2010 International Conference on Broadband, Wireless Computing, Communication and Applications

Impact of Source and Destination Movement on MANET Performance Considering BATMAN and AODV Protocols

Elis Kulla*, Makoto Ikeda†, Leonard Barolli‡, Rozeta Miho§

*Graduate School of Engineering
Fukuoka Institute of Technology (FIT)
3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811–0295, Japan
E-mail: eliskulla@yahoo.com

†Center for Asian and Pacific Studies
Seikei University
3-3-1 Kichijoji-Kitamachi, Musashino-Shi, Tokyo 180–8633, Japan
E-mail: makoto.ikd@acm.org

‡Department of Information and Communication Engineering
Fukuoka Institute of Technology (FIT)
3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811–0295, Japan
E-mail: barolli@fit.ac.jp

§Department of Electronics and Telecommunication
Polytechnic University of Tirana
Mother Teresa Square, Nr.4, Tirana, Albania
E-mail: rmiho@fie.upt.al

***Abstract*—In recent years, MANETs are continuing to attract the attention for their potential use in several fields such as military activities, rescue operations and time-critical applications. In this paper, we present the implementation and analysis of our implemented MANET testbed considering AODV and BATMAN protocols for wireless multi-hop networking. We investigate the effect of mobility and topology changing in MANET. We evaluate the performance of the routing protocols through experiments in real environment. In this work, we consider four scenarios: Static, Source Moving, Destination Moving and Source-Destination Moving. We assess the performance of our testbed in terms of throughput, number of dropped packets and delay. We found that, when routes are changing often, the BATMAN has better performance than AODV because it put the packets in a buffer. While, because AODV is a reactive protocol, it introduces some delay in the network.**

***Keywords*-MANET; Testbed; BATMAN; AODV; Routing Protocols.**

I. Introduction

During recent year, we have witnessed a lot of research on wireless networks [1]–[8]. There are two network architectures for wireless networks: infrastructure and ad-hoc architecture.

Wireless networks often extend, rather than replace, wired networks, which are referred to as infrastructure networks. The wide area and local area wired networks are used as the backbone network. The wired backbone connects to special switching nodes called Base Stations (BSs). The BSs are often conventional PCs and workstations equipped with custom wireless adapter cards. They are responsible for coordinating access to one or more transmission channel(s) for mobiles located within the coverage cell.

Ad-hoc networks, on the other hand, are multi-hop wireless networks in which a set of mobile nodes cooperatively maintain network connectivity. Ad-hoc networks are characterized by dynamic, unpredictable, random, multi-hop topologies with typically no infrastructure support. The mobile nodes must periodically exchange topology information which is used for routing updates.

A Mobile Ad hoc Network (MANET) is a collection of wireless mobile terminals that are able to dynamically form a temporary network without any aid from fixed infrastructure or centralized administration. In recent years, MANET are continuing to attract the attention for their potential use in several fields. Mobility and the absence of any fixed infrastructure make MANET very attractive for mobility and rescue operations and time-critical applications.

Most of the work for MANETs has been done in simulation, as in general, a simulator can give a quick and inexpensive understanding of protocols and algorithms [9]–[12]. However, experimentation in the real world are very important to verify the simulation results and to revise the models implemented in the simulator. A typical example of

978-0-7695-4236-2/10 $26.00 © 2010 IEEE
DOI 10.1109/BWCCA.2010.54

this approach has revealed many aspects of IEEE 802.11, like the gray-zones effect [13], which usually are not taken into account in standard simulators, as the well-known *ns-2* simulator [14].

There are a lot of computer simulation results on the performance of MANET, e.g. in terms of end-to-end throughput, delay and packet loss. However, in order to assess the computer simulation results, real-world experiments are needed and a lot of testbeds have been built to date [15]–[17]. The baseline criteria usually used in real-world experiments is guaranteeing the repeatability of tests, i.e. if the system does not change along the experiments. How to define a change in the system is not a trivial problem in MANET, especially if the nodes are mobile.

In our previous work, we found the following results. We proved that while some of the OLSR's problems can be solved, for instance the routing loop, this protocol still have the self-interference problem. There is an intricate interdependence between MAC layer and routing layer, which can lead the experimenter to misunderstand the results of the experiments. We carried out the experiments considering stationary nodes of ad-hoc network. We considered the node mobility and carry out experiments for Ad-hoc On demand Distance Vector (AODV), Optimized Link State Routing (OLSR) and Better Approach to Mobile Ad hoc Networks (BATMAN) protocols [18]. We found that throughput of TCP were improved by reducing Link Quality Window Size (LQWS), but there were packet loss because of experimental environment and traffic interference. For TCP data flow, we got better results when the LQWS value was 10.

