Design of an Adhoc Testbed for IoT and WSAN Applications using Raspberry Pi

Hiroya Oda, Elis Kulla, Ryo Ozaki, Noritaka Nishihara

Abstract With recent advancement in wireless sensor technologies, IoT and WSAN applications have emerged and new algorithms and protocols have been proposed. Many simulations have been conducted to test these new protocols and algorithms. However in order to verify the new protocols and algorithms in real experiments, we designed an adhoc testbed, where we plan to conduct experiments for IoT and WSAN applications. In this paper we describe our testbed and show some results, while investigating the performance of the network for different packet sizes.

1 Introduction

In the next generation of wireless communication systems, such as 5G [1, 2], the need for rapid deployment of independent mobile users will increase [3]. Significant examples include establishing networks which are survivable, efficient and able to communicate dynamically for emergency/rescue operations, sensing in Smart Cities [4], Internet of Things (IoT) [5] and so on. Such networks cannot rely on centralized management, but rather distributed and autonomous operation. In general they can be considered as applications of Adhoc Networks, or if nodes are mobile, known as Mobile Adhoc Networks (MANETs).

A MANET is a collection of wireless mobile hosts that can dynamically establish a temporary network without any aid from fixed infrastructure. The mobile hosts act

Hiroya Oda
Okayama University of Science, Department of Information ad Computer Engineering, 1-1 Ridai-cho, Kita-ku, 700-0005 Okayama, Japan
e-mail: t16jm01oh@ous.jp

Elis Kulla, Ryo Ozaki, Noritaka Nisihara
Okayama University of Science, Department of Information ad Computer Engineering, 1-1 Ridai-cho, Kita-ku, 700-0005 Okayama, Japan
e-mail: kulla@ice.ous.ac.jp, ozaki@ice.ous.ac.jp, nisihara@ice.ous.ac.jp

L. Barolli et al. (eds.), *Advances on Broad-Band Wireless Computing, Communication and Applications*, Lecture Notes on Data Engineering and Communications Technologies 2, DOI 10.1007/978-3-319-49106-6_53

as routers for each other and they are connected via wireless links. Mobility and the absence of any fixed infrastructure make MANET very attractive for rescue operations and time-critical applications. Wireless Sensor Networks (WSN), Wireless Sensor Actor Networks (WSAN), Vehicular Ad-hoc Networks (VANET) and other new and existing network technologies are based on MANETs. Thus, in order to expand the usage of MANET, extensive research and verification are required.

A lot of research for MANETs is going on, usually in simulations, because in general, a simulator can give a quick and inexpensive evaluation of protocols and algorithms. However, experimentations in the real world are very important to verify the simulation results and to revise the models implemented in the simulator.

In our paper, we show the design and implementation of a testbed for MANET, where we can test and verify different algorithms. By adding Raspberry Pi devices to the testbed, the network becomes heterogeneous. Raspberry Pi can be converted in a sensor node by connecting sensors to its 40-pin GPIO. On the other hand, it can be used to control different robots (we have implemented for RAPIRO [6]), and because Raspberry Pi is portable, our testbed network has mobile sensors and mobile actors, which makes it suitable to implement WSN, IoT and WSAN applications in it.

Moreover, we conduct some experiments in order to analyze the performance of BATMAN routing protocol in a mixed outdoor and indoor environment. We implement simple scenarios in our adhoc testbed, which consists of 10 mobile machines (Note PCs and Raspberry Pi). We investigate the network performance considering throughput and delay, by sending data in different format.

The structure of the paper is as follows. In Section 2, some related works are discussed. An overview of BATMAN routing protocol is given in Section 3. In Section 4, we describe the design and implementation of our adhoc testbed. We discuss the experimental results and evaluate the performance of the testbed, in Section 5. Finally, we draw conclusions in Section 6.

2 Related Work

The authors in [7, 8, 9, 10] conducted many experiments with their MANET testbed. They carried out the experiments with different routing protocols such as OLSR and Better Approach to Mobile Ad-hoc Networks (BATMAN) [11] and found that throughput of TCP was improved by reducing Link Quality Window Size (LQWS), but there were packet loss because of experimental environment and traffic interference. Moreover, they found that the node join and leave operations affect more the TCP throughput and Round Trip Time (RTT) than UDP [9]. In [10], they showed that BATMAN buffering feature showed a better performance than Ad-hoc On-demand Distance Vector (AODV), by handling the communication better when routes changed dynamically.

