

## DTNs Routing Protocol Based on Urban Public Transport System

Xuebin Ma, Xin Zhan, Zhenchao Ouyang, Jing Bai

Department of Computer Sciences  
Inner Mongolia University  
Hohhot, China

{maxuebin, zhanxin0223, ouyangkid, bayjing1218}@gmail.com

**Abstract**—Urban public transport networks are special types of Delay Tolerant Networks (DTNs) which consist of buses and other forms of transport. It is hard to establish an end-to-end link and maintain a stable network topology in this kind of networks. However, unlike other DTNs, the bus's mobility in public transport networks is regular, which means the bus's line is determined. Thus, traditional routing protocols are no longer fit. In this paper, we search and analyze many real scenarios and present the Bus Line-based Routing (BLR), and our BLR is based on the minimum distance between bus lines. We simulate the performance of BLR in ONE simulator, and the result shows that our routing protocol has better performance than traditional routing algorithms in urban scenarios.

**Keywords**- DTNs; public transport system; routing protocol; ONE.

### I. INTRODUCTION

DTNs are a kind of challenged networks, which is proposed by Kevin Fall<sup>[1]</sup>. DTNs use "store-carry-forward" routing mechanism for transferring messages. DTNs don't have complete end-to-end links and stable network topology, and this kind of networks has these characteristics, such as intermittent connection, high-latency and high error rate. DTNs are mainly used in where communications infrastructure is lacked, the density of nodes is sparse and nodes have irregular movements. DTNs can be applied in many fields. Wildlife tracking system<sup>[2]</sup> collected and monitored the information about white-tailed deer and other animals through the sensor nodes. ZebraNet<sup>[3]</sup> was designed by Princeton University and used to track the African grasslands zebra migration. Massachusetts Institute of Technology developed CarTel<sup>[4]</sup>, which is an information collection and dissemination system that based on vehicle sensor. This system can be used for environmental monitoring, traffic reports, route navigation and accident warning. DakNet<sup>[5]</sup> provided a very low cost of digital communication and helped the villagers who live far apart to exchange information, which had been deployed in remote areas of India and Cambodia.

There are some classic routing algorithms in DTNs, such as Direct Delivery Routing, Epidemic Routing, Prophet Routing, Spray and Wait Routing<sup>[6-10]</sup>. These algorithms are mainly based on flooding or node history information, their

design principles are relatively simple. But the efficiency of these algorithms is low and the applicable scope is limited. For example, Epidemic Routing<sup>[7]</sup> forwards messages through flooding, and this can consume large amounts of network resources, such as buffer, bandwidth and battery energy. In Spray and Wait Routing<sup>[9]</sup>, a node sprays a number of copies into the network, and then waits till one of these nodes meets the destination. This way can reduce copies and resource consumption. Prophet Routing<sup>[8]</sup> can successfully deliver a message by exploring its history information. It only replicates a message if the delivery probability exceeds a threshold when transferring messages. Prophet can reduce the copies and increase the transmission success rate.

An emerging field for development of DTNs is urban public transport network. Buses, light rail vehicles, subways and other forms of transport have already covered every area of the city. We can thus use this kind of networks to route information and provide a specific service, for example, weather forecasting, traffic information sharing, and advertisement service.

In this paper, we study the construction of a DTN on the public transport network, because the buses move regularly, and they have a fixed route for traveling. According to these characteristics, we present BLR (Bus Line-based Routing) and carry out a simulation. Experimental results show that BLR is superior to the traditional algorithm in terms of the delivery rate, delay time and copies.

We first mainly present some previous works related with routing protocols (section II), we then establish a network model and introduce our new algorithm (section III). After that, we test and comment the performances (section IV).

### II. STATE OF THE ART

In public transport networks, the idea of forwarding messages, based on bus lines and minimum distance, is used to increase the delivery rate, reduce message latency and copies. In this section, we briefly describe some existing routing algorithms and the way they work.

