Investigation into BATMANd-0.3.2 Protocol Performance in an Indoor Mesh Potato Testbed

Edmundo Chissungo
Department of Computer Science
University of Cape Town
Rondebosch 7701, South Africa
27 21 650 3661
chsedm001@uct.ac.za

Edwin Blake
Department of Computer Science
University of Cape Town
Rondebosch 7701, South Africa
27 21 650 3661
edwin@cs.uct.ac.za

Hanh Le
Department of Computer Science
University of Cape Town
Rondebosch 7701, South Africa
27 21 650 2667
hanh@cs.uct.ac.za

ABSTRACT

In this paper, we describe the performance of the B.A.T.M.A.N daemon (Batmand) protocol on an indoor Mesh Potato (MP) testbed. The MPs are small devices used for voice communications over the wireless medium but also supports data. The batmand protocol is designed for *ad hoc* wireless networks. We measure delay, packet loss in order to understand the MPs network performance. The experiments used packets of varying sizes over multiple hops. We analyze the data to see if the network latency for up to four hops is within the recommended boundaries set by ITU-Recommendation G. 114. We also observe the how the network's performance is affected by the varying packet sizes. Finally the experiments also reveal the common issues found on the wireless medium and also indoor testbeds.

Categories and Subject Descriptors

C.2 [**COMPUTER-COMMUNICATION NETWORKS**]:

General Terms

Measurement, Performance, Design, Experimentation.

Keywords

BATMANd-0.3.2, Testbed.

1. INTRODUCTION

Wireless nodes in *ad hoc* wireless mesh networks lack the capability for communicating with nodes not directly connected to it. Due to a limited communication range routing protocols exist as a mechanism to overcome this problem and thus are in charge of performing data forwarding between nodes helping to form an *ad hoc* network. There exists an abundant number of routing protocols [1] each fitting into a pre-existing taxonomy. However even with so many protocols one has still to be developed that is better than all others in all aspects. Previous works into one protocol called Better Approach to Mobile *ad hoc* Networking (B.A.T.M.A.N or Batman) suggests that "Batman is the panacea that community wireless mesh networks have been waiting for" [2].

Batmand is an open-source wireless routing protocol and is the predominant implementation of B.A.T.M.A.N routing algorithm as it used extensively as the routing protocol in a wireless communication device called the Mesh Potato (MP) [3]. The MP is used as an alternative communication device for communities. MPs use Voice over IP (VoIP) to allow users to wirelessly make calls between connected nodes on the *ad hoc* wireless network.

The Batmand protocol's performance on the device (MP) has yet to be measured. This would be useful as it would give us valuable insight into the real-world performance of this protocols when used as a solution for community wireless networks. Here we present a practical insight into a real-world performance of Batmand.

This work is structured as follows: we start in Section 2 with the background on the protocols. This is followed by Section 3 which describes the experiment set up. The results section is follows in Section 4. Finally conclusions are drawn in Section 5.

2. Related work

An overview of the Batmand routing protocol is presented next. This is followed by an overview of the MP and lastly the relevant literature.

2.1 B.A.T.M.A.N

Batman is a simple and robust algorithm for establishing multi-hop routes in *ad hoc* networks [4]. As explained by Johnson, D., *et al* [2], Batman does not maintain the full route to the destination, each node along the route only maintains the information about the next link through which the node can find the best route. The objective is to maximize the probability of delivering a message. Batman does not attempt to check the quality of each the link, it just checks its existence. The protocol does these checks by having all nodes periodically broadcasts hello packets to its neighbours, these packets are known as originator messages (OGM).

The structure of the OGM packet periodically sent is here presented:

- originator address
- sending node address: this is changed by receiving nodes and then the packet is re-broadcasted
- unique sequence number: The sequence number is used to check the concurrency of the message
- bidirectional link flag: used when the OGM packet received is its own and the sender is someone else
- time to leave (TTL)

The links are compared in terms of the number of originator messages that have been received within the current sliding

SAICSIT '11, October 3–5, 2010, Cape Town, South Africa.

window this value is called the transmission quality (TQ) and is the routing metric used by Batman. The sliding window is a fixed value that defines a range of the unique sequence numbers afforded to each OGM packet sent by a node.