In [12], the authors implemented multi-hop mesh network called Massachusetts Institute of Technology (MIT) 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 single-hop link in terms of throughput and connectivity. In [13], the authors analyze the performance of an outdoor ad-hoc network, but their study is limited to reactive protocols such as AODV [14] and Dynamic Source Routing (DSR) [15].

The authors of [16] compare the performance of two typical routing protocols, AODV and DSR, in real multi-hop environment. Apart from testing the end-to-end packet loss, delay and routing path parameters, they also assess the performance of AODV and DSR in terms of some applications based on IOT, such as Radio Frequency Identification (RFID) service, voice service and temperature monitoring service.

The paper in [17] proposes an original solution to integrate and exploit MANET overlays, collaboratively formed over WSNs, to boost urban data harvesting in IoT.

3 Overview of BATMAN Routing Protocol

In the well-known OLSR, there was 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 Time To Live (TTL) value and an increasing Sequence Number (SQ).
2. Checking of the best one-hop neighbor for every 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, which are used by BATMAN nodes to create routes for all the nodes in the network. 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, the receiving node executes the following steps.

1. If the sequence number of the OGM (SQ(OGM)) is less than the corresponding NBRF, then drop the packet.
2. Otherwise, update the NBRF=SQ (OGM) in the ranking table.

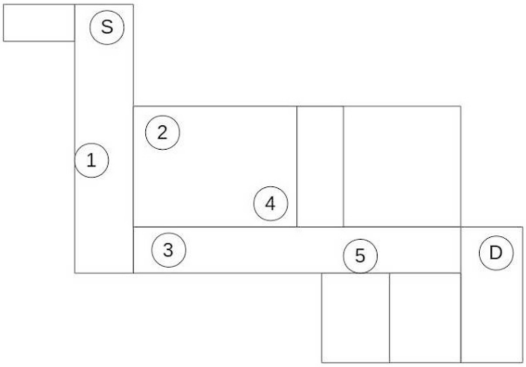


Fig. 1 Experimental environment.

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.

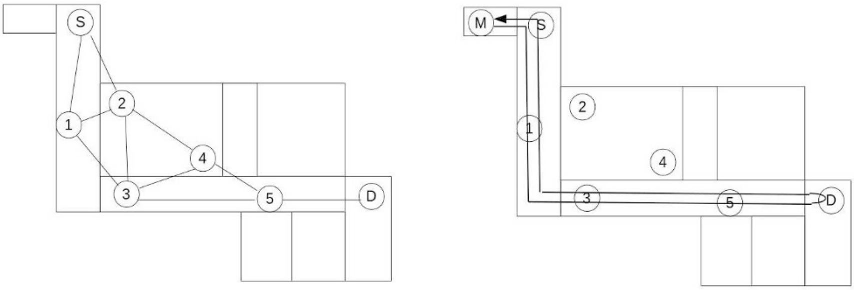
Finally, the ranking procedures select the best one-hop neighbor, the neighbor which has the highest rank in the ranking table. Let us note that the same OGM packet is used for: link sensing, neighbor discovery, bi-directional link validation and flooding mechanism. Other details on BATMAN can be found in [11].

4 Design and Implementation of MANET Testbed

4.1 Testbed Description

We implemented our testbed in our academic environment around our five-floor academic building, as shown in Fig. 1. The experiments were conducted in a mixed environment, where nodes are put inside rooms or in the outside compartments (stairs etc.) of Building 18 of Okayama University of Science campus. We used six Note PCs (Panasonic CF-T7 Let's Note model) equipped with external USB wireless cards (BUFFALO WLI-UC-GNM LAN Adapter) [18] and two Raspberry Pi 2 model B devices equipped with the same wireless cards.

The Note PC machines operate on UBUNTU 14.04 LTS OS [19], with kernel 3.16.0-30, while Raspbery Pi machines operate on the native Raspbian Jessie OS [20]. We set up their wireless cards to operate with transmitting power of 16+/-1dBm and receiving sensitivity of -80dBm.



(a) Static (b) Mobile

Fig. 2 Static and Mobile Scenarios.

The traffic in the network is sent by Distributed Internet Traffic Generator (D-ITG) software, version 2.7.0 Beta2, which is an Open Source Traffic Generator [21]. With D-ITG, we can inject different type 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.

