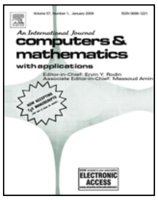
Computers and Mathematics with Applications 63 (2012) 339–349

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa



Performance comparison of OLSR and BATMAN routing protocols by a MANET testbed in stairs environment

Elis Kulla [a,*], Masahiro Hiyama [a], Makoto Ikeda [b], Leonard Barolli [b]

[a] Graduate School of Engineering, Fukuoka Institute of Technology (FIT), 3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811–0295, Japan

[b] Department of Information and Communication Engineering, Fukuoka Institute of Technology (FIT), 3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811–0295, Japan

ARTICLE INFO

Keywords:
MANET
Stairs environment
OLSR
BATMAN
Testbed

ABSTRACT

A Mobile Ad-hoc Network (MANET) is a set of mobile terminals, which move in different directions with different speeds. MANETs are found very useful in real applications such as time-lacking implementations and indoor environments. In this paper, we analyze the behavior of Optimized Link State Routing (OLSR) and Better Approach To Mobile Ad-hoc Network (BATMAN) routing protocols, in stairs environment, based on real data from our MANET testbed. We evaluate the performance considering three metrics: throughput, delay and packet loss. We design and implement two experimental scenarios: static stairs scenario and shifting stairs scenario. From our experiments, we show that as the number of hops increases to three or more hops, the performance of the communication decreases. The performance decreases further when the nodes are mobile. However, OLSR shows a better performance than BATMAN.

1. Introduction

A collection of wireless mobile hosts that can dynamically establish a temporary network without any aid from fixed infrastructure is known as a Mobile Ad-hoc Network (MANET). The mobile hosts act as routers for each other and they are connected via wireless links. Recently, MANETs are continuing to attract the attention for their applications in several fields, where the communication infrastructure is expensive and/or time consuming. Mobility and the absence of any fixed infrastructure make MANETs very appropriate for rescue operations and time-critical applications.

A lot of research for MANETs has been done in simulation, because in general, a simulator can give a quick and inexpensive evaluation of protocols and algorithms. However, experimentation in the real world is very important to verify the simulation results and to revise the models implemented in the simulator. A typical example of this approach has revealed many aspects of IEEE 802.11, like the gray-zones effect [1], which usually are not taken into account in standard simulators, as the well-known *ns*-2 simulator.

Many researchers performed valuable research in the area of wireless multi-hop networks by computer simulations and experiments [2,3]. Most of them are focused on throughput improvement, but they do not consider mobility [4].

In [5,6], the authors propose a dynamic probabilistic broadcasting scheme for mobile ad-hoc networks where nodes move according to different mobility models. Simulation results show that their approach outperforms the Fixed Probability Ad hoc On-demand Distance Vector (FP-AODV) and simple AODV in terms of saved rebroadcast under different mobility models. It also achieves higher saved rebroadcast and low collision as well as a low number of relays than the fixed probabilistic scheme and simple AODV.

---

[*] Corresponding author.
E-mail addresses: eliskulla@yahoo.com (E. Kulla), masahiro.hiyama@gmail.com (M. Hiyama), makoto.ikd@acm.org (M. Ikeda), barolli@fit.ac.jp (L. Barolli).

0898-1221/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2011.07.035

In [7], the authors implemented a multi-hop mesh network called Massachusetts Institute of Technology Roofnet, which consists of about 50 nodes. They consider the impact of node density and connectivity in the network performance. The authors show that the multi-hop link is better than the single-hop link in terms of throughput and connectivity. In [8], the authors analyze the performance of an outdoor ad-hoc network, but their study is limited to reactive protocols such as AODV [9] and Dynamic Source Routing (DSR) [10].

In [11], the authors perform outdoor experiments of non standard proactive protocols. Other ad-hoc experiments are limited to identify MAC problems, by providing insights on the one-hop MAC dynamics as shown in [12]. In [13], the disadvantage of using hysteresis routing metric is presented through simulation and indoor measurements.

In [14], the authors presents performance of OLSR using the standard hysteresis routing metric and the Expected Transmission Count (ETX) metric in a 7 by 7 grid of closely spaced Wi-Fi nodes to obtain more realistic results. The throughput results are effected by hop distance, and they are similar to our previous work [15].

So far, there are published many 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 simulation results, real-world experiments are needed and a lot of testbeds have been built to date [16]. The baseline criteria usually used in real-world experiments guarantee the repeatability of tests, i.e. if the system does not change along the experiments. The definition of a change in the system is a difficult task in MANET, especially because of node mobility.

We conducted many experiments with our MANET testbed, which are shown in our previous works [15,17]. We proved that while some of the Optimized Link State Routing (OLSR) problems can be solved, for instance the routing loop, this protocol still has the self-interference problem. There is an intricate inter-dependence between the MAC layer and the routing layer, which can lead the experimenter to misunderstand the results of the experiments. For example, the horizon is not caused only by IEEE 802.11 Distributed Coordination Function (DCF), but also by the routing protocol.