In this paper, we focus on comparing the performance of two types of routing algorithms AODV, which is a reactive routing protocol, and BATMAN, which is a proactive routing protocol, for source and destination moving scenarios. These protocols have been gaining great attention within the scientific community. Furthermore, the *aodv-uu* [19] and the *batmand* [20] softwares we have used in our experiments are the most updated softwares we have encountered.

In this work, we compare the performance of AODV and BATMAN for different scenarios. We implemented four MANET scenarios and carried out many experiments using our testbed.

The structure of the paper is as follows. In Section II, we give a short description of AODV and BATMAN. In Section III, we describe the testbed and its implementation. In section IV, we present the testbed topology description. In Section V, we present experimental evaluation. Finally, conclusions are given in Section VI.

II. Routing Protocols

A. AODV Overview

AODV is one of the most popular reactive routing protocol for MANETs [21]. As a reactive (on demand) protocol, when a node wants to transmit data, it first starts a route discovery process, by flooding a RREQ (Route Request) packet. The RREQ packets are forwarded by all nodes when they are received. This procedure continues until the destination is found. On the way to destination, the RREQ informs all the intermediate nodes about the route to the source. When the RREQ reaches the destination, destination sends a Route Reply (RREP) packet which follows the reverse path discovered by RREQ. This informs all intermediate nodes about a route to the destination node. After RREQ and RREP are delivered to their destination, each intermediate node on the route knows the node to forward data packets in order to reach source or destination. Thus data packets do not need to carry addresses of all intermediate nodes in the route. They just need the address of the destination node, thus decreasing noticeably routing overheads.

A third kind of routing message, called Route Error (RERR), allows nodes to notify errors, for example, because a previous neighbor has moved and is no longer reachable. If the route is not active (i.e., there is no data traffic flowing through it), all routing information expire after a timeout and is removed from the routing table.

In AODV, the route discovery process may last for a long time, or it can be repeated several times, due to potential failures during the process. This introduces extra delays, and consumes more bandwidth as the size of the network increases.

B. BATMAN Overview

In OLSR, there is a serious synchronization problem between the topology messages and the routing information stored inside every node. In other words, a mismatch between what is currently stored in the routing tables and the actual topology of the network may arise. This is due to the propagation time of the topology messages. Routing loops are the main effect of such problem. To solve this problem, BATMAN has been introduced.

In BATMAN, there is no topology message dissemination. Every node executes the following operations.

1) Sending of periodic advertisement messages, called OriGinator Message (OGM). The size of these messages is just 52 bytes, containing: the IP address of the originator, the IP address of the forwarding node, a TTL value and an increasing Sequence Number (SQ).
2) Checking the best one-hop neighbor for every (known) destination in the network by means of a ranking procedure.
3) Re-broadcasting of OGMs received via best one-hop neighbor.

The timer in BATMAN is used for sending OGMs. The bi-directionality of links is checked using the SQ of OGM. If the SQ of and OGM received from a particular node falls within a certain range, the corresponding link is considered bi-directional. For example, suppose that in a time interval T, the node A sends Tr messages, where r is the rate of OGM messages. The neighbors of A will re-broadcast the OGMs of A and also other node's OGMs. When A receives some OGMs from a neighbor node, say B, it checks if last received OGM from B has a SQ less or equal to Tr. If it does, then B is considered bi-directional, otherwise it is considered unidirectional. Bi-directional links are used for the ranking procedure. The quantity Tr is called bidirectional sequence number range. The ranking procedure is the same as the link quality extension of OLSR. In few words, every node ranks its neighboring nodes by means of a simple counting of total received OGMs from them. The ranking procedure is performed on OriGinator (OG) basis, i.e. for every originator. Initially, for every OG, every node stores a variable called Neighbor Ranking Sequence Frame (NBRF), which is upper bounded by a particular value called ranking sequence number range. We suppose that there is a rank table in every node which stores all the information contained in the OGMs.

Whenever a new OGM is being received via a bi-directional link, the receiving node executes the following steps.

1) If the sequence number of the OGM is less than the corresponding NBRF, then drop the packet.
2) Otherwise, update the NBRF=SQ (OGM) in the rank table.
3) If SQ (OGM) is received for the first time, store OGM in a new row of the rank table.
4) Otherwise, increment by one the OGM count or make ranking for this OGM.