#### A. Link state-based routing protocol

Optimized Link State Routing Protocol (OLSR)<sup>[11]</sup> is a link-state protocol. Nodes periodically exchange topology

information with each other in the network. This protocol uses multipoint relays (MPRs) <sup>[12]</sup> to control the number of messages. Delay Tolerant Link State Routing (DTLSR) <sup>[17]</sup> references the OSPF <sup>[18]</sup> and minimizing the estimated expected delay (MEED) <sup>[19]</sup>, makes forwarding decisions based on link state. But all of the above are simple and not suitable for the severe changes in network.

### B. Position-based routing protocol

Greedy Perimeter Stateless Routing (GPSR) <sup>[13]</sup>, which is an earlier methods in geographic routing. GPSR uses two data forwarding schemes: greedy forwarding and perimeter forwarding. The former is the primary forwarding strategy, while the latter is used in regions where the primary one fails. In greedy forwarding, when a node receives a packet with the destination's location, it chooses the next node from its neighbors which is geographically closest to the destination, and then forwards the packet to it. When a node whose neighbors are all farther away from the destination than itself, perimeter forwarding is performed. In Greedy Perimeter Coordinator Routing (GPCR) <sup>[14]</sup>, the current node looks for coordinators in an intersection, and as far as possible to forward messages to them, this will reduce the copies in the network. VANET Cross Link Corrected Routing (VCLCR) <sup>[15]</sup> is composed of greedy forwarding and perimeter forwarding as well, but the packet records edges and intersections which it can pass or not. The node can choose the next hop more accurately with the information stored in the packet. GeoDTN+Nav <sup>[16]</sup> is a hybrid routing algorithm and has three forwarding modes: greedy mode, perimeter mode and DTN mode. According to the network state, nodes switch modes, and then make forwarding decisions. These routing algorithms contain flooding strategy and have several disadvantages. For instance, if there is no coordinator exists in an intersection, GPCR shows a poor performance. VCLCR need packets to record edges and intersections, which is very complicated and needs additional network resources.

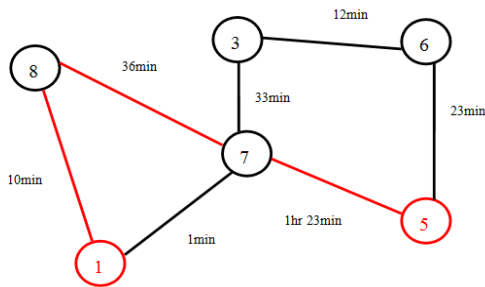


Figure 1. Bus lines graph example

### C. Routing protocol applied in bus network

Urban public transport network is a typical representative of DTNs <sup>[20-22]</sup>. In this kind of network, many routing protocols were presented. MaxProp Routing <sup>[23]</sup> forward messages to other nodes in a special order that takes into account message hop counts and message delivery probabilities based on previous encounters. But it is a generic algorithm for public transport network. Bus Line-based

Effective Routing (BLER) <sup>[24]</sup> uses the route contact oracle to find a shortest path among bus lines. In Fig. 1, a path from bus line 1 to bus line 5 goes through bus line 8 and 7, because a total duration of the contacts on this path is longer than others, hence the probability of having meetings is higher and the effect is better. Although this algorithm can ensure the message to be forwarded successfully, in the real world, the time of connection between vehicles is always changing, and should not forward messages entirely based on the connection time.

## III. ALGORITHM DESIGN

### A. Simulation scenarios

We set up the simulation scenario with buses, light rail vehicles and Access points (APs). Each bus line or light rail line has a certain number of stops, and the APs only exist in each commercial center. We regard the buses or light rail vehicles moving on the same road as one group. Moreover, if a bus or light rail vehicle passes a commercial center, then it and the AP also belong to the same group, we assign ID numbers for them. This ID number includes group number and individual number. Each bus or light rail vehicle stores geographic information of all stations, and our routing strategy is based on that. AP receives service information from buses or light rail vehicles, and then broadcast them for people in the commercial centers. In order to make simulation scenario to be more close to the real world, we then set up the second one, where we use the crowd nodes instead of APs. The performance of our routing protocol in these simulation scenarios will be commented in section IV.