We carried out experiments with different routing protocols such as OLSR and Better Approach to Mobile Ad-hoc Networks (BATMAN) and 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. Moreover, we found that the node join and leave operations affect more the TCP throughput and Round Trip Time (RTT) than UDP [18]. In [19], we showed that BATMAN buffering feature showed a better performance than AODV, by handling the communication better when routes changed dynamically.

In this paper, we compare OLSR [20] and BATMAN [21] routing protocols in an indoor stairs environment. We designed and implemented a MANET testbed, consisting of five mobile machines. We implemented two experimental scenarios and investigated the performance of our MANET testbed considering throughput, delay and packet loss metrics.

The structure of the paper is as follows. In Section 2, we give an overview of OLSR and BATMAN routing protocols. In Section 3, we show the design and implementation of the MANET testbed. In Section 4, we discuss the experimental results and evaluate the performance of the testbed. Finally, we draw conclusions in Section 5.

2. Routing protocols

2.1. Overview of OLSR protocol

The link state routing protocol which is most popular today in the open source world is OLSR from [20]. OLSR with Link Quality (LQ) extension and fish-eye-algorithm works quite well. The OLSR protocol is a proactive routing protocol, which builds up a route for data transmission by maintaining a routing table inside every node of the network. The routing table is computed upon the knowledge of topology information, which is exchanged by means of Topology Control (TC) packets. The TC packets in turn are built after every node has filled its neighbor list. This list contains the identity of neighbor nodes. A node is considered as a neighbor if and only if it can be reached via a bi-directional link.

OLSR makes use of `HELLO` messages to find its one-hop neighbors and its two-hop neighbors through their responses. The sender can then select its Multi Point Relays (MPR) based on the one-hop nodes which offer the best routes to the two-hop nodes. In this way, the amount of control traffic can be reduced. Each node has also an MPR selector set which enumerates nodes that have selected it as an MPR node. OLSR uses TC messages along with MPR forwarding to disseminate neighbor information throughout the network. OLSR checks the symmetry of neighbor nodes by means of a 4-way handshake based on `HELLO` messages. This handshake is inherently used to compute the packet loss probability over a certain link. This can sound odd, because packet loss is generally computed at higher layers than routing one. However, an estimate of the packet loss is needed by OLSR in order to assign a weight or a state to every link. Host Network Address (HNA) messages are used by OLSR to disseminate network route advertisements in the same way that TC messages advertise host routes.

In our OLSR code, a simple RFC-compliant heuristic is used [22] to compute the MPR nodes. Every node computes the path toward a destination by means of a simple shortest-path algorithm, with hop-count as target metric. In this way, a shortest path can result to be also not good, from the point of view of the packet error rate. Accordingly, recently *olsrd* has been equipped with the LQ extension, which is a shortest-path algorithm with the average of the packet error rate as metric. This metric is commonly called as the ETX, which is defined as $\mathrm{ETX}(i) = 1/(\mathrm{NI}(i) \times \mathrm{LQI}(i))$. Given a sampling window W, $\mathrm{NI}(i)$ is the packet arrival rate seen by a node on the i-th link during W. Similarly, $\mathrm{LQI}(i)$ is the estimation of the packet arrival rate seen by the neighbor node which uses the i-th link. When the link has a low packet error rate, the ETX metric is higher. The LQ extension greatly enhances the packet delivery ratio with respect to the hysteresis-based technique [23].

2.2. Overview of BATMAN protocol

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 problems. 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 Time To Live (TTL) value and an increasing Sequence Number (SQ).
2. Checking of 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 BATMAN uses a timer for sending OGMs and SQ (OGM) for checking the bi-directionality of links. If the SQ of an OGM received from a particular node falls within a certain range, the corresponding link is considered to be 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 the last received OGM from B has an SQ less than or equal to Tr. If it does, then B is considered to be bi-directional, otherwise it is considered to be unidirectional. Bi-directional links are used for the ranking procedure. The quantity Tr is called the bi-directional sequence number range. The ranking procedure is a substitute of 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 the 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 ranking 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 (the 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 [24].

In BATMAN, every node re-broadcasts received OGMs only once, and only those OGMs, which have been received via the best-ranked neighbor. This is a kind of selective flooding, which practically reduces the overhead of the flooding. Another analogy can be found in gossip protocols [24]. In gossip protocol, every node decides to re-broadcast received data with some probability p. This is equivalent to eliminating some links in the network and then supposing that every node re-broadcast with probability 1. In gossip protocol, there is a threshold for p and the density of nodes after which the success ratio[1] is almost surely 1. In BATMAN, the probability p is changed according to the ranking procedure. It is the probability that an OGM is reached via the best rank neighbor. The expression of this probability is left for further analysis. Let us note that the selective flooding eliminates possible misbehavior of the ranking procedure. In fact, cumulative count of the OGM could be greater than the total number of OGM received via the current best neighbor.

