI, PREP and DHR

As we know, EPI is one of the earliest proposals for routing in delay tolerant networks, which likes the pattern of pandemic virus transmitting [2]. Simultaneously, it is important to note that PREP [3] and DHR [14] are the best routing protocols in the flooding family and forwarding family, respectively. The key idea of PREP is to impose a partial ordering on the message called bundle based on the delivery cost (the number of hops) to destination and source, and expiration time. In DHR, the network topology information is utilized to effectively select the best path. Then, the message is forwarded from node to node along this path. In this subsection, we analyze the influence of buffer size and radio transmission range on packet delivery ratio, normalized routing overhead and average end-to-end delay among LDPR, EPI, PREP and DHR.

First, we choose buffer size as the parameters of X axis and consider the packet delivery ratio, normalized routing overhead and average end-to-end delay as the parameter of Y axis, respectively. For observing the impact of the performance of the protocols caused by the change of buffer size, we set the radio transmission range at 50 meters without thinking about other circumstances. Figures 12, 13, and 14 show the details of the comparison results. The first interesting aspect that we analyze is the packet delivery ratio, a characteristic aspect of a protocol for delay tolerant networks. As shown in Fig. 12, the packet delivery ratio increases as the buffer size increases. All the curves sharply rise up when the buffer size is less than 100 and then gradually go up until they reach 100%. This is intuitive, since a larger buffer size means that there is enough space in a node to store large numbers of packets so as to guarantee the packets' lifetime and delivery. Seen from Fig. 12, LDPR, obviously, outperforms the other three routing protocols. It is worth noting that the packet delivery ratio of LDPR is the largest when the buffer size is smaller than 500 and can reach 100% as the buffer size over 500. In particular, the packet delivery ratio of LDPR performs well with a very small buffer size. When the buffer size is only 50, the packet delivery ratio of LDPR can still be 93.9% which can satisfy general communication demand. In this condition, however, the packet delivery ratio of EPI, PREP and DHR are only 79.7%, 89.4% and 90.32%, respectively. In the extreme situation where the buffer size is only 10, all the protocols fail because the delivery ratios are below 50%. Comparing them, it is easy to say that LDPR is more non-sensitive to buffer size than the other ones. This is due to the fact that LDPR effectively utilize and manage the buffer

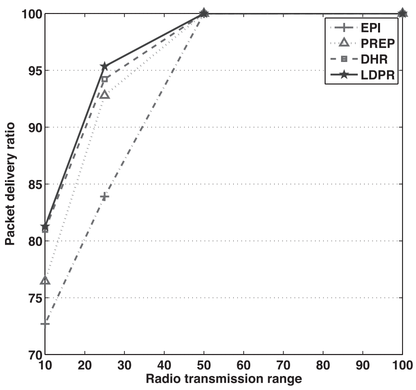


Fig. 15 Packet delivery ratio versus radio transmission range.

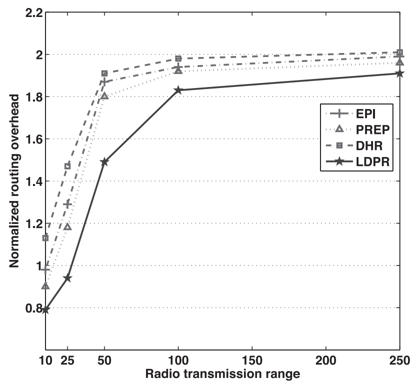


Fig. 16 Normalized routing overhead versus radio transmission range.

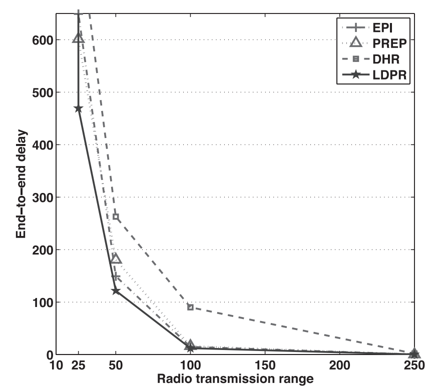


Fig. 17 End-to-end delay versus radio transmission range.