Finally, the ranking procedures select the best one-hop neighbor as that neighbor which has the highest rank in the ranking table. Let us note that the same OGM packet is used for: link sensing, neighbor discovery, bi-directional link validation and flooding mechanism. While this feature eliminates routing loops because no global topology information are flooded, the self-interference due to data traffic can cause oscillations in the throughput as we will see in our experiments. Other details on BATMAN can be found in [22], [23].

III. Testbed Description

A. Testbed Environment

We implemented a MANET testbed and carried out experiments in the fifth floor of Building D, at Fukuoka Institute of Technology. This testbed provides the environment to make different measurements for indoor and outdoor communications. However, in this paper we deal only with indoor environment.

B. Operating System and Routing Software

The operating system installed on machines is Ubuntu 9.04 Linux (x5), eeeUbuntu 9.04 Linux (x1) all with kernel 2.6.28-18-generic and Fedora Core 4 Linux (x1) as shown in Table I. Each of them can support all routing software installed.

Table I
Node addressing table.

Node ID	IP Address	Operating System
Node 1	192.168.0.1	Fedora Core 4
Node 2	192.168.0.2	Ubuntu 9.04
Node 3	192.168.0.5	Ubuntu 9.04
Node 4	192.168.0.6	eeeUbuntu 9.04
Node 5	192.168.0.7	Ubuntu 9.04
Node 6	192.168.0.10	Ubuntu 9.04
Node 7	192.168.0.11	Ubuntu 9.04

(a) Node 1

(b) Node 2

Figure 1. Hardware of the testbed.

In each machine, the AODV and BATMAN routing softwares are installed from their source code in their respective web pages. Both of them are open source (see [19], [20] for more information).

C. Network Configuration

All machines used their own wireless adapter, except for the Fedora machine which uses a Linksys wireless card.

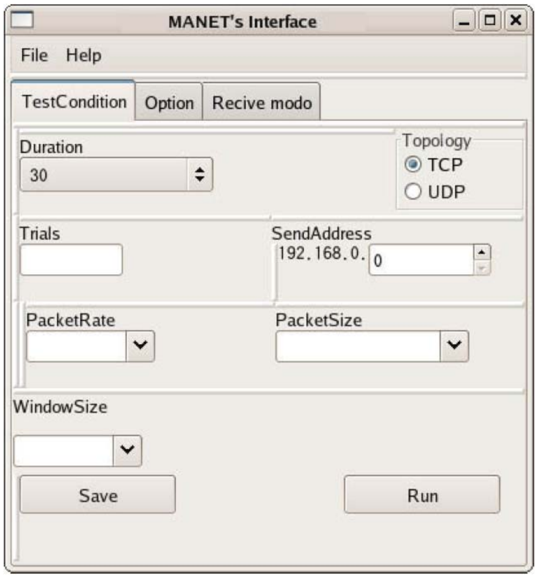


Figure 2. GUI interface for parameter settings.

The driver of Linksys can be found in [24]. Each machine wireless card transmits at a frequency of 2.412 GHz (channel 1), and is put to ad-hoc mode. In Fig. 1, we show a screenshot of every node we used in experiments. Node IDs and IP addresses are shown in Table I.

D. Traffic Generation and Getting the Data

After configuring the network all nodes are put to their respective position, in accordance to the experimental scenario. To generate some traffic between nodes, we used D-ITG (Distributed Internet Traffic Generator) software, which is a Traffic Generator [25]. With D-ITG, one could send different type of traffics from one node to another. The amount of information to be sent and the duration of the transmission is set as an option. After finishing the transmission, D-ITG offers decoding tools to get information about network metrics along the whole transmission duration.

E. Testbed Interface

All settings, editing and calculations can be done with the aid of a Graphical User Interface (GUI) as shown in [26]. This is helpful in saving time in the case of repeated experiments, and avoiding misprints during set-up. The GUI uses wxWidgets tool and each operation is implemented by Perl language. wxWidgets is a cross-platform GUI and tools library for GTK, MS Windows and Mac OS X. Many parameters are implemented in the interface such as transmission duration, number of trials, source address, destination address, packet rate, packet size, LQWS, and topology setting function. These parameters can be saved in a text file and can manage the experimental conditions in a better approach. The GUI interface of the implemented testbed is shown in Figure 2.

Table II
EXPERIMENTAL PARAMETERS.