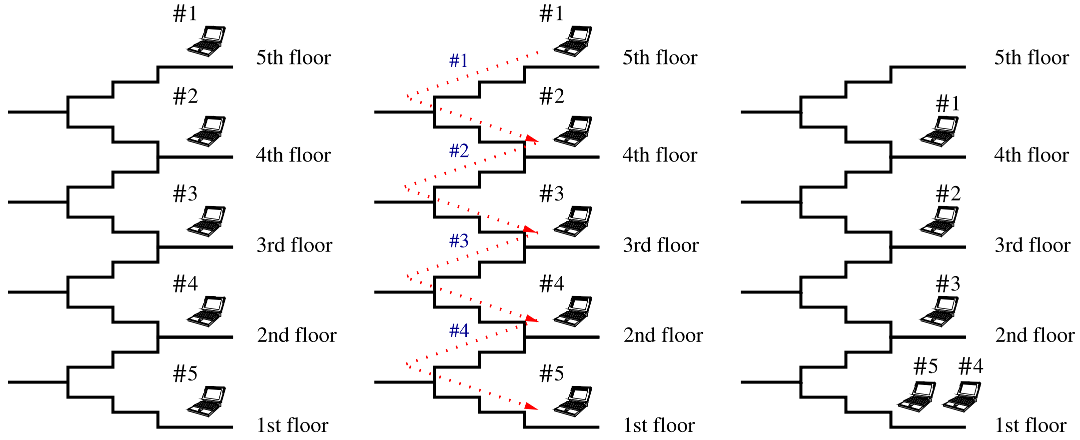
3. Design and implementation of MANET testbed

3.1. Testbed description

We conducted our experiments in the stairs environment of our five-floor academic building. The testbed consists of five laptops. We modeled two experimental scenarios. In Fig. 1(a), all nodes are stationary. We call this, Stairs Static scenario and will refer to it as STAS scenario. In Fig. 1(b), only node #5 is stationary. The other nodes move one floor down and replace each other. The final topology (positions) of the nodes is shown in Fig. 1(c). We will call this, Stairs Shift scenario and will refer to it as SHIS scenario. In Table 1, we show the status of nodes for each experimental scenario. A snapshot of each node during experiments in STAS scenario is shown in Fig. 2. In Table 2, we show the parameters used to perform our experiments.

---

[1] That is the probability that broadcast messages are received by all nodes in the network.



(a) STAS scenario. (b) Node movement. (c) Final position.

Fig. 1. SHIS scenario.

(a) Node ID #1 on 5th floor. (b) Node ID #2 on 4th floor. (c) Node ID #3 on 3rd floor.

(d) Node ID #4 on 2nd floor. (e) Node ID #5 on 1st floor.

Fig. 2. Snapshots of testbed nodes.

Table 1
Node's pattern for experimental scenarios.

Scenario	Moving nodes		Static nodes
	Source node	Relay node	
STAS	0	0	5
SHIS	1	3	1

The machines operate on Ubuntu 9.04 OS with kernel 2.6.28 suitably modified in order to support the wireless cards. The linksys wireless network cards (Model: WUSB54G ver. 4) are usb-based cards with an external antenna of 2 dBi gain, transmitting power of 16 + / − 1 dBm and receiving sensitivity of −80 dBm. We verified that the external antenna improves the quality of the first-hop link, which is the link connecting the ad-hoc network. The driver can be downloaded from the web site [25,26].

In our testbed, we have two systematic traffic sources we could not eliminate: the control traffic and the other wireless Access Points (APs) interspersed within the campus. The control traffic is due to the `ssh` program, which is used to remotely start and control the measurement software on the source node. The other traffic source brings interferences occupying the

Table 2
Experimental parameters.

Parameters	Values
Number of nodes	5
MAC	IEEE 802.11b/Channel 2
Transmitted power	16 + / − 1 dBm
Flow type	CBR
Packet rate	200 pps
Packet size	256 bytes
Number of trials	10
Duration	150 s
Routing protocol	OLSR, BATMAN

available bandwidth, which is typical in an academic scenario. Thus we changed our wireless cards to transmit at Channel 2 (2.417 GHz).

The traffic in the network is sent by Distributed Internet Traffic Generator (D-ITG) software, which is an Open Source Traffic Generator [27]. With D-ITG, we can inject different types of data flows in the network. After finishing the transmission, D-ITG offers decoding tools to get information about different metrics along the whole transmission duration.

3.2. *Testbed interface*

Upon the first implementation of the testbed, all the parameter settings and editing were done by using command lines of bash shell (terminal), which resulted in many misprints and the experiments were repeated many times. In order to make the experiments easier, we implemented a Graphical User Interface (GUI). We used 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.