size with the drop scheme. Another critical aspect we investigated is the normalized routing overhead, which indicates the system resource utilization and consumption. It is an important criterion to estimate the performance of routing protocols. Figure 13 describes the change of normalized routing overhead of all routing protocols with the increase in buffer size. As expected, LDPR exhibits its advantages in terms of normalized routing overhead compared with the other protocols. In LDPR, every node is able to store its own location and moving direction information. In order to minimize the communication overhead, all the information will not be exchanged with each other unless they are required from other nodes. As depicted in Fig. 13, for each metric, the normalized routing overhead of LDPR is lower than that of the three others. The last aspect we analyzed is the average end-to-end delay. Delay is inevitable in DTNs, however, it is still of interest to find out how much time it takes for a message to be delivered. In fact, it makes sense to compare the routing protocols with regards to average end-to-end delay. In other words, it reflects on the difference of delivery time in order to make a better choice in the complicated environment of real applications. Let us observe the result reported in Fig. 14. LDPR has lower average end-to-end delay in all cases of buffer size compared with the others. It is noteworthy that the average end-to-end delay of all the protocols hastily increase until the buffer size is 100, and then decline a little as the buffer size further increases. This is possible due to the buffer size playing an important role. When the buffer size is very small, the successful packet deliveries are limited in one hop or two hops. Any packet wanting to be transmitted to some farther destination could be dropped due to the small buffer size of the intermediate nodes. Yet, more and more packets can be successfully delivered to their destinations with an increase in buffer size. Simultaneously, the average end-to-end delay sharply increases. When the buffer size is over 100, there is enough space to store and manage packets so that the change in the delay is not intense.

Second, we choose radio transmission range as the parameters of X axis in order to compare the performance of LDPR, EPI, PREP and DHR with regards to packet delivery ratio, normalized routing overhead and average end-to-end delay. In this case, we fix the buffer size as 500 so as to observe the impact of different radio transmission ranges. Figures 15, 16, and 17 show the details of the comparison result. Observed from Fig. 15, it is easy to find that all the routing protocols have 100% packet delivery ratio when the radio transmission range is larger than 50 m. On the other hand, in the extreme conditions where the radio transmission range is only 10 m, LDPR still outperforms the other three protocols. It indicates that LDPR has enough ability to adapt to a complex environment of real applications. When the radio transmission range increases to 25 meters, the packet delivery ratio of LDPR is 95.4% which can still ensure successful packet delivery. Seen from Fig. 16, we note that LDPR has lower normalized routing overhead than the three others at all the metrics. The shapes of these curves reveal that the normalized routing overhead increases with the increment in radio transmission range. Figure 17 depicts the comparison result of average end-to-end delay among the routing protocols. In particular, the average end-to-end delay of LDPR reaches 469 sec when the radio transmission range is 25 meters, while that of EPI, PREP and DHR come up to 648 sec, 601 sec and 810 sec. In the worst case, when the radio transmission range is only 10 m, then the average end-to-end delay of the entire protocols trend to be infinite. We note that, in LDPR, nodes require enough time to obtain the information of location and moving direction. Moreover, all the nodes move randomly across this area. It dooms that the nodes need to usually gather and update the information in a certain time interval. LDPR has the lowest average end-to-end delay among the four routing protocols determined from the curves. However, the delay is still very large in all the metrics. The feature of packet delivery delay in LDPR determines that LDPR can be implemented in an environment which focuses on the reliability of the message transmission rather than delivery delay.

4.2.2 Effect of Node Density

We now only analyze the influence of the node density chosen in the study. We consider four different numbers of

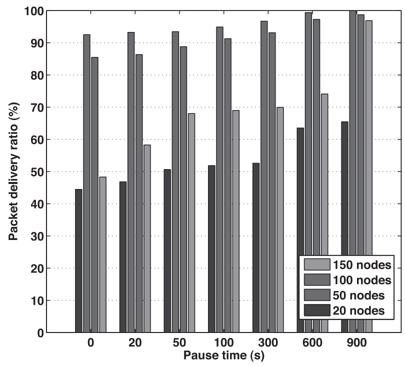


Fig. 18 Packet delivery ratio versus pause time under different nodes density.