In the real world, commercial center is a place with convenient transportations, and there must be many stations of buses or light rail vehicles. Therefore, the simulation scenarios we designed are reasonable and close to the reality. We mainly study how to broadcast public service information among commercial centers through public transport networks. Our proposed mechanism is based on the idea of finding a shortest path between two bus lines instead of finding a shortest path between two buses <sup>[24]</sup>. Then we simplify the method for acquiring the route information of buses or light rail vehicles and change the method of choosing the shortest path, all these can improve routing efficient.

The specific improvement as follows:

1. Bus lines and light rail lines are fixed and have numerous stations. We use coordinates of the stations to represent a bus line or light rail line, we can't acquire the route exactly, but we can acquire the positions where buses or light rail vehicles must pass and stop. Most buses run on the main road, therefore, using coordinates of the stations to replace a bus line or a light rail line is reasonable, and simplify the previous method that using GPS to get path information.

2. We design a data table to store the minimum distance between current position and the target line, replace the route contact oracle what is used for storing the total duration of the contacts among buses. Routing protocol forwards messages based on the data table, then it can avoid an

inaccurate condition that traffic jams lead to increase the contact time among buses.

### B. The BLR algorithm

According to the characteristics of urban public transport networks, we propose Bus Line-based Routing (BLR), show in Fig. 2 and Fig. 3. The following notations are used in our algorithm:

- Msg: a message.
- Dis: the minimum distance between current location and one station of target line.
- Sta: a station of target bus line or light rail line.
- MsgDstNumber: the group number of message's destination.
- Con: a connection between two nodes.
- Loc: current position.

```

1: Procedure Contact (Node, Msg, Con)
2:   When node i contacts node j
3:   Initialize the local network topology
4:   If Con.isUp ()
5:     NodeID ← GetNodeID(Node)
6:     To node i & j:
7:     For Collection<Message> Msg: GetMessageCollection ()
8:       MsgDstNumber ← GetMsgDstNumber(Msg)
9:       Switch (MsgDstNumber)
10:      Dis ← CalculateDistance (Loc, Sta)
11:      End Switch
12:      UpdateDistance (NodeID, Msg, Dis)
13:    End For
14:  End If

```

Figure 2. The contact procedure of BLR

```

1: Procedure Routing ()
2: While (True)
3:   For Connection Con: GetConnection ()
4:     If otherHost.isTransferring ()
5:       Continue
6:     End If
7:     For Collection<Message> Msg: GetMessageCollection ()
8:       If otherHost.hasMessage (Msg)
9:         Continue
10:      End If
11:      If GetDistance (Msg, otherHost) < GetDistance (Msg, Host)
12:        AddInTransitQueue (Msg)
13:      End If
14:    End For
15:  End For
16:  Forwarding messages
17: End While

```

Figure 3. The routing procedure of BLR

In our algorithm, each node stores all the station information, and we use the minimum distance between the current position and one station of the target line to replace the minimum distance between lines. During the initial phase, nodes acquire the group number of each message's destination to calculate the minimum distance, and then the nodes update their data tables. During the transmission phase, if the other side is sending messages, then disconnect the connection and to find the next node. If both sides have a

same message, the message will not be pushed into the transit queue. If the other node does not have a message, moreover, the other node is closer to the target line, then pushes the message into the transit queue, forwards the message to it.

### C. The CCBLR algorithm

BLR can take full advantage of public transport networks to forward messages, and make full use of the idea that the minimum distance between bus lines. We also use location information of scattered stations to replace a specific route, and then make it simple to calculate the minimum distance between lines. But if a message reach the target node's group, when two nodes which are in the same group encounter each other and will also forward a copy of the message to the other node, this will lead to increase the number of copies. In order to further reduce the copies and overhead, we only improve the part of routing procedure and propose Copy Constrained Bus Line-based Routing (CCBLR). The part of CCBLR's routing procedure shows in Fig. 4.

```

1: Procedure Routing ()
2: While (True)
3:   For Connection Con: GetConnection ()
4:     If otherHost.isTransferring ()
5:       Continue
6:     End If
7:     For Collection<Message> Msg: GetMessageCollection ()
8:       If otherHost.hasMessage (Msg)
9:         Continue
10:      End If
11:      If GetGroupNumber (otherHost) = GetGroupNumber (Host)
12:        If GetDistance (Msg, otherHost) < GetDistance (Msg, Host)
13:          AddInTransitQueue (Msg)
14:        End If
15:      Else
16:        If GetMsgDstNumber (Msg) == GetGroupNumber (otherHost)
17:          If otherHost moving in the direction of the destination
18:            AddInTransitQueue (Msg)
19:          End If
20:        End If
21:      End If
22:    End For
23:    Forwarding messages
24:    If (startTransfer (Msg, Con) ==OK) &&
25:      (GetMsgDstNumber (Msg) == GetGroupNumber (otherHost))
26:      Host.deleteMessage (Msg, ture)
27:    End If
28:  End While