We implemented many parameters in the interface such as transmission duration, number of trials, source address, destination address, packet rate, packet size and topology setting function. We can save the data for these parameters in a text file and can manage in a better way the experimental conditions. Moreover, we implemented collection function of experimental data in order to make easier the experimenter's work.

3.3. *Experimental environment*

Our testbed provides an experimental platform for evaluating protocols and algorithms using realistic parameters. In this testbed, we can implement different topology scenarios and analyze different routing protocols considering different metrics. In this work, we take the following considerations.

The experiments are conducted in an indoor stairs environment inside our five-floor departmental building. We analyzed our network for two experimental scenarios: STAS scenario, where all nodes are stationary and SHIS scenario, where only one node is stationary. In SHIS scenario, the mobile nodes move at regular speed toward their final destination and when they arrive at corners, they stop for about three seconds. The shifting time is about five seconds. We discuss the effect of multihop and mobility regarding throughput, delay and packet loss using OLSR and BATMAN routing protocols.

4. Experimental results

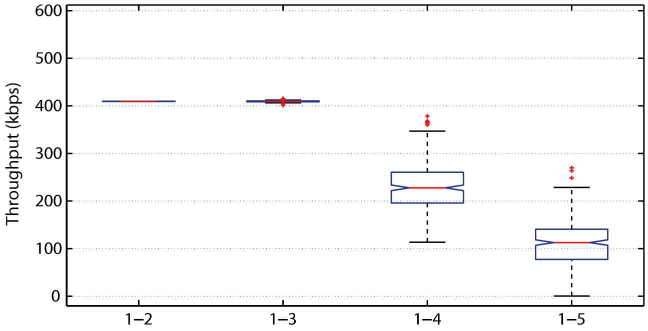
4.1. *Experimental settings*

The experimental parameters are shown in Table 2. We study the impact of best-effort traffic for Mesh Topology (MT). In the MT scheme, the MAC filtering routines are not enabled. We collected data for three metrics: throughput, delay and packet loss. These data are collected using D-ITG.

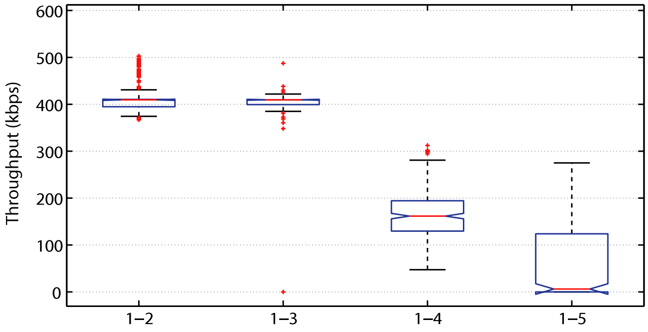
The data sent to conduct the experiments was of CBR type over UDP. The packet rate of CBR was 200 pps with each packet of size 256 bytes/p, totaling a Data Transmission Rate (DTR) of 200 pps × 256 bytes/packet × 8 bit/bytes = 409 600 bps = 409.6 kbps.

We measured throughput, delay and packet loss of UDP, which are computed at the receiver. We display the measured data by the box and whisker plot of the metrics. Box and whisker plot is a convenient way to show groups of numerical data by lower quartile, median, upper quartile, and the outliers. In the plot, the bottom and top of the box are 25th and 75th percentiles, and the band near the middle of the box is the median. The end of the whiskers can represent the lowest datum which is still within 1.5 inter-quartile range of the lower quartile, and the highest datum which is still within 1.5 inter-quartile range of the upper quartile. We used boxplot display to show the results in order to give a better understanding of the communication oscillations. We also showed the average values of every flow, in order to get an understanding of the communication in general terms.

As MAC protocol, we used IEEE 802.11b. The transmission power was set in order to guarantee a coverage radius equal to one hop physical distance. Since we were interested mainly in the performance of the routing protocol, we kept unchanged



(a) STAS scenario (OLSR).



(b) SHIS scenario (OLSR).

Fig. 3. Throughput results OLSR.

Table 3
Average throughput (kb/s).

Flow	Floor	STAS scenario		SHIS scenario	
		OLSR	BATMAN	OLSR	BATMAN
#1 → #2	4th	409.6000	409.6000	409.5904	409.2450
#1 → #3	3rd	409.5304	407.2257	391.7681	345.1153
#1 → #4	2nd	232.4234	184.5043	166.2921	132.5324
#1 → #5	1st	112.7911	76.7847	63.4341	146.5412

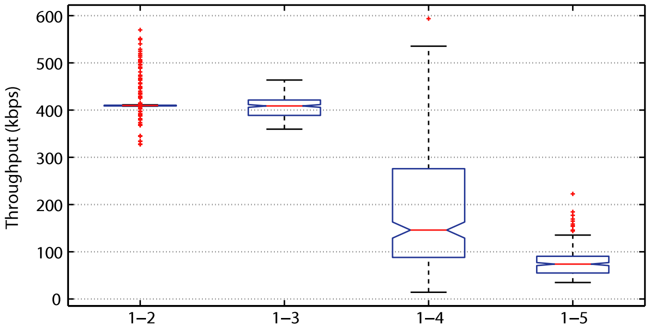
all MAC parameters, such as the carrier sense, the retransmission counter, the contention window and the RTS/CTS threshold. Moreover, the channel central frequency was set to 2.417 GHz (channel 2). In regard to the interference, it is worth noting that, during our tests, almost all the IEEE 802.11 spectrum had been used by other APs disseminated within the campus.

