Situation-aware Routing Based on Link Quality for Static Mesh Networks with Mobile Nodes

Hlabishi I. Kobo, William D. Tucker and Xiaoming Liu
Department of Computer Science
University of the Western Cape, Private Bag X17, Bellville 7535 South Africa
Tel: +27 21 9592461, Fax: +27 21 9591274
E-mail: {**2654952**, btucker, 2478875}@uwc.ac.za

Abstract- Situation-aware routing seeks to improve quality of service on hybrid wireless mesh networks by making routing decisions based on the current situation of the network. BATMAN-adv is a mesh routing protocol that counts beacons as a link quality metric. We modified BATMAN-adv to give more recently received beacons more weight, thereby giving a more precise indication of the current state of a link. We then compared the original protocol with our modification in a small laboratory test bed. Results show little relation between jitter and packet loss. Jitter is, however, proportional to throughput. The average throughput achieved on both protocols was almost the same but we noticed that the throughput on our modified version increases as the network grows. Our protocol modification suffered from packet loss at low bandwidth rates but this reduces as the transfer rate increases and buffer size shrinks. We conclude that our situation-aware protocol modification shows potential to address issues pertaining to scalable and congested static mesh networks with mobile nodes.

Index Terms—Mobile/wireless protocols, TCP/IP & Layer 3 protocols, mesh, WiFi 802.11

I. INTRODUCTION

Situation-aware routing concerns making routing decisions based on the ongoing dynamic status of networked nodes. This paper presents a situation-aware routing metric calculation of only the most recent link quality data to inform routing decisions on static wireless mesh networks with mobile nodes. This type of network can be referred to as hybrid mesh network. We believe these types of mesh networks will become prevalent as mesh network protocols improve and mobile devices become more powerful and able to run such protocols. Fig. 1 illustrates the hybrid mesh network concept. Our goal is to improve link quality-aware routing in BATMAN (described below) in order to optimize quality of service (QoS) and throughput on such networks.

A Better approach to mobile ad-hoc network (BATMAN) is a proactive mesh network routing protocol. BATMAN's control messages, called originator messages (OGMs), are relatively small packets of about 52 bytes. BATMAN's nodes do not maintain the routing information of the entire network [1]. Rather, each node only maintains information about the best next-hop towards the destination [1][2]. This reduces the signal overhead and avoids unnecessary knowledge about the whole network. The objective of this protocol is to enhance the probability of delivering a packet. The protocol maintains information about the existence of a node and thus does not check the quality of the packet [3].

Fig. 1: A hybrid mesh network, applicable to a rural area, with static mesh routers inside homes, and mobile mesh nodes on cell phones. Note that link quality in such networks is continually changing as phones move around.

All BATMAN nodes periodically send/broadcast control packets, or OGMs. Each OGM contains the original sender's address, address of the node rebroadcasting the OGM, TTL (time to live) and a sequence number. The sequence number is incremented for each OGM, i.e. the first OGM gets 1 and so on. Thus, BATMAN also keeps track of the freshness of an OGM. Any sequence number received with a value lower than the previous one gets dropped [2]. The TTL is used to limit the number of hops on which the packets must pass through before it expires (gets dropped). Upon receiving the OGM, each node then rebroadcasts it to its neighbors. However, each node only rebroadcasts OGMs coming through the current best next-hop. The number and the reliability of the OGMs determine the route discovery as well as neighbor selection.

This paper describes our efforts to optimize BATMAN, a mesh routing protocol, to be more effective for a hybrid mesh network. The crux of the problem is to optimize the routing protocol so it can adapt and react quickly to rapid and dynamic topological changes. We propose situation-aware methods to improve the routing decisions based on link quality to achieve better QoS and throughput. BATMAN's routing algorithm checks for the existence of a link and increases the probability of delivering a packet through that link [4]. Our method adapts the routing protocol to use a simple weighting mechanism. We believe recent packets provide a clearer indication of link quality at a particular moment in time so we give more recently received packets more weight in the routing decision. We tested the protocol enhancement on a small WMN with some mobility. The results are encouraging, and not too far off the mark from the original BATMAN.

The rest of the paper is organized as follows: Section II reviews some related work. Section III presents the methods for the protocol enhancement and the experimental setting. Section IV presents preliminary results, and Section V discusses them. Section VI concludes the paper and recommends some future work.

II. RELATED WORK

A. Routing Protocols

Routing is a process of delivering data packets from a source (sender) node to destination (receiver) node on a network. Routing protocols deal with the maintenance, creation, establishment and discovery of such routes [2]. Routing protocols are based on three protocol classification categories: reactive, proactive and hybrid.

Reactive protocols also referred to