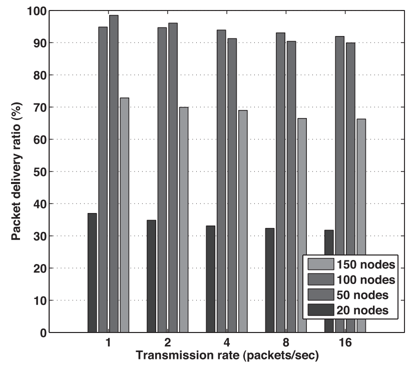


Fig. 19 Packet delivery ratio versus transmission rate under different nodes density.

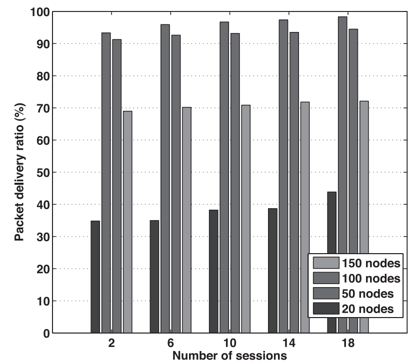


Fig. 20 Packet delivery ratio versus number of session under different nodes density.

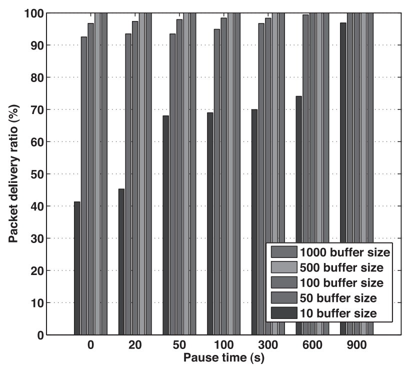


Fig. 21 Packet delivery ratio versus pause time under different buffer size.

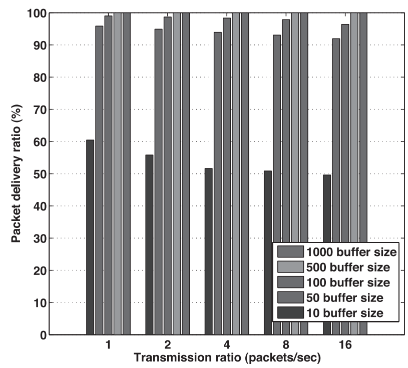


Fig. 22 Packet delivery ratio versus transmission rate under different buffer size.

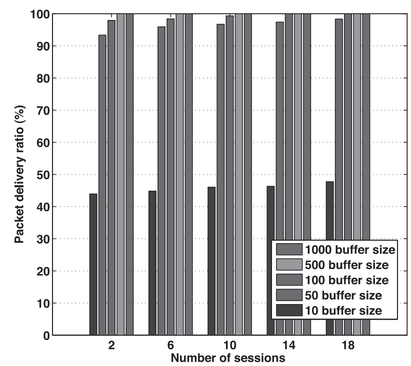


Fig. 23 Packet delivery ratio versus number of session under different buffer size.

nodes in the square 1000 m x 1000 m, where there are 20 nodes, 50 nodes, 100 nodes and 150 nodes, respectively. We choose the parameters of *X* axis with respect to pause time, transmission rate and number of sessions. And we only consider the packet delivery ratio as the parameter of *Y* axis. Figures 18, 19 and 20 show the influence of the node density on the packet delivery ratio of LDPR itself.

Viewed from Fig. 18, at first, we note that the packet delivery ratio gradually increases as the pause time rises. This is intuitive, since a larger pause time means that nodes are more close to static and the networks are more stable. Note here that the key point is to observe the influence of node density. In general, a higher node density leads to a higher packet delivery ratio. In LDPR, for each metric, the packet delivery ratio in the 20-node network is lower than that in the other networks with higher node density. In addition, it is worth noting that the packet delivery ratio of LDPR is the highest when the number of nodes reaches 50. If the number of nodes exceeds 50, the packet delivery ratio can still be maintained at a high level. But, by further increasing the number of nodes, the packet delivery ratio decreases. That's because, in LDPR, all the nodes delivering messages depend on the information of nodes' location and moving direction. It is complex and difficult to deal with these location and moving direction information when the node density is quite high. At the same time, it is possible that one node may not obtain accurate information resulting in packet delivery failure. To summarize, these experiments show that LDPR is not able to guarantee good performance when the number of nodes is large. It states that LDPR is not good at scalability.