Parameters	SS	SMS	DMS	SDMS
Nr. of Experiments	20	10	10	10
Duration of Experiment (s)	60	120	120	120
Packet Rate (pkt/s)	200	200	200	200
Packet Size (bytes)	512	512	512	512

IV. TOPOLOGY DESCRIPTION

The implemented testbed provides a real-time system for analysing various aspects of MANETs. The purpose of this paper is to evaluate the performance of two routing protocols: AODV and BATMAN. Performance evaluation is done for four different scenarios. The MAC filtering is not used in these experiments, so the nodes form e Mesh Topology. We describe the four scenarios in the following. The topologies for different experiments are shown in Fig. 3. All experimental parameters are shown in Table II.

For static scenario, 20 experiments were performed for each protocol, and every experiment lasted 60 seconds. The source node sent 512-byte packets, with a frequency of 200 packets per second. For moving scenarios, we performed 10 experiments and the experimental time was 20 seconds.

A. Static Scenario

In the Static Scenario (SS), first all nodes are put in the positions shown in Fig. 3(a). Then, in each machine, the routing protocol deamons are started. In this paper, we consider AODV and BATMAN and their deamons *aodvd* and *batmand*, respectively. To let the routing protocol initialize routes, no data was transmitted for the first five minutes.

B. Source Moving Scenario

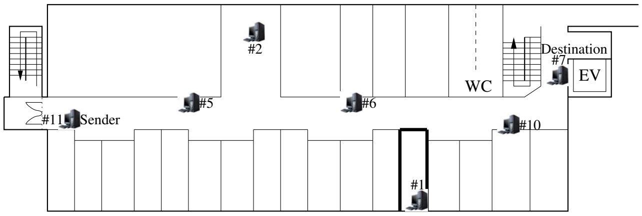
The Source Moving Scenario (SMS) is shown in Fig. 3(b). The nodes are in the same position as in SS (Fig. 3(a)), except that source node moved towards the destination node, as shown in Fig. 3(b). This movement is realized using a simple wheeled office chair.

C. Destination Moving Scenario

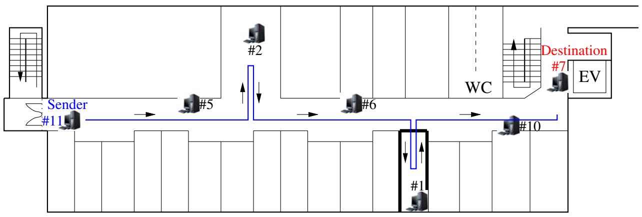
In Fig. 3(c), we show the Destination Moving Scenario (DMS). The destination node moves away from the source, starting its movement in the same position as the source node. At the end of 120 seconds, destination node and source node have the maximum distance between them.

D. Source-Destination Moving Scenario

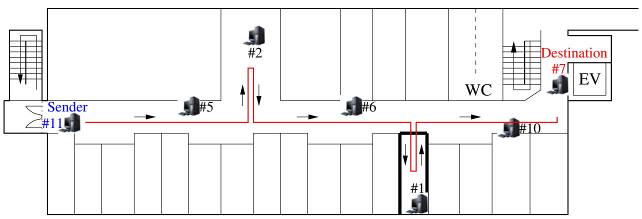
As shown in Fig. 3(d), in Source-Destination Moving Scenario (SDMS), both source node and destination nodes are moving. Starting near the position of node 6, they both move away from each other for the first 60 seconds. Then, they go back by the same route to the starting position for the last 60 seconds.



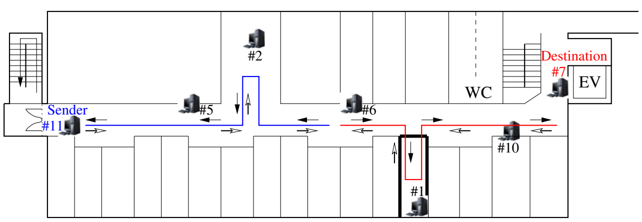
(a) SS



(b) SMS



(c) DMS



(d) SDMS

Figure 3. Different topologies for experiments.

V. Performance Evaluation

A. Experimental Settings

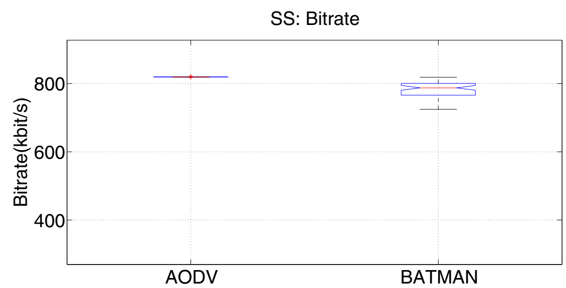
We performed the experiments in indoor environment (our departmental floor) with the size nearly 70 m × 25 m as shown in Figure 3. The D-ITG is used to create the traffic and to collect the data. Data in the network were collected in a Mesh Topology for different scenarios of node movement and for two routing protocols. We were interested in Bitrate (kbps), Delay (ms) and Packetloss (No.of packets).