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.

4.2 Experimental Scenario Settings

In order to evaluate our testbed, we implemented two physical scenarios, based on node position and movement, and different cases, based on the type of data sent in the network.

Two scenarios can be seen in Fig. 2. In Static scenario, all nodes are static. S and D are Raspberry Pi devices, and we think of them as sensor and actor nodes, respectively. When testing the settings of the testbed, we noticed that some packets were sent directly from S to D or by using only two hops. But we wanted our network to use multihop routes. Therefore, we setup MAC filtering in order to limit the connectivity distance, as shown in Fig. 2(a). The results of Static scenario will be used as basis when compared to Mobile scenario. In Mobile scenario, we introduce a mobile node M, which does not use filtering (it can connect to every node in the network) and moves from the location of node S to location of node D and back to node S in 3 minutes, as shown in Table 1. Mobility of node M, brings topology changes during movement, so it makes the network more dynamic.

Table 1 Node M Mobility pattern

Time	Action	Description
$0s - 30s$	Warm-Up	Node M needs some time to enter the network by sending and receiving OGMs.
$30s - 90s$	Move1	Moving from node S to D.
$90s$	Turn	Turn at the location of node D.
$90s - 150s$	Move2	Moving from node D to node S.
$150s - 180s$	Cool-Down	In order to match the Warm-up period.

4.3 Considerations

An experimental environment gives us realistic results and can be used to test and verify real aspects of MANETs and its applications. However, our experiments are based on some simple considerations, in order to make the results consistent and usable by other researchers.

- We analyze the data based on two metrics: Throughput and Round-trip delay.
- Moving nodes are moved by human force. We carry laptops, while walking in the experimental area. When reaching turning points, we stop for about 3 seconds before resuming movement.
- We run the experiments 10 times for every setting. This is important, so we can get an average measure of network performance.
- We analyze the effect of multihop communication and mobility in real networks, and we try to implement scenarios with a high degree of similarity with realistic applications.

In our testbed, we have a systematic traffic sources we could not eliminate. There are other wireless Access Points (APs) interspersed within the campus. This brings interferences occupying the available bandwidth, which is typical in an academic scenario.

Table 2 Data Types

Parameter	Case1	Case2	Case3	Case4
Packet Size ($Bytes$)	600	900	1200	1500
Packet Rate (pps)	104	69	52	42
Sent Throughput ($kbps$)	500	500	500	500

5 Experimental Results

We sent data through the network, from node S to node D, in four different patterns and evaluated the performance of the transmission based on two metrics: Throughput and Delay. We wanted to investigate the effect of packetsize in multihop communications, so we differ the packetsize and packetrate of the data, and keep the throughput constant at 500kbps ((See Table. 2)). The results are shown in average values, in Fig. 3 and in time-domain representation in Figs. 4 and 5.

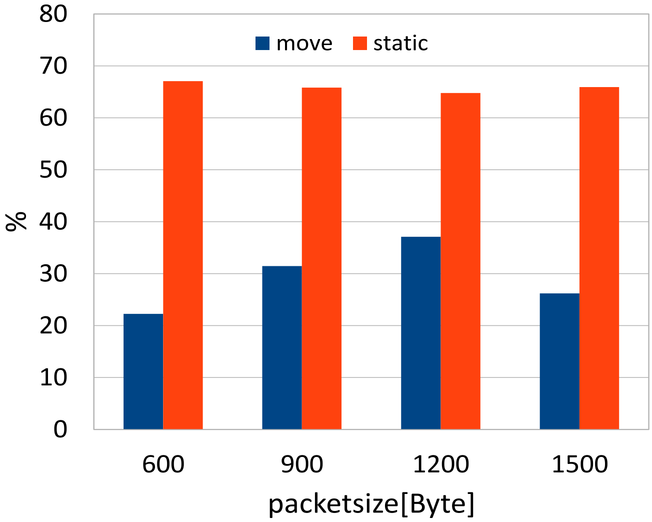
We can see from the average values of throughput (Fig. 3(a)), that for Static scenario, throughput is more than 60% and similar in all cases, which is an acceptable value for wireless multihop communication. However, in Mobile scenario, the throughput rate drops to less than 40%. In this scenario, we also notice the effect of packetsize in the performace. Greater packetsizes have smaller packetrate, so the network is not congested and vice-versa. In cases with frequent route changes, congestion lowers the performance, because it means more possible lost packets. But in Case4, where packetsize is 1500Bytes (MTU value), the throughput rate drops, because a big packet means a lot of operation time before forwarding. Big packets keep the nodes busy for longer periods of time. The same applies for the delay metric.