4.2. *Experimental analysis*

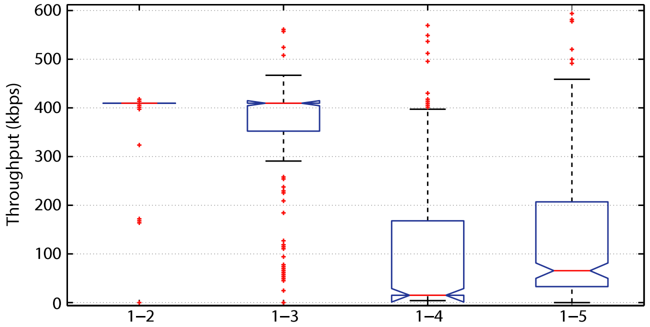
We show the throughput results in Figs. 3 and 4 and Table 3. In Figs. 3(a) and 4(a), we show the throughput results for STAS scenario. For communication flow from node #1 to node #2 and node #3, the throughput has only a few oscillations but its average value is very close to DTR value (409.6 kbps) for both protocols. When the data flows from node #1 to node #4 or node #5 in STAS scenario the oscillations are increased noticeably. The average throughput value for the communication from node #1 to node #4 decreases to almost 50% lower than DTR and when data flows from node #1 to node #5, the throughput reaches values around 100 kbps.

Also in SHIS scenario, for three-hop and four-hop communications we notice oscillations (see Figs. 3(b) and 4(b)). When we use BATMAN, two-hop communications present more oscillations than in the case of OLSR. Thus, the average values for OLSR and BATMAN decrease with 5% and 15% respectively, when the communication occurs in two or more hops. For the average values of each flow, see Table 3.

In SHIS scenario, when transmitting from node #1 to node #5, we notice that the throughput for BATMAN increases in comparison to #1 → #4 transmission. As shown in Fig. 1(c), the destination node (#5) is in static state during experiment time for SHIS scenario. Also, at the final position of SHIS scenario, the nodes are closer to each other and the links have better quality, which results in better performance.



(a) STAS scenario (BATMAN).



(b) SHIS scenario (BATMAN).

Fig. 4. Throughput results BATMAN.

Table 4
Average delay (s).

Flow	Floor	STAS scenario		SHIS scenario	
		OLSR	BATMAN	OLSR	BATMAN
#1 → #2	4th	0.0045	0.0198	0.0209	0.0241
#1 → #3	3rd	0.0210	0.0885	0.4754	0.7207
#1 → #4	2nd	1.3185	2.6800	2.6742	3.6724
#1 → #5	1st	1.6403	4.0517	3.5424	4.4339

Table 5
Average packet loss (pps).

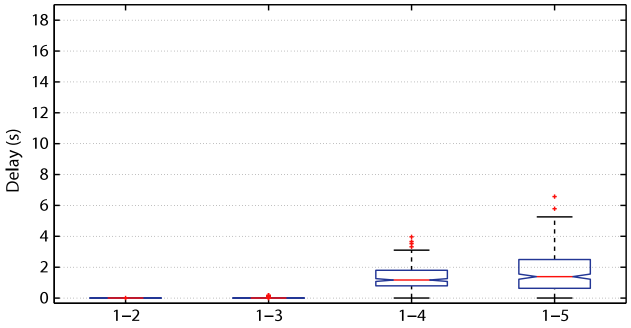
Flow	Floor	STAS scenario		SHIS scenario	
		OLSR	BATMAN	OLSR	BATMAN
#1 → #2	4th	0.0063	0	0.0133	0.0867
#1 → #3	3rd	0.0170	0.4703	3.5033	10.7967
#1 → #4	2nd	18.5473	17.5783	29.8418	23.0733
#1 → #5	1st	35.0030	38.7510	25.5562	18.8267

The results for delay are shown in Figs. 5 and 6. In STAS scenario, the delay values for both protocols increase constantly when the number of hops increases. When we use BATMAN as a routing protocol, for the #1 → #5 flow, we notice great oscillations. However, the average value in this case is 4.05 s; see Table 4.