We used CBR (Constant Bit Rate) over UDP to create the traffic. The transmission rate of the data flow is 200 pkts/s, and the packet size is fixed to 512 kB, meaning a maximum bitrate of 819.2 kbps. Nodes (laptops) could access each other within the 70 meter region where the experiments were performed. We checked this by the *ping* command of Ubuntu 9.04. In total, we performed 8 experiments, as shown in Table III.

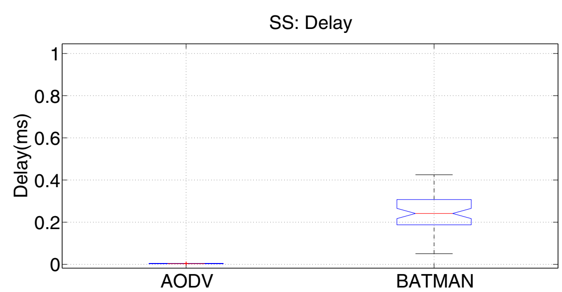
As MAC protocol we used the IEEE 802.11b protocol and configured the wireless cards to operate at central frequency 2.412 GHz (channel 1) and with enough power to have

Table III
Average values for different experiments.

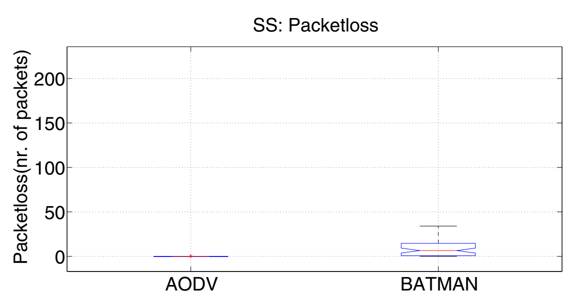
Nr.	Scenario	Protocol	Bitrate	Delay	Packetloss
1	SS	AODV	819.1863	0.0032	0.000076
2		BATMAN	783.0118	0.2456	0.0724
3	SMS	AODV	613.9733	1.5855	0.2942
4		BATMAN	641.5053	1.5066	0.2432
5	DMS	AODV	720.2372	0.7445	0.1654
6		BATMAN	729.6102	0.7197	0.1363
7	SDMS	AODV	727.7739	0.8986	0.2265
8		BATMAN	742.9634	1.1925	0.2052



(a) Bitrate



(b) Delay



(c) Packetloss

Figure 4. Different metrics for different protocols in SS.

connectivity with every node in the network. The main interest on these experiments was in the routing protocols and their behaviour in different scenarios, so all MAC parameters were kept unchanged. We should mention that during experiments all the IEEE 802.11 spectrum had been used by other access points operating within the campus, causing a considerable interference.

We took samples of 500 ms for every experiment, and computed the averages of each sample, using linux bash scripting and Matlab.

B. Experimental Measurements

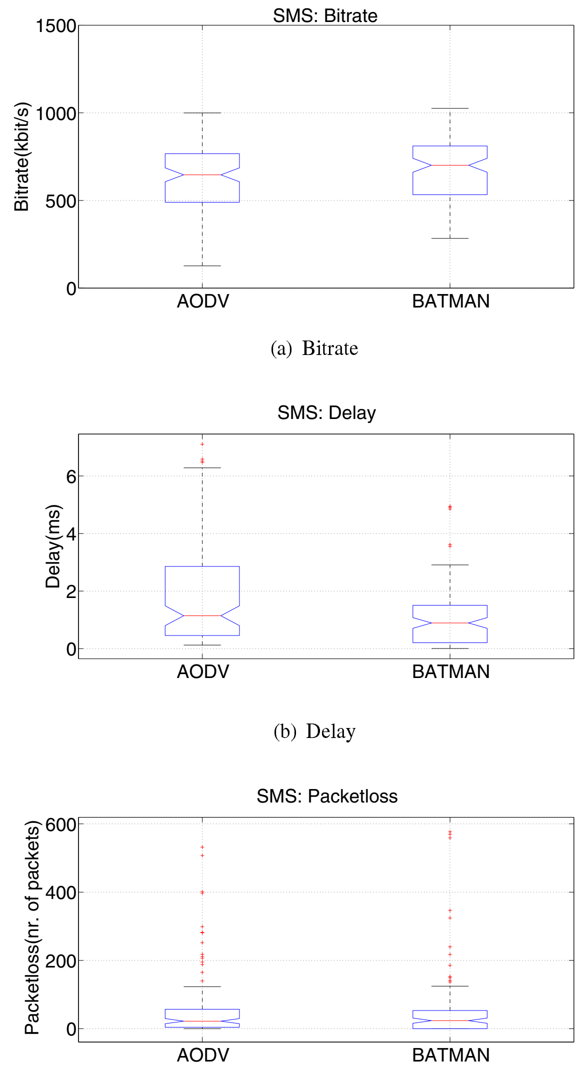
In Table III, we show all the calculated average values for every experiment. We investigated all mean values of Bitrate, Delay and Packetloss, which are measured in "kilobits per second (kbps)", "milliseconds (ms)" and "percentage (%)", respectively.