We also investigate the performance of the network in different times during the experiment. In all the cases, throughput and delay are almost constant in the Static scenario. The interesting part is the Mobile scenario. In general, As node M moves from node S to node D, we notice a drop in performance (decreased throughput and increased delay). At the beginning, the routes are stabilized (Warm-up period), so the packets arrive correctly at node D. While node M starts to move, the topology becomes dynamic, and because the routes change, there are more lost packets, due to occurring of unavailable links and routes. Throughput becomes 0% in most cases, at the period of time $130 - 150$, when node M is almost back to node S.

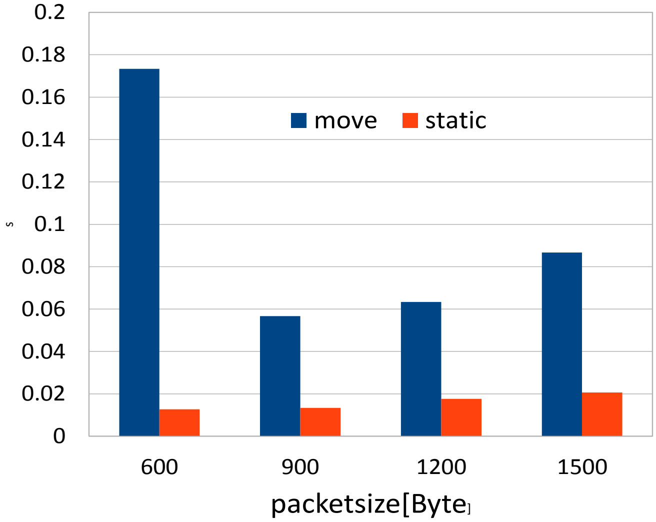
We would like to investigate on that more, in out future works, but it seems that routes are disturbed by the moving node, and there might be a false-positive effect because of mobility. Node D is the destination, so node S and other relay nodes are interested in OGM packets coming from node D. Those packets transverse the network from node D to node S, the same direction node M is moving. In this period of time, it might happen that OGM packets are stored inside node M's memory and forwarded at a later time (remember that BATMAN creates a buffer in each node for this purpose). When forwarded they may create the idea that node M is the best ranked neighbour, but that may be a false positive, because node M may not even have a valid route to node D anymore.

6 Conclusions and Future Works

In this paper, we showed the design and implementation of a testbed for MANET, where we can test and verify different algorithms. It is suitable to implement WSN,