Batman is in essence a proactive routing protocol as it pre builds its routing table, however the way in which it conducts route discovery and maintenance are unlike any other routing protocols so does not fit into other pre-existing taxonomies [5]. Batman has three implementations, the two we will mention are the layer three (OSI stack) which is implemented as daemon in Unix operating systems (OS) it is called Batman daemon (Batmand) to date on version 0.3.2. The second one is a layer two implementation called Batman advanced (batman-adv) [6]. The difference between the two implementations is layer in which they operate.

Batman's routing technique causes low processing and traffic cost. This makes it an attractive option for use on devices that have small processors such as the MP. In this work we focus on the Batmand-0.3.2 and it performance on the MP. The MP is described next.

2.2 Mesh Potato (MP)

The village telco group [3] descibe the MP as a wireless System on Chip (SoC) – the processor and all wireless functionality is combined in a single chip. MP uses the *ad hoc* demo profile. It is slightly different from normal *ad hoc* in order to get around some bugs. The *ad hoc* profile allows any wireless node to connect to any other node within range which forms the wireless blanket or cloud and with the use of Batmand as a routing protocol creates a communication network.

The MP was initially developed for Voice over IP (VoIP) using plain old telephones (POTs). The MP can also be used for data networks.

2.3 Literature

The experiments conducted were performed on an indoor testbed; existing works show us the benefits and drawbacks of this approach.

Lundgren, [7] surveys the field of *ad hoc* routing and related real world testbeds. The author in this work argues that different *ad hoc* routing protocols need to be complemented with real-world experiments this view is also supported by [8]. Their reasoning is that real-world experiments need to be done in order to reveal real-world effects that may not be visible in simulation studies and also to gain practical experience.

3. Experiment

Our approach in the experiments was to set up a testbed and have the actual MPs be the nodes in the testbed. We mimicked techniques described by P.Gunningberg, *et al*, [9] and B.Hagelstein, *et al*, [10]. The authors use techniques such as intentional attenuation of the signal level on each node in the testbed to enable some nodes to be out of range of others and thus creating multi-hop network topologies.

3.1 Physical Testbed

The physical testbed used in our experiments was achieved by deploying a MP network in the Computer Science building at the University of Cape Town (UCT). Figure 1 shows the connections achievable in the largest implementation of the MP testbed given the space available and signal propagation issues caused by the close proximity of the nodes. We used two Unix boxes ad MPs all running the Batmand-0.3.2 routing protocol. One Unix box was place in the farthest room on the third floor. This is show on the far left of Figure 1 (bottom left corner) Node (0). In the opposite direction, we placed the second Unix box also in the farthest room. This is show of the farthest right of Figure 1 (bottom right corner) Node (6). In between these two Unix boxes are the MP Nodes (1-5). The MPs did all the routing on the network. The dotted lines in Figure 1, between the network nodes, represent the existing links between nodes. Each link (dotted line) represents a hop network. These Unix boxes would generate and receive the packet traffic on the network and are passive network nodes from a routing perspective.

3.2 Scenarios

The testbed was rolled out as need and eventually looked like Figure 1. Each of the hops included two Unix boxes and zero or more MP placed in between the Unix machines as need to achieve the desired number of hops. This is shown on the