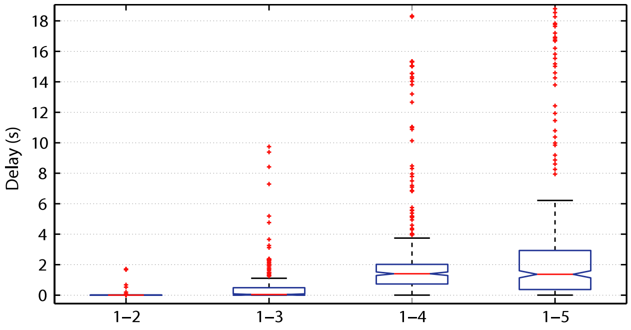
For SHIS scenario, the oscillations of delay start to become noticeable since the two-hop communication, for both OLSR and BATMAN. The average values are more than 2.5 s for three or four hops. However, OLSR shows a better performance than BATMAN regarding the delay metrics.

The experimental results for packet loss are shown in Figs. 7 and 8. In Figs. 7(a) and 8(a), we show the boxplots of packet loss values for STAS scenario. In one-hop and two-hop communications, the packet loss values have no oscillations and the testbed shows good performance. As shown in Table 5, the average value for packet loss in these cases do not exceed the value of 0.47 pps. In the case when node #1 transmits to node #4 or node #5, the packet loss shows oscillations and the average values are 17–39 pps (see Table 5).

In Figs. 7(b) and 8(c), we show the results of packet loss for SHIS scenario. In comparison to results of STAS scenario, the average values of packet loss are greater, as shown in Table 5. This is due to movement of nodes, which brings frequent route

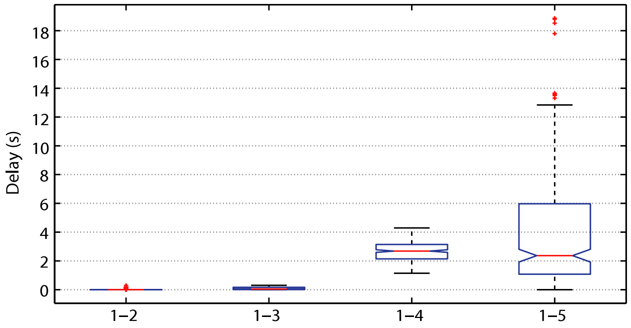


(a) STAS scenario (OLSR).

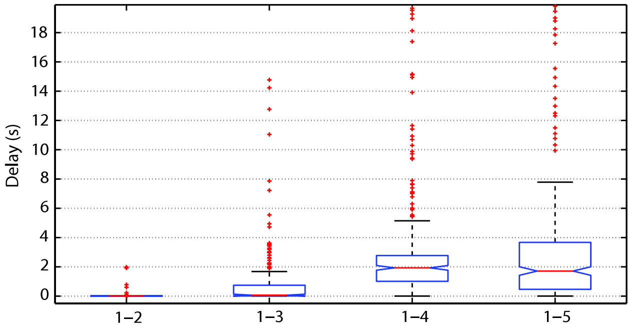


(b) SHIS scenario (OLSR).

Fig. 5. Delay results OLSR.

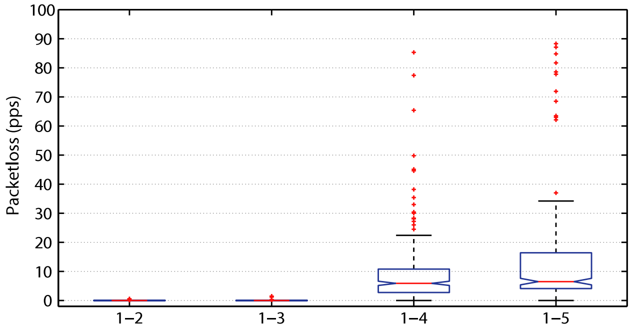


(a) STAS scenario (BATMAN).

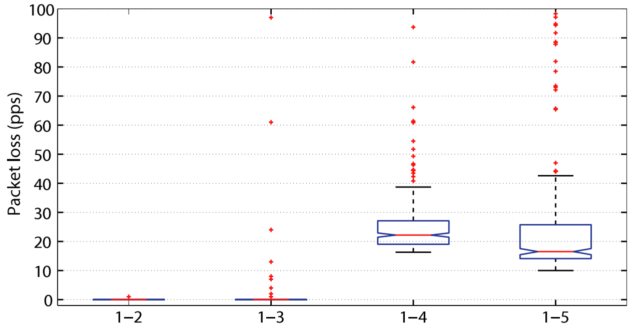


(b) SHIS scenario (BATMAN).

Fig. 6. Delay results BATMAN.