For SS, in Fig. 4, we can see that in the case of AODV the bitrate is almost the maximum ($max = 819.2$). This means the routes have been established and the communication is performed at almost maximum bitrate. This is also shown in Table III. While in the case of BATMAN protocol, we encountered some changes. The bitrate value falls to less than 800 kbps and the delay reaches values up to $246ns$. This phenomena is caused by radio irregularities at the time of performing the experiments. When the communication link is broken BATMAN uses another link which is probably slower.

In SMS, the source node is approaching the destination node, but at two time periods the nodes loose the Line of Sight (LOS) and a complete route of 2 or more hops is difficult to be established. In Fig. 5, we show three metrics in boxplot. When no routes are available for BATMAN, packets are buffered, inside nodes, instead of being dropped like in AODV. This fact is also reflected in Table III and in Fig. 5(a), where BATMAN shows a better performance than AODV. We observe the same performance for packetloss as shown in Fig. 5(c). However, as shown in Fig. 5(b), BATMAN shows a slightly better performance than AODV considering delay.

In DMS, the destination node is moving away from the source node. In Fig. 6, we show three metrics in boxplot. As shown in Fig. 6(a), BATMAN has a better performance than AODV. As shown in Fig. 6(b), BATMAN shows a slightly better performance than AODV considering delay metric. When the bitrate has a low value, we noticed a proportional increase in packetloss. In Fig. 6(c) is shown that AODV has a slightly worse performance than BATMAN.

In SDMS, during the first 60 seconds both nodes are moving away from each other and then during the last 60 seconds they are approaching each other via the same route of movement. As shown in Fig. 7(a), BATMAN has a better performance than AODV regarding bitrate metric. Also, as shown in Fig. 7(b), BATMAN shows a slightly better performance than AODV considering delay. Regarding packetloss metric, both protocols show almost the same performance as shown in Fig. 7(c).



(a) Bitrate

(b) Delay

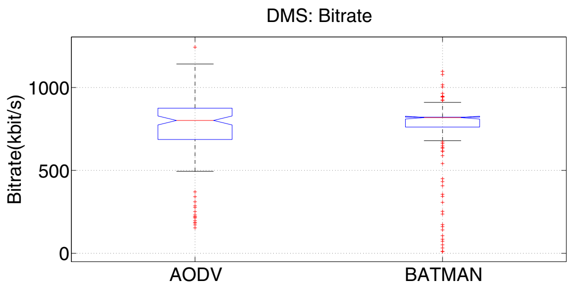
(c) Packetloss

Figure 5. Different metrics for different protocols in SMS.

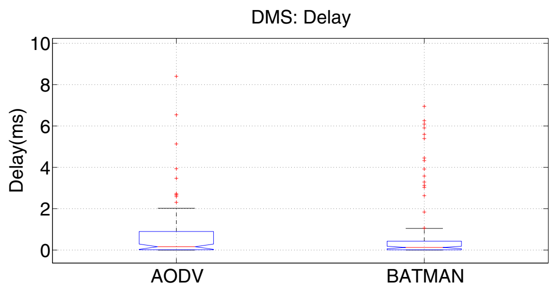
VI. Conclusions

In this paper, we carried out experiments for a small MANET with 7 nodes. We used AODV and BATMAN protocols for experimental evaluation and comparison and we implemented four scenarios: SS, SMS, DMS and SDMS. We considered 3 metrics for performance evaluation: bitrate, delay and packetloss.