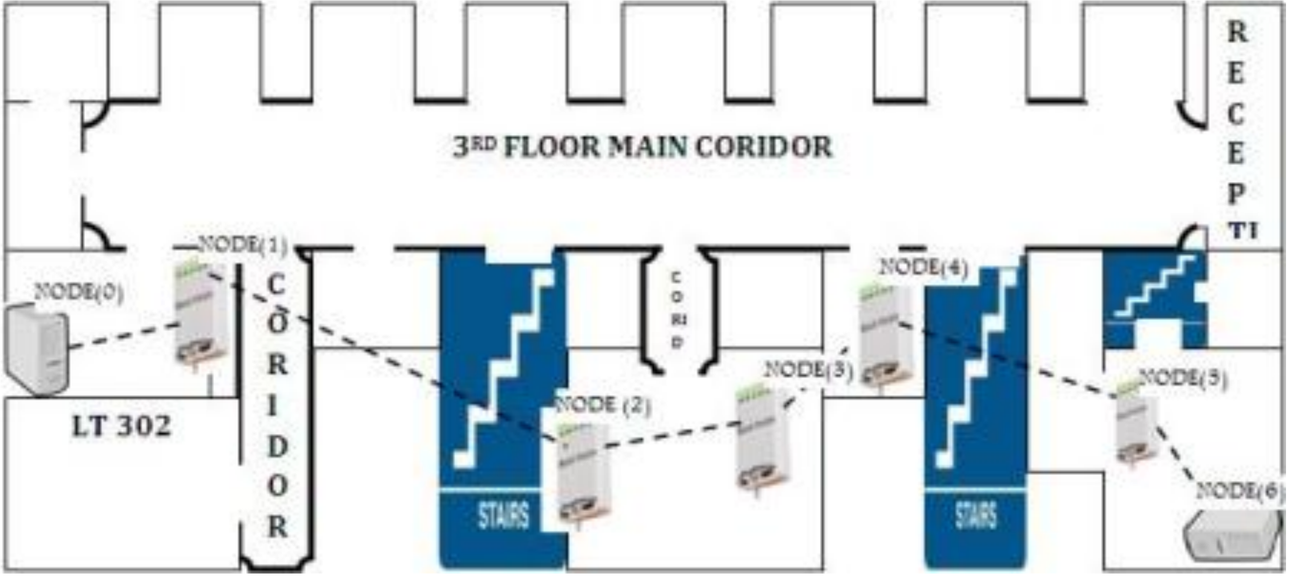

Figure 1 shows the physical network topology for the testbed used in the experiments conducted in this work. On the far left (bottom left) and far right (bottom right) are the Unix boxes which generate and receive the network traffic. In between are the MP nodes that perform the routing. Each node ran the batmand-0.3.2 protocol. Each dotted line represented a hop in the network.

Figures 2, 3 and 4 these were a few of the scenarios used in the experiments. We note that the one hop scenario does not use any MPs, the data gathered from it serve for comparison purposes with the scenarios with the MPs.

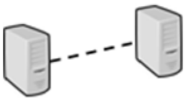

Figure 2 Scenario 1 (1Hop) One meter distance between the Unix boxes.

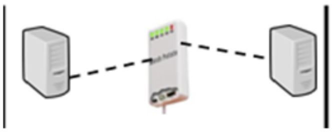

Figure 3 Scenario 2 (2Hop), one meter distance between the left Unix box and the MP. 15 Meters between the MP and the Unix boxes.

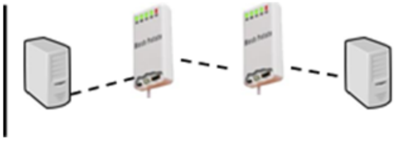

Figure 4 Scenario 3 (3 Hop) , the Unix boxes were separated by approximately 40 meters, the MP by 38 meters, one meter between Unix box and closest MP.

3.3 Testing

The testing was conducted in the physical topology mentioned in Figure 1. In the testbed the Unix box nodes generate traffic in the form of data packets. We use packets of size 73 bytes and 1500 bytes, each representing voice packet or standard Ethernet packets respectively. In doing this we hoped to compare the performance of the network when dealing with voice and data packets sizes.

In each of the experiments conducted we varied the load which were packets generated and sent by the Unix box on the far left on Figure 1. We sent 1000 UDP packets of size 73 bytes, this was repeated 60 times, referred here as iteration. We then increased the packets size to 1500 bytes. We also iterated this 60 times as well. We did each of the experiments for each independent number of hops represented by the scenarios in Figures 2, 3 and 4. The chosen metrics are Packet Loss Ratio (PLR) and Delay (D). For each hop we observe how load and number of hops affects each of the metrics chosen to be scrutinized.