Figures 19 and 20 represent the same phenomenon under various transmission rates and different number of sessions. Of course, the packet delivery ratio with respect to 150 nodes is higher than that with regards to 20 nodes. From the comparison results, in most cases, a node density of 50 is the best choice and outperforms other node densities. It means that we should choose a suitable node density rather than the highest node density in real applications in order to achieve the best performance.

4.2.3 Effect of Buffer Size

Another aspect we observed is that the influence of buffer size on the packet delivery ratio. We think over five situations where the nodes' buffer sizes are 10, 50, 100, 500, and 1000. We also choose pause time, transmission rate and number of sessions as the parameters of *X* axis, and only consider the packet delivery ratio as the parameter of *Y* axis. Figures 21, 22 and 23 illustrate the influence of the buffer size on the packet delivery ratio based on pause time, transmission rate, and number of sessions, respectively.

In general, the influence of buffer size is evident on the packet delivery ratio. The change in the packet delivery ratio is notable with an increase in buffer size. In particular, the packet delivery ratio of LDPR exceeds 90% when the buffer size is larger than 50 in all the metrics. Moreover, the packet delivery ratio can reach 100% when the buffer size is over 500 in all the metrics. Only if the buffer size equals 10, the packet delivery ratio has a significant decline. Figures 21, 22 and 23 reveal that buffer size is an important factor in DTNs and always has a heavy impact on the performance of the routing protocols in DTNs. Specially, in LDPR, buffer size can be more efficiently utilized and managed. We can achieve a high packet delivery ratio with a reasonable buffer size.

To summarize, these simulations state that LDPR is able to guarantee good performance in the presence of normal buffer size (around 50). LDPR doesn't need infinite buffer size to ensure the packet delivery ratio as in EPI. According to real environment and real resource consumption, we can decide the suitable buffer size in order to obtain good performance by utilizing LDPR, in comparison to the other protocols taken into consideration.

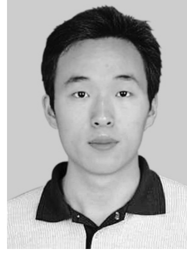
5. Conclusions

In this paper, we have investigated delay tolerant networks, where a lot of new applications are viable. It is important to note that an exciting future could be exhibited if the underlying mechanisms are presented. Therefore, we have proposed a routing protocol named Location and Direction Aware Priority Routing (LDPR) for DTNs, which utilizes the location and moving direction of nodes to deliver a message from source to destination. We have shown that a node can get the location and moving direction of itself by receiving beacon packets periodically from anchor nodes and referring to received signal strength indicator (RSSI) for the beacon. LDPR contains two schemes named transmission scheme and drop scheme, which take advantage of the nodes' information of the location and moving direction to transmit the message and store the message into buffer space, respectively.

The simulation experiments have shown that LDPR is able to ensure the validity and reliability of the message transmission and the buffer size in LDPR can be efficiently utilized and managed so as to achieve a high packet delivery ratio. Moreover, LDPR is able to guarantee good performance with a lower routing overhead in the presence of suitable buffer size. In particular, the packet delivery ratio can still be more than 90% in some extreme environment where the buffer size is only 50 or the radio transmission range is only 25 meters. However, LDPR is not good at scalability and can only be implemented in environments which focus on the reliability of the message transmission rather than delivery delay. Our future work is to overcome these problems to design a more robust routing protocol for harsh operational environments.