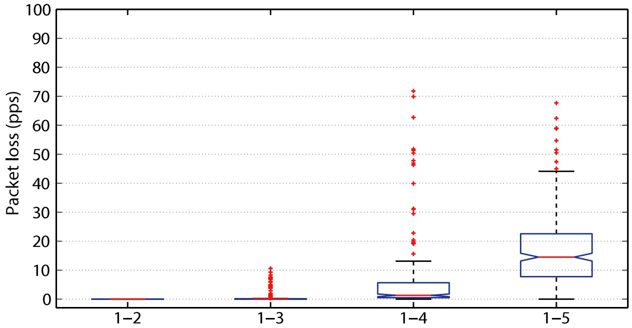


(a) STAS scenario (OLSR).

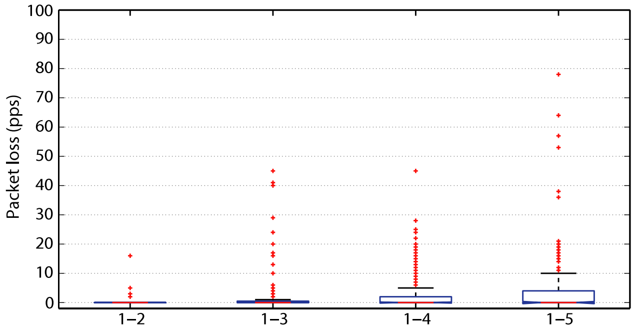


(b) SHIS scenario (OLSR).

Fig. 7. Packet loss results OLSR.



(a) STAS scenario (BATMAN).



(b) SHIS scenario (BATMAN).

Fig. 8. Packet loss results.

changes and instabilities for both OLSR and BATMAN protocols. Moreover, when nodes shift positions, they stay at almost the same position for a certain amount of time (more than five seconds). Based on this fact the routes are uncertain and they change frequently during this period of time.

However, when the destination node is node #5 there is an improvement in performance for both OLSR and BATMAN protocols. When the destination node is static (#1 → #5 flow), the testbed shows better results than when destination is moving (#1 → #4 flow), even though the hop distance is greater.

In general, OLSR shows a better performance than BATMAN. However, BATMAN has an extra feature. BATMAN do not drop packets directly, when routes are unstable, but keeps them in a buffer for a certain amount of time. For SHIS scenario, when transmitting from node #1 to node #5, BATMAN performance is increased related to packet loss and throughput metrics, but delay is still increasing.

5. Conclusions

In this paper we designed and implemented an indoor MANET stairs testbed, in which we implemented two scenarios: STAS scenario, in which all nodes are static and SHIS scenario, in which only node #5 is static. We used throughput, delay and packet loss as metrics for performance assessment of two proactive routing protocols, OLSR and BATMAN.

From our analysis, we found the following results:

- Both protocols show a good performance in both STAS and SHIS scenario, for one-hop and two-hop communications.
- For 3 or 4 hops from source (node #4 and #5, respectively), the throughput decreased with 50% compared with DTR, regarding STAS scenario.
- In the case of SHIS scenario, when destination was node #4 or node #5, the throughput decreased about 60% compared with DTR.
- The delay and packet loss increased after three hops for STAS scenario and after two hops for SHIS scenario.
- In general, OLSR protocol showed better performance than BATMAN protocol.
- BATMAN showed a better performance than OLSR for packet loss metrics. BATMAN buffers the packets when routes are unstable. However, delay still increases.
- Regarding #1 → #5 flow, in SHIS scenario, source and destination nodes were closer to each other so the packet loss was lower compared with #1 → #5 in STAS scenario.

In our future works, we would like to consider different parameters of routing protocols. Moreover, we would like to evaluate different scenarios in our vertical stairs environment and compare them with simulation data.

Acknowledgment

This work is supported by a Grant-in-Aid for scientific research of Japan Society for the Promotion of Science (JSPS). The authors would like to thank JSPS for the financial support.