4. Results

4.1 Packet Loss Ratio

VoIP is not tolerant of packet loss to the extent that high packet loss can degrade the call quality. In VoIP, high packets loss will cause a call to break up, and too much of this will make the conversation incomprehensible [11]. Table 1, below, shows the average percentage packet lost in each hop throughout the experiments.

Table 1, below, shows us what we expected to see, the larger the hop number the higher the packet loss. The same idea also works for packet sizes. Larger packet sizes can also generate higher packet losses. Larger packets are broken down into smaller chunks to be sent; therefore, larger packets have larger number of chunks to be sent which increase the probability of loss, aggravated by the increasing hop numbers

Table 1 shows us that for 1500 byte packets the loss rate rises sharply with hops at 74% on the second hop and 87% on the fourth hop. The data suggests that perhaps the MP network is not well suited for services with large data packets such as Ethernet. The 73 byte data shows that for voice packets the loss remains comparatively lower than 1500 byte packets for lower hop numbers. The data shows that packet loss affects all packet sizes at the fourth hop the same way as both 73 byte and 1500 byte packets experience 85 and 87% loss respectively.

Finally, the third hop for the 73 byte data is lower than that of hop two which is counter intuitive to what we expected. This suggest that the third hop link was strangely better then the two hop links. Again these kinds of variations are due to the nature of the medium which is often unpredictable and even unstable. After packet loss, delay is considered the "second most disruptive impairment in VoIP networks" [12] and we address delay on the MP testbed next.

Table 1Average (AVG) and Standard Deviation (SD) for Packet Loss

Hop	Avg 73 Byte Data	SD for 73 Byte Data	Avg 1500 Byte Data	SD for 1500 Byte Data
1	0.02	0.0550090	2.3004	0.9498273
2	56.14	10.185532	74.288	8.8777094
3	45.360	20.656266	79.300	16.536426
4	85.154	4.1411533	87.561	7.833822

4.2 Delay/Latency

Delay is the time taken to transmit a packet from a source to a destination (one-way latency) in milliseconds (ms). The effects of delay to the caller generally appear as echo and lag. Acceptable and unacceptable delay values for voice applications where established by the International Telecommunication Union G series (ITU-G) [13]. According to ITU-Recommendation G. 114 [14] delay values below 150ms is acceptable, 150ms to 400ms is acceptable provided callers are aware of the impairment. Values above 400ms are unacceptable.

Table 2, shows the values we measured on our testbed. Delay was expected to increase with the increasing hops and packets sizes because it takes longer to send more data over larger number of hops. Again we notice that in the third hop data delay improves which could be due to improvements in the medium in the three hop scenario, Figure 4.

Our delay values are all within the ITU-Recommendation G. 114 boundaries for acceptable delay meaning that the MP network is well suited for VoIP. However, whether delay on MP networks for other applications are within acceptable boundaries is applications depended. The relevant test would have to be carried out for those applications. Ethernet it is within boundaries.

Table 2 Average (Avg) and Standard Deviation (SD) Delay/Latency for the 73 and 1500 Byte Packets

Hop	Avg 73 Byte Data	SD for 73 Byte Data	Avg 1500 Byte Data	SD for 1500 Byte Data
1	29.224457	12.60131	32.73293	14.681697
2	22.944083	17.930959	55.02398	7.9203358
3	15.33785	8.7488883	141.2587	33.652806
4	42.057758	24.007469	150.6713	48.744058

5. Conclusion

In our research through the literature surrounding the Batmand routing protocol we did not see any evidence of tests run on the one device that uses it the most, the MP. We chose to perform tests on an actual MP testbed.