```

Figure 4. The routing procedure of CCBLR

The improved routing protocol provides that when communication connection is established, the nodes check that whether they have a same group number or not. If the nodes and the message's destination have a same group number, then forward the messages based on the size of the vector angle, show in Fig.5 and Fig. 6. One message is on the BUS-A and its destination is A. If two buses travel in the same direction of A, then we can know that  $0 < \cos \beta < \cos \alpha$ , and if two buses travel in the opposite direction of A, then we can know that  $\cos \alpha < \cos \beta < 0$ , therefore, BUS-A forward this message

to BUS-B. If one bus travels in the direction of A and the other one travels in the opposite direction of A, then we can also know that  $\cos \beta > 0 > \cos \alpha$ , Bus-A forward the message to Bus-B. After transmitting successfully, the node itself will delete the message, and this way will reduce the copies and overhead.

#### IV. SIMULATION AND RESULT

In this section, we simulate the performance of BLR and CCBLR with the Opportunistic network environment (ONE) simulator [25], and deployed the geographic information of Helsinki as map scenario. We test our routing algorithms in two simulation scenarios. At first, we only test BLR in a simple scenario, with a set of 9 commercial centers, 3 light rail lines and 8 bus lines. Each commercial center set with one AP.

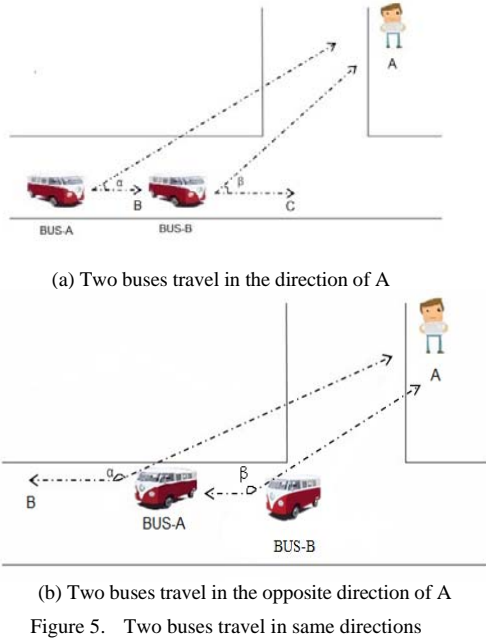


Figure 5. Two buses travel in same directions

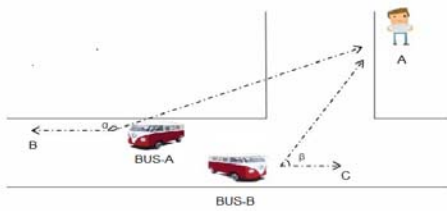


Figure 6. Two buses travel in different directions

In this scenario, the transmission range of each node is 350m and their buffer is 500MB. If buffer is full and a new message arrives, then the message with the least TTL is dropped. We simulate a 12-hours public traffic and randomly generate 124 messages. The message has a fixed size of 7MB. We simulate the performance of CCBLR and BLR in

second simulation scenario which is mentioned in section III. Crowd nodes' transmission range is 10m and their buffer is 50MB. In this experimental scenario, the different group of nodes can send messages to each other through the urban public transport network.