References

[1] H. Lundgren, E. Nordström, C. Tschudin, Coping with communication gray zones in IEEE 802.11b based ad-hoc networks, in: Proc. of International Workshop on Wireless Mobile Multimedia, WOWMOM-2002/MOBICOM-2002, 2002, pp. 49–55.
[2] E. Nordström, APE-a large scale ad-hoc network testbed for reproducible performance tests, Master's Thesis, Uppsala University, 2002.
[3] Y. Owada, Y. Takahashi, T. Suda, H. Terui, F. Taki, T. Yagi, K. Mase, A large scale wireless mobile ad-hoc network testbed, in: Proc. of IEEE Vehicular Technology Conference, September 2005, pp. 324–328.
[4] R. Draves, J. Padhye, B. Zill, Comparison of routing metrics for static multi-hop wireless networks, in: Proc. of International Conference on Special Interest Group on Data Communication, SIGCOMM-2004, 2004, pp. 133–144.
[5] A.M. Hanashi, A. Siddique, I. Awan, M. Woodward, Performance evaluation of dynamic probablilistic flooding under different mobility models in MANETs, in: Proc. of International Conference on Parallel and Distributed Systems, ICPADS-2007, vol. 2, December 2007, pp. 1–6.
[6] A.M. Hanashi, I. Awan, M. Woodward, Performance evaluation with different mobility models for dynamic probabilistic flooding in MANETs, International Journal of Mobile Information Systems, IJMIS 5 (1) (2009) 65–80.
[7] J. Bicket, D. Aguayo, S. Biswas, R. Morris, Architecture and evaluation of an unplanned 802.11b mesh network, in: Proc. of International Conference on Mobile Computing and Networking, MOBICOM-2005, pp. 31–42, 2005.
[8] D.A. Maltz, J. Broch, D.B. Johnson, Lessons from a full-scale multihop wireless ad-hoc network testbed, Journal on Personal Communications 8 (1) (2001) 8–15.
[9] C. Perkins, E. Belding-Royer, S. Das, Ad-hoc on-demand distance vector (AODV) routing, in: RFC 3561, Experimental, July, 2003.
[10] D.B. Johnson, D.A. Maltz, J. Broch, DSR: the dynamic source routing protocol for multi-hop wireless ad-hoc networks, Journal on Ad-Hoc Networks (2001) 139–172. (Chapter 5).
[11] R.S. Gray, D. Kotz, C. Newport, N. Dubrovsky, A. Fiske, J. Liu, C. Masone, S. McGrath, Y. Yuan, Outdoor experimental comparison of four ad-hoc routing algorithms, in: Proc. of Modeling, Analysis and Simulation of Wireless Mobile Systems, MSWiM-2004, 2004, pp. 220–229.
[12] E. Borgia, M. Conti, E. Gregori, IEEE 802.11b ad-hoc networks: performance measurements, Journal of Cluster Computing 8 (2–3) (2005) 135–145.
[13] T. Clausen, G. Hansen, L. Christensen, G. Behrmann, The optimized link state routing protocol, evaluation through experiments and simulation, in: Proc. of IEEE Symposium on Wireless Personal Mobile Communications, September 2001. Available online at: http://hipercom.inria.fr/olsr/wpmc01.ps.
[14] D. Johnson, G. Hancke, Comparison of two routing metrics in OLSR on a grid based mesh network, Journal on Ad-Hoc Networks 7 (2) (2009) 374–387.
[15] G. De Marco, M. Ikeda, T. Yang, L. Barolli, Experimental performance evaluation of a pro-active ad-hoc routing protocol in outdoor and indoor scenarios, in: Proc. of International Conference on Advanced Information Networking and Applications, AINA-2007, May 2007, pp. 7–14.
[16] W. Kiess, M. Mauve, A survey on real-world implementations of mobile ad-hoc networks, Journal on Ad-Hoc Networks 5 (3) (2007) 324–339.

[17] L. Barolli, M. Ikeda, G. De Marco, A. Durresi, F. Xhafa, Performance analysis of OLSR and BATMAN protocols considering link quality parameter, in: Proc. of International Conference on Advanced Information Networking and Applications, AINA-2009, May 2009, pp. 307–314.
[18] M. Ikeda, L. Barolli, M. Hiyama, T. Yang, G. De Marco, A. Durresi, Performance evaluation of a MANET testbed for differenet topologies, in: Proc. of International Conference on Network-Based Information Systems, NBiS-2009, Indianapolis, August 2009, pp. 327–334.
[19] E. Kulla, M. Ikeda, L. Barolli, R. Miho, Impact of source and destination movement on MANET performance considering BATMAN and AODV protocols, in: Proc. of International Conference on Broadband, Wireless Computing, Communication and Applications, BWCCA-2010, 2010, pp. 94–101.
[20] A. Tonnesen, OLSRd: implementation code of the OLSR. Available on line at: http://www.olsr.org/.
[21] Open mesh webpage for a collection of tools to build free and open mesh networks. Available online at: http://www.open-mesh.org/.
[22] T. Clausen, P. Jacquet, Optimized link state routing protocol (OLSR), in: RFC 3626, Experimental, 2003.
[23] D.S.J.D. Couto, D. Aguayo, J. Bicket, R. Morris, A high throughput path metric for multi-hop wireless routing, in: Proc. of MobiCom-2003, 2003, pp. 134–146.
[24] Z.J. Haas, J.Y. Halpern, L. Li, Gossip-based ad-hoc routing, IEEE/ACM Transactions on Networking 14 (3) (2006) 479–491.
[25] Rt2x00 Project for Linux Wireless Cards Drivers. Available online at: http://rt2x00.serialmonkey.com/.
[26] Ralink RT2570 USB Enhanced Driver. Available online at: http://homepages.tu-darmstadt.de/p_larbig/wlan/.
[27] A. Dainotti, A. Botta, A. Pescap, Do you know what you are generating? in: Proc. of International Conference on Emerging Networking Experiments and Technologies, CoNEXT-2007, New York, NY, USA, 2007, pp. 1–2.