(a) Throughput

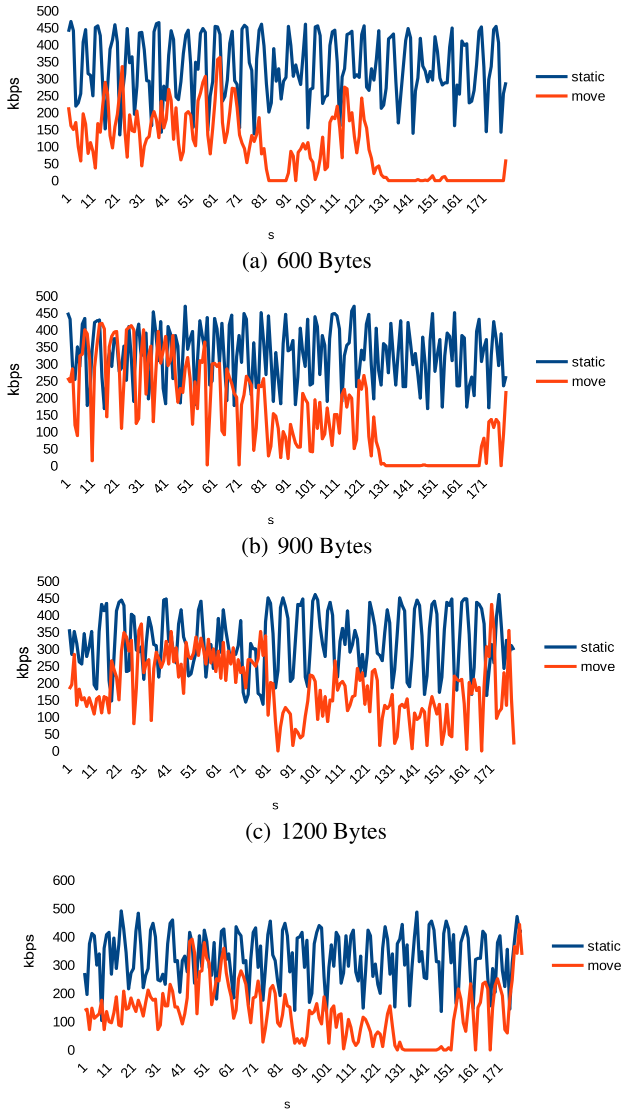


(b) Delay

Fig. 3 Average Results for Throughput and Delay.

IoT and WSAN applications in it. We conducted some experiments in order to analyze the performance of BATMAN routing protocol based on throughput and delay. From simulation results we concluded the following.

- For small packet sizes and very big packet sizes, the performance drops, due to congestion of network and nodes, respectively.
- Mobility of nodes, brings topology changes, route changes and decreased performance.



(a) 600 Bytes

(b) 900 Bytes

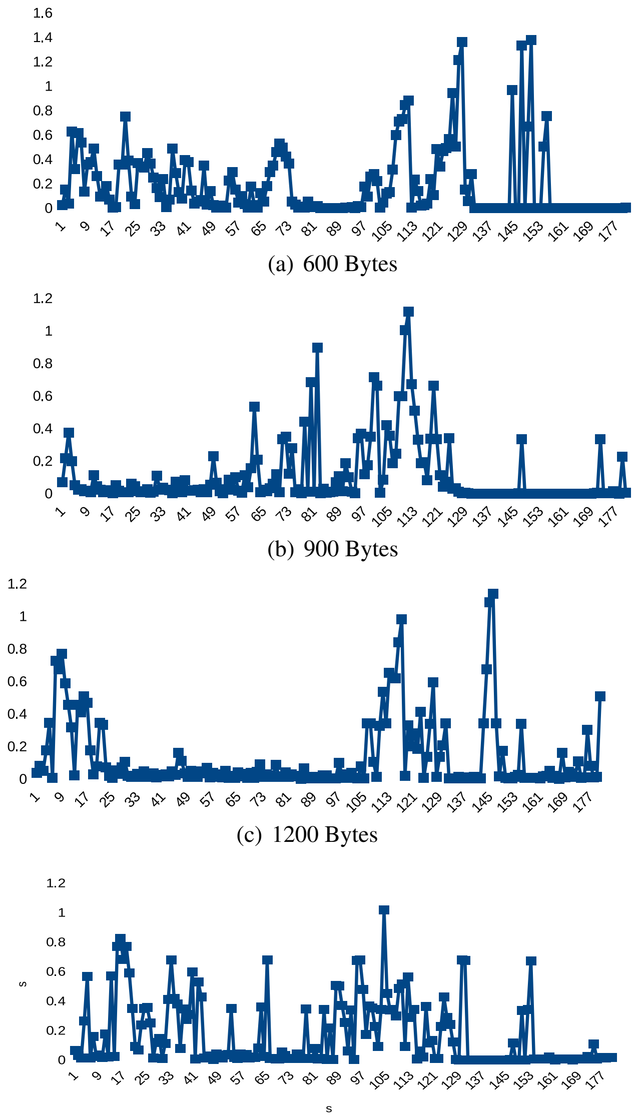
(c) 1200 Bytes

(d) 1500 Bytes

Fig. 4 Time-Domain Results for Throughput.

- Mobility of nodes, brings topology changes, route changes and decreased performance.
- The direction of movement may have an effect on Ranking Procedure of BATMAN.

In out future works, we would like to continue our implementations of sensor node and actor node in the network. We would also like to develop new algorithms



(a) 600 Bytes

(b) 900 Bytes

(c) 1200 Bytes

(d) 1500 Bytes

Fig. 5 Time-Domain Results for Delay.

that will be suitable for WSAN applications. Moreover, we plan to implement a smart Laboratory, where home automation and IoT will be researched and improved.