#### A. Simulation scenario I

In this part, we compare the BLR with Epidemic Routing (EP), Spray and Wait Routing (SWR), Prophet Routing (PR) and First Contact Routing (FCR) [10], and show the performance in terms of delivery rate, average delay, hop count and the number of copies. Table I shows the distribution of experimental nodes. Each parameter only represents the number of nodes for each road or commercial center. We run 6 simulations for each algorithm and each set of parameters.

TABLE I. DISTRIBUTION OF EXPERIMENTAL NODES

Node Types	Experiment No.					
	I	II	III	IV	V	VI
Bus or Light Rail	2	3	4	5	6	7
AP	1	1	1	1	1	1

Fig. 7 and Fig. 8 show the results of delivery rate and the number of copies. FCR and SWR are simple routing protocols, hence the number of copies is smaller than others and they have a poor performance in delivery rate. Because of the targeted routing strategies, BLR has a much higher delivery rate which is closed to the result of ER. BLR also tries to reduce the number of copies, so it has a smaller number of copies than ER and PR.

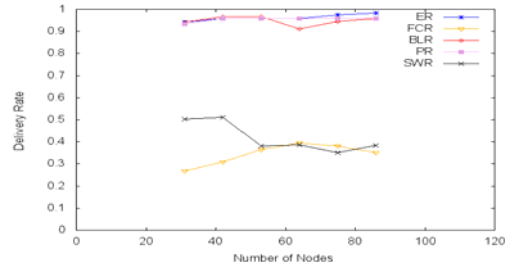


Figure 7. Delivery rate for different number of nodes

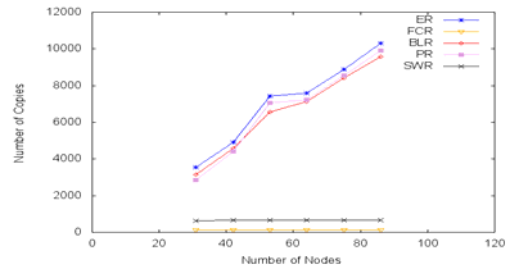


Figure 8. Number of copies for different number of nodes

Fig. 9 shows the results of each routing protocol in average hop count. Because FCR has no explicit purpose for forwarding messages, it shows poor performance in average hop count. ER has a maximum number of copies and also has no explicit purpose for forwarding messages, thus, the average hop count is bigger than SWR, PR and BLR. SWR is a routing protocol that nodes spray a certain number of copies, then carry messages, until meet the target node they deliver the messages, hence the average hop count is minimum. BLR has a better result in average hop count due to the routing strategy that forward messages selectively.

FCR which forwards only one copy of each message to the first available contact, hence the average delay is longer than others. ER don't limit the number of copies lead to the average delay is minimum. BLR has a smaller number of copies than ER, it forwards messages based on minimum distance between current position and the target bus line or light rail line, hence there is a situation that current position is close to the target line, but it is far away from the target node, then this situation increase the delay time. In Fig. 10, we can also see that BLR is better than PR and SWR in average delay.