We focused our attention on packet loss and delay in order to help us understand the performance of the MP network with increasing hops and packet sizes. The results we obtained for delay suggest that even at higher hops the network can support VoIP as the values fall well within the boundaries recommended by the ITU-Recommendation G. 114. However the packet loss data gathered suggests the opposite as it shows that network’s performance decreases sharply after two hops for both voice and Ethernet sized packets. Based on the delay data we can say that the MP network is well suited for VoIP traffic. The sharp rise in packet loss for Ethernet sized suggests that data traffic might not be well suited to the MP network.

Further work involving other metrics such as jitter and throughput needs to be performed in order for us to gain a clearer view of the performance of the Batmand protocol on the MPs. Our research stands as a first attempt at analyzing the wireless mesh routing protocol’s performance of a device that is intended for community wireless mesh networks.

Lastly, we witnessed a few network anomalies which we attributed to the nature of radio packet networks. In future we hope to re run the same experiments on a different floor of the building in order to see if these anomalies are really due to the nature of the communication medium or of the network itself. Furthermore, comparing Batmand and batman-adv would give us insights into the performance differences between layer 2 and layer 3 routing protocols. This would be a valuable contributions and previous work done on this [5] had inconclusive results. Revisited proof read

6. REFERENCES

[1] K. Kowalik M.Davis. (2006) Why are there so many routing protocols for wireless mesh networks.

[2] N.Ntlatlapa, C.Aichele D.Johnson, "Simple pragmatic approach to mesh routing using BATMAN," in *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries, CSIR, South Africa*, Pretoria, 2008, pp. 6-7.

[3] Village Telco Group. (2010, May 10) Village Telco. [Online]. http://www.villagetelco.org/2009/06/voice-over-b-a-t-m-a-n/

[4] C. “Elektra” Aichele, M. Lindner A.Neumann, "B.A.T.M.A.N Status Report," Status Report 2007.

[5] M.W.Dixon, T.Koziniec D.Murray, "An experimental comparison of routing protocols in multi hop ad hoc networks," 2010.

[6] open-mesh. (2011, June) open-mesh. [Online]. http://www.open-mesh.org/wiki/batman-adv

[7] H.Lundgren, "Implementation and real-world evaluation of routing protocols for wireless ad hoc networks," in *IT Licentiate theses, Uppsala Universitet*, 2002.

[8] R.Hain R.Ramanathan, "An ad hoc wireless testbed for scalable, adaptive QoS support," in *Wireless Communications and Networking Conference*, 2000, pp. 998-1002.

[9] P.Gunningberg, H.Lundgren E.Nordstrom, "A testbed and methodology for experimental evaluation of wireless mobile ad hoc networks," in *First International Conference on: Testbeds and Research Infrastructures for the Development of Networks and Communities*, 2005, pp. 100-109.

[10] B.Hagelstein, M.Abolhasan J.C.P.Wang, "Experimental evaluation of IEEE 802.11 s path selection protocols in a mesh testbed," in *4th International Conference on: Signal Processing and Communication Systems (ICSPCS)*, 2010, pp. 1-3.

[11] M.Domingo, J.Arnedo, A.Talevski, P.Radman J.Singh, "VoIP: making secure calls and maintaining high call quality," in *in Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, 2010, pp. 56-62.

[12] Brady Volpe. (2001, May) bradyvolpe.com. [Online]. http://bradyvolpe.com/troubleshooting-docsis/voip_impairments_delay_jitter/

[13] International Telecommunication Union. (2011, May) ITU-G Transmission system and media, digital systems and networks. [Online]. http://www.itu.int/rec/T-REC-g

[14] O.W.T. Time, "ITU-T Recommendation G. 114," *ITU-T*, May 2000.

[15] R.Raichle, R.Dougla J.T.Kaba, "Testbed on a desktop: strategies and techniques to support multi-hop manet routing protocol development," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, 2001, pp. 164-172.

[16] Inc Geeknet. (2011, May) iperf.sourceforge.net. [Online]. http://iperf.sourceforge.net/

[17] Inc Geeknet. (2011, May) fping.sourceforge.net. [Online]. http://fping.sourceforge.net/

[18] S.Fowler,S. Hasan, Y.Ben-David. JaldiMAC--Beyond Simulation. article.