References

1. C. Felita and M. Suryanegara, “5G Key Technologies: Identifying Innovation Opportunity”, in International Conference QiR (Quality in Research), pp. 235-238, June 2013.
2. A. Gohil, H. Modi and S. Patel, “5G Technology of Mobile Communications: A Survey”, International COnference on INtelligent Systems and Signal Processing (ISSP-2013), pp. 288-292, March 2013.
3. X.Y. Li, “Wireless Ad Hoc and Sensor Networks: Theory and Applications”, Cambridge University Press, 2008, Hardcover Print.
4. S. Pellicer, G. Santa, A.L. Bleda, R. Maestre, A.J. Jara, A. Gomez Skarmeta, ”A Global Perspective of Smart Cities: A Survey”, in Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2013) pp.439,444, 3-5 July 2013. doi: 10.1109/IMIS.2013.79
5. L. Atzoria, A. Ierab, G. Morabitoc, “The Internet of Things: A survey”, Computer Networks, Elsevier, Vol. 54, Issue 15, pp. 2787-2805, October 2010.
6. RAPIRO, The programmable DIY Robot with Endless Possibilities, http://www.rapiro.com/, Last Accessed on August 2016.
7. G. De Marco, M. Ikeda, T. Yang and L. Barolli, “Experimental Performance Evaluation of a Pro-active Ad-hoc Routing Protocol in Outdoor and Indoor Scenarios”, Proc. of International Conference on Advanced Information Networking and Applications (AINA-2007), pp. 7-14, May 2007.
8. 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 International Conference on Advanced Information Networking and Applications (AINA-2009), pp. 307-314, May 2009.
9. M. Ikeda, L. Barolli, M. Hiyama, T. Yang, G. De Marco and A. Durresi, “Performance Evaluation of a MANET Testbed for Differenet Topologies,” Proc. of International Conference on Network-Based Information Systems (NBiS-2009), Indianapolis, pp. 327-334, August 2009
10. E. Kulla, M. Ikeda, L. Barolli and R. Miho, “Impact of Source and Destination Movement on MANET Performance Considering BATMAN and AODV Protocols“, Proc. of International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA-2010), pp. 94-101, 2010.
11. Open Mesh Webpage for a Collection of Tools to Build Free and Open Mesh Networks, Available online at http://www.open-mesh.org/
12. J. Bicket, D. Aguayo, S. Biswas, and R. Morris, “Architecture and Evaluation of an Unplanned 802.11b Mesh Network”, Proc. of International Conference on Mobile Computing and Networking (MOBICOM-2005), pp. 31-42 2005.
13. D. A. Maltz, J. Broch, and D. B. Johnson, “Lessons from a Full-scale Multihop Wireless Ad-Hoc Network Testbed”, Journal on Personal Communications, Vol. 8, No. 1, pp. 8-15, February 2001.
14. C. Perkins, E. Belding-Royer, and S. Das, “Ad-hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561 (Experimental), Jul. 2003.
15. D. B. Johnson, D. A. Maltz, and J. Broch, “DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad-Hoc Networks”, Journal on Ad-Hoc Networking, Chap. 5, pp. 139-172, 2001.
16. S. Hou, M. Wu, W. Liao and D. Wang, “Performance Comparison of AODV and DSR in MANET Test-Bed Based on Internet of Things”, IEEE 82nd Vehicular Technology Conference (VTC Fall), pp. 1-5, Boston, MA, 2015. doi: 10.1109/VTCFall.2015.7391074
17. P. Bellavista, G. Cardone, A. Corradi and L. Foschini, “Convergence of MANET and WSN in IoT Urban Scenarios” IEEE Sensors Journal, vol. 13, no. 10, pp. 3558-3567, October 2013. doi: 10.1109/JSEN.2013.2272099
18. Buffalo Products, WLI-UC-GNM: AirStation N-Technology 150Mbps USB 2.0 Client, http://www.buffalo-technology.com/, Last Accessed on February 2015.

19. Ubuntu 14.04.2 LTS (Trusty Tahr), http://releases.ubuntu.com/14.04/, Last Accessed on February 2015.
20. Raspbian Jessie Operating System for Raspberry Pi, https://www.raspberrypi.org/downloads/raspbian/, Last Accessed on August 2016.
21. A. Dainotti, A. Botta, and A. Pescap, “Do you know what you are generating?” Proc. of International Conference on Emerging Networking Experiments and Technologies (CoNEXT-2007), New York, USA, pp. 1-2, 2007.