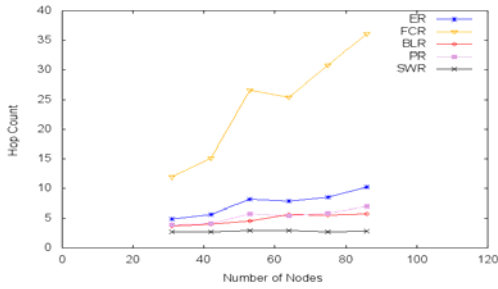


Figure 9. Number of hops for different number of nodes

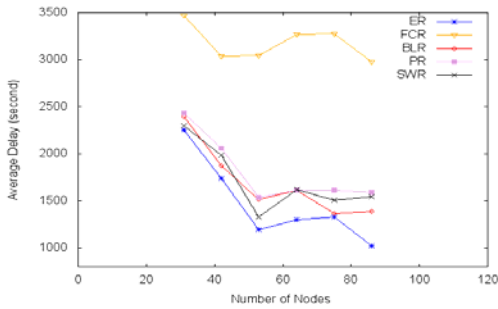


Figure 10. Average delay for different number of nodes

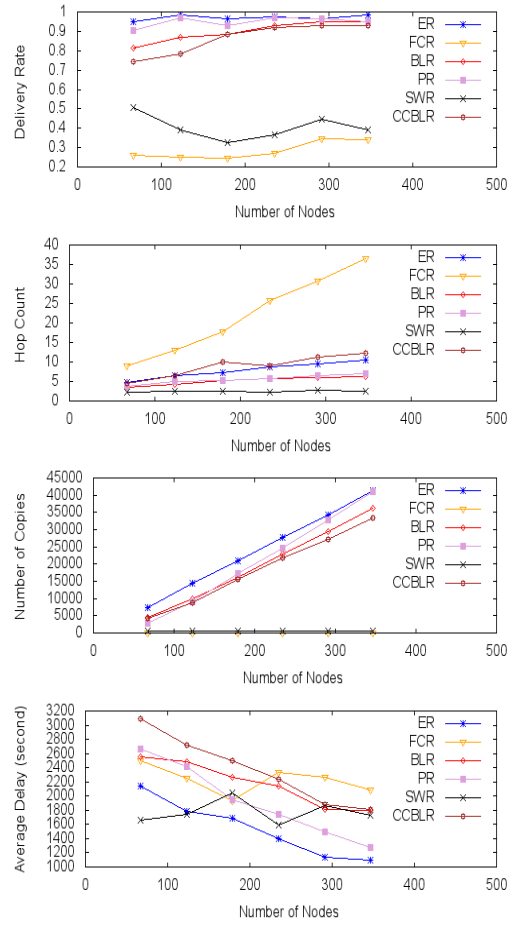


Figure 11. Performance for different routing

## B. Simulation scenario II

We simulate the performance of CCBLR and BLR in the second experimental scenario, Table II shows the distribution of experimental nodes.

TABLE II. DISTRIBUTION OF EXPERIMENTAL NODES

Node Types	Experiment No.					
	I	II	III	IV	V	VI
Bus or Light Rail	2	3	4	5	6	7
Crowd	5	10	15	20	25	30

From Fig. 11, we can see that by adding mobile nodes in each commercial center, BLR and CCBLR also have a good performance in delivery rate with the increasing number of nodes. But they generate less message copies than ER and PR. In CCBLR, nodes are limited to forward messages when messages reach the target group, this lead to the copies are little less than BLR, its average hop count and delay time are higher than before. But with the nodes increasing, compared to other routing protocols, we can conclude that CCBLR has a better performance.

## V. CONCLUSION AND FUTURE WORKS

Urban public transport network is regarded as one of the hot topic of current researches in DTNs. We proposed the BLR and CCBLR, which based on the minimum distance between bus lines to forward messages in a city scenario. We used scattered stations of bus to replace the bus lines, this simplify the previous method that using the GPS to get the information of the full path. We designed the data table to store the minimum distance between bus lines, develop a specific routing strategy to forward messages. Although the BLR, CCBLR have poor performance in delay time, they have higher delivery ratio and better results in copies and overhead. In the future, we will add taxi, pedestrians and other communication nodes in the scene to make the network model much closer to the real world, and improve the performance of the proposed routing protocols in this scene.

### ACKNOWLEDGMENT

This research was supported by the Foundation of Inner Mongolian Autonomous Region Educational Committee (NJZY11017, NJZZ12005), "Spring Sunshine Program" of the PRC Ministry of Education (z2009-1-01011), National Natural Science Foundation of China (61162006).