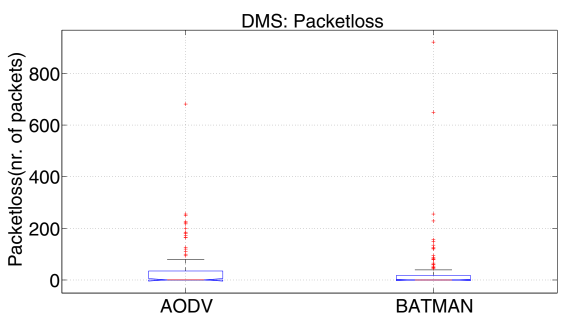
We investigated the performance of MANET when two communicating nodes loose LOS during a period of time. We found that BATMAN protocol has a better behaviour than AODV in the case when no routes are available, because of the buffering feature.



(a) Bitrate

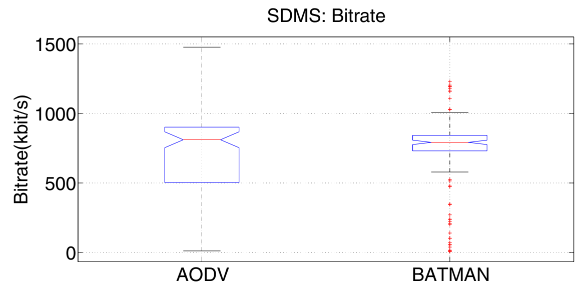


(b) Delay

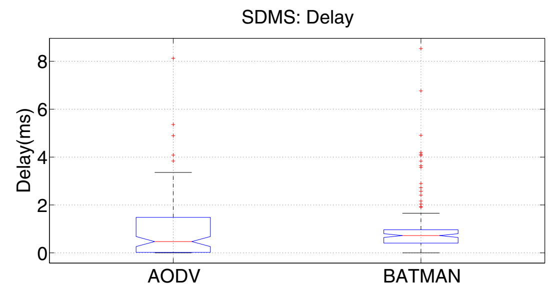


(c) Packetloss

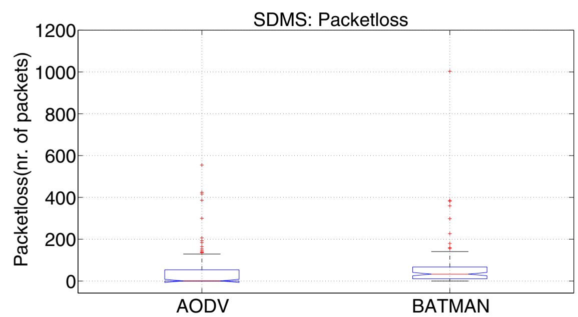
Figure 6. Different metrics for different protocols in DMS.



(a) Bitrate



(b) Delay



(c) Packetloss

Figure 7. Different metrics for different protocols in SDMS.

As future work, we would like to perform experiments with other parameters. We will consider the case of multiple flows between the communicating nodes and compare the performance of AODV and BATMAN protocols for linear and mesh topologies.

ACKNOWLEDGEMENTS

This work is supported by Japan Society for the Promotion of Science (JSPS). The authors would like to thank JSPS for the financial support.

REFERENCES

[1] A. Durresi, P.Zhang, M. Durresi, L. Barolli, "Architecture for Mobile Heterogeneous Multi Domain Networks", *Mobile Information Systems*, Vol. 6, No. 1, pp. 49-63, 2010.

[2] Gj. Mino, L. Barolli, F. Xhafa, A. Durresi, A. Koyama, "Implementation and Performance Evaluation of Two Fuzzy-based Handover Systems for Wireless Cellular Networks", *Mobile Information Systems*, Vol. 5, No. 4, pp. 339-361, 2009.

[3] S. S. Manvi, M. S. Kakkasageri, J. Pitt, "Multiagent Based Information Dissemination in Vehicular Ad Hoc Networks", *Mobile Information Systems*, Vol. 5, No. 4, pp. 363-389, 2009.

[4] V. Pham, E. Larsen, O. Kure, P. Engelstad, "Routing of Internal MANET Traffic over External Networks", *Mobile Information Systems*, Vol. 5, No. 3, pp. 291-311, 2009.

[5] A. Aikebaier, T. Enokido, M. Takizawa, "Design and Evaluation of Reliable Data Transmission Protocol in Wireless Sensor Networks", *Mobile Information Systems*, Vol. 4, No. 3, pp. 237-252, 2008.

[6] J. Goh, and D. Taniar, "Mining Frequency Pattern from Mobile Users", *Proc. of the 8th International Conference on Knowledge-Based Intelligent Information and Engineering Systems KES-2004*, Part III, Lecture Notes in Computer Science, Springer, Vol. 3215, pp. 795-801, 2004.