References

[1] K. Fall, "A delay-tolerant network architecture for challenged internets," Proc. Annual Conf. of the Special Interest Group on Data Communication (ACM SIGCOMM'03), pp.27–34, Aug. 2003.
[2] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Technical Report, CS-200006, Duke University, April 2000.
[3] R. Ramanathan, R. Hansen, P. Basu, R.R. Hain, and R. Krishnan, "Prioritized epidemic routing for opportunistic networks," Proc. ACM MobiSys Workshop on Mobile Opportunistic Networks (MobiOpp 2007), June 2007.
[4] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," ACM SIGMOBILE Mobile Computing and Communications Review, vol.7, no.3, pp.19–20, July 2003.
[5] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," Proc. ACM SIGCOMM, 2004.
[6] E.P.C. Jones and P.A.S. Ward, "Routing strategies for delay-tolerant networks," Submitted to Computer Communication Review (under review), 2008.
[7] C.E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," Computer Communication Review, pp.234–244, Oct. 1994.
[8] J. Broth, D.B. Johnson, and D.A. Maltz, "The dynamic source routing protocol for mobile ad hoc networks," Internet Draft, 1998.
[9] C.E. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," IEEE Workshop on Mobile Computing Systems and Applications, pp.99–100, Feb. 1999.
[10] J. Shen, S.M. Moh, and I.Y. Chung, "Routing protocols in delay tolerant networks: A comparative survey," 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), pp.1577–1580, July 2008.
[11] CMU Monarch Project. The CMU Monarch Project's wireless and mobility extensions to ns. ftp.monarch.cs.cmu.edu/pub/monar-ch/wireless-sim/ns-cmu.ps, Aug. 1999.
[12] IEEE Computer Society, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," 1997.
[13] E.P.C. Jones, L. Li, and P.A.S. Ward, "Practical routing in delay-tolerant networks," Proc. ACM SIGCOMM Workshop on Delay-tolerant Networking, pp.237–243, Sept. 2005.
[14] C. Liu and J. Wu, "Scalable routing in delay tolerant networks," Proc. MobiHoc'07, pp.51–60, Sept. 2007.
[15] Received signal strength indication, http://en.wikipedia.org/wi-ki/Received_Signal_Strength_Indication
[16] K. Fall, "A delay-tolerant network architecture for challenged internets," Intel Research Technical Report, IRB-TR-03-003, Feb. 2003.



Jian Shen received the B.E. degree from Nanjing University of Information Science and Technology, Nanjing, China, in 2007 and the M.E. degree in Computer Science from Chosun University, Gwangju, Korea, in 2009. Since 2009, he is working toward the Ph.D degree in Computer Science from Chosun University, Gwangju, Korea. His research interests include computer networking, security systems, mobile computing and networking, ad hoc networks and systems, and ubiquitous sensor networks.

Sangman Moh received the Ph.D. degree in computer engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea in 2002. Since late 2002, he has been a faculty member in the School of Computer Engineering at Chosun University, Gwangju, Korea. From 2006 to 2007, he was on leave at Cleveland State University, Cleveland, Ohio, USA. Until 2002, he had been with Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea, where he served as a project leader, since he received the M.S. degree in computer science from Yonsei University, Seoul, Korea in 1991. He received the national qualification for the registered Professional Engineer (PE) in information technology from the Korean Government in 1993. He is listed in Who's Who in the World, 20th edition and Who's Who in Science and Engineering, 8th edition, published by Marquis Who's Who, New Providence, New Jersey, USA in 2003 and 2005, respectively. He is also listed in 2000 Outstanding Intellectuals of the 21th Century published by International Biographical Center (IBC), Cambridge, England, UK in 2004. His research interests include mobile computing and networking, ad hoc networks and systems, ubiquitous sensor networks, network based computing, parallel and distributed computing, and high-performance computer systems. He has three international book chapters and published more than 100 papers in international and domestic journals and conference proceedings, and has held more than 40 overseas and domestic patents. He serves on the program committees of international conferences and workshops in his areas of interest. Dr. Moh is a member of the IEEE, the ACM, the KIISE, the IEEK, the KIPS, the KICS, the KMMS, and the KPEA.

Ilyong Chung received the B.E. degree from Hanyang University, Seoul, Korea, in 1983 and the M.S. and Ph.D. degrees in Computer Science from City University of New York, in 1987 and 1991, respectively. From 1991 to 1994, he was a senior technical staff of Electronic and Telecommunication Research Institute (ETRI), Dajeon, Korea. Since 1994, he has been a Professor in Department of Computer Science, Gwangju, Korea. His research interests are in computer networking, security systems and coding theory.