### REFERENCES

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets," Proc. Special Interest Group Data Communication (SIGCOMM '03), Karlsruhe, Germany, 2003, pp.27-34.
- [2] A. Tovar, T. Friesen, K. Ferens, B. McLeod, "A DTN wireless sensor network for wildlife habitat monitoring," Electrical and Computer Engineering (CCECE), Calgary, Canada, 2010, pp.1-5.
- [3] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.S. Peh, D. Rubenstein, "Energy-Efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," Proc. 10th Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS-X), San Jose, CA, Oct. 2002, pp.96-107.
- [4] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "CarTel: a distributed mobile sensor computing system," Proc. 4th Embedded networked sensor systems, Boulder, USA, Nov. 2006, pp. 125-138.
- [5] A. Pentland, R. Fletcher, and A. Hasson, "DakNet: Rethinking connectivity in developing nations," Computer, Vol.37, no.1, Aug., 2004, pp.78-83.
- [6] T. Spyropoulos, K. Psounis, C.S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," Proc. IEEE Conf. Sensor and Ad Hoc Communications and Networks (SECON), 2004, pp.235-244.
- [7] V. Ahdad, D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke Univ., Tech. Rep. CS-200006, Apr., 2000.
- [8] A. Lindgren, A. Doria, O. Schelen, "Probabilistic routing in intermittently connected networks," SIGMOBILE Mobile Comput. Commun. Rev., Vol.7, no.3., Jul., 2003, pp.19-20.
- [9] T. Spyropoulos, K. Psounis, C.S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," Proc. ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN), Philadelphia, USA, 2005, pp.252-259.
- [10] S. Jain, K. Fall, R. Patra, "Routing in a delay tolerant network," Proc. Special Interest Group on Data Communication (SIGCOMM'04), Portland, USA, 2004, pp.145-158.
- [11] T. Clausen, P. Jacquet et al., "Optimized Link State Routing Protocol," Request for Comments 3626, MANET Working Group, <http://www.ietf.org/rfc/rfc3626.txt>, October 2003. Work in progress.
- [12] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint Relaying: An Efficient Technique for Flooding in Mobile Wireless Networks," INRIA res. rep. RR-3898, 2000.
- [13] B. Karp, H.T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," Proc. Sixth Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking (MobiCom 2000), Boston, USA, Aug. 2000, pp. 243-254.
- [14] C. Lochert, M. Mauve, H. Füllner, "Geographic Routing in City Scenarios," ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), vol. 9, no. 1, pp.69-72, 2005.
- [15] K.C. Lee, P.C. Cheng, J.T. Weng, L.C. Tung, M. Gerla, "VCLCR: A Practical Geographic Routing Protocol in Urban Scenarios," UCLA Computer Science Department, Tech. Rep. TR080009, 2008.
- [16] P.C. Cheng, K.C. Lee, M. Gerla, J. Häiri, "Geographic DTN Routing with Navigator Prediction for Urban Vehicular Environments," Mobile Networks and Applications, Springer, vol. 15, Feb. 2010, pp. 61-82.
- [17] M. Demmer, K. Fall, "DTLSR: Delay Tolerant Routing for Developing Regions," Proc. Networked Systems for Developing Regions (NSDR '07), Kyoto, Japan, 2007, Article No. 5
- [18] J. Moy, OSPF version 2, 1998.
- [19] E.P.C. Jones, L. Li, J.K. Schmidtke, P.A.S. Ward, "Practical routing in delay-tolerant networks," Proc. ACM SIGCOMM Workshop Delay Tolerant Networking (WDTN '05), 2005, pp. 943-959.
- [20] X. Zhang, J. Kurose, B.N. Levine, D. Towsley, H. Zhang, "Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing," In MobiCom '07: Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking, Montreal, Quebec, Canada, 2007, pp. 195-206.
- [21] H.S. Park, J.D. Kim, "Modeling and Analysis of DTN in Metropolitan Bus Network," Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication, SEOUL, KOREA, 2011, Article No. 20.
- [22] A. Bujari, C.E. Palazzi, D. Maggiorini, C. Quadri, G.P. Rossi, "A Solution for Mobile DTN in a Real Urban Scenario," in Proc. of the IEEE Wireless Communications and Networking Conference Workshops, WCNCW 2012, Paris, France, Apr 2012, pp. 344-349.
- [23] J. Burgess, B. Gallagher, D. Jensen, B. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," University of Massachusetts, 2006.
- [24] M. Sede, X. Li, D. Li, M.Y. Wu, M. Li, W. Shu, "Routing in Large-Scale Buses Ad Hoc Networks," Wireless Communications and Networking Conference, Las Vegas, USA, 2008, pp.2711 - 2716.
- [25] A. Keränen, J. Ott, T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," Proceedings of the 2nd International Conference on Simulation Tools and Techniques, Brussels, Belgium, 2009, Article No. 55