[7] D. Taniar, J. Goh, "On Mining Movement Pattern from Mobile Users", *International Journal of Distributed Sensor Networks*, Vol. 3. No. 1, pp. 69-86, 2007.

[8] K. Xuan, G. Zhao, D. Taniar, B. Srinivasan, "Continuous Range Search Query Processing in Mobile Navigation", *Proc of IEEE ICPADS-2008*, pp. 361-368, 2008.

[9] J. Haemi, M. Fiore, F. Filali, and C. Bonnet, "A Realistic Mobility Simulator for Vehicular Ad Hoc Networks", *EURECOM Technical Report*, Available at: http://www.eurecom.fr/util/publidownload.en.htm?id=1811, 2007.

[10] A. M. Hanashi, I. Awan, and M. Woodward, "Performance Evaluation with Different Mobility Models for Dynamic Probabilistic Flooding in MANETs", *Mobile Information Systems*, Vol 5, No. 1, pp. 65-80, 2009.

[11] M. Piorkowski, M. Raya, A. L. Lugo, M. Grossglauser, and J. P. Hubaux, "Joint Traffic and Network Simulator for VANETs", *Proc. of Mobile and Information Communication Systems Conference (MICS-2006)*, Available on line at: http://www.mics.ch/, 2006.

[12] T. Yang, G. De Marco, M. Ikeda, L. Barolli, "Impact of Radio Randomness on Performances of Lattice Wireless Sensors Networks Based on Event-reliability Concept", *International Journal of Mobile Information Systems*, Vol. 2, No. 4, pp. 211-227, 2006.

[13] H. Lundgren, E. Nordstrom, and C. Tschudin, Coping with Communication Gray Zones in IEEE 802.11b Based Ad hoc Networks., *Proc. of WoWMoM-2002*, pp. 49-55, 2002.

[14] The Network Simulator NS-2, Available on line at http://www.isi.edu/nsnam/.

[15] W. Kiess and M. Mauve, "A Survey on Real-world Implementations of Mobile Ad-hoc Networks", *Ad Hoc Networks*, Vol. 5, No. 3, pp. 324-339, 2007.

[16] C. Tschudin, P. Gunningberg, H. Lundgren, and E. Nordstrom, "Lessons from Experimental MANET Research, *Ad Hoc Networks*, Vol. 3, No. 2, pp. 221-233, 2005.

[17] V. Kawadia and P. Kumar, "Experimental Investigations into TCP Performance over Wireless Multihop Networks", *Proc. of E-WIND-2005, ACM SIGCOMM workshop on Experimental Approaches to Wireless Network Design and Analysis*, pp. 29-34, 2005.

[18] L. Barolli, M. Ikeda, G. De Marco, A. Durresi, and F. Xhafa, "Performance Analysis of OLSR and BATMAN Protocols Considering Link Quality Parameter", *Proc. of IEEE AINA-2009*, pp. 307-314, 2009.

[19] Core Software, AODV Software, Available on line at http://core.it.uu.se/core/index.php/ AODV-UU.

[20] Open Mesh, Available on line at http://www.open-mesh.org/wiki/.

[21] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", *RFC3561*, Nokia Research Center, University of California, University of Cincinnati, Technical Report, Available on line at http://www.ietf.org/rfc/rfc3561.txt, 2003.

[22] A. Neuman, C. Aichele, M. Lindner, and M. Wunderlich, "Better Approach To Mobile Ad-hoc Networking (B.A.T.M.A.N)", Work in Progress, Technical Repport, Available on line at http://www.ietf.org/id/draft-wunderlich-openmesh-manet-routing-00.txt, 2008.

[23] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based Ad Hoc Routing", *IEEE/ACM Transaction on Networking*, Vol. 14, No. 3, pp. 479-491, 2006.

[24] P. Larbig, RaLink RT2570 USB Enhanced Driver, Available on line at http://homepages.tu-darmstadt.de/∼p_larbig/wlan/.

[25] A. Dainotti, A. Botta, and A. Pescap'e, "Do You Know What You Are Generating?", *Proc. of ACM CoNEXT-2007*, pp. 1-2, 2007.

[26] M. Ikeda, L. Barolli, M. Hiyama, G. De Marco, T. Yang, and A. Durresi, "Performance Evaluation of Link Quality Extension in Multihop Wireless Mobile Ad-hoc Networks", *Proc. of International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2009)*, pp. 311-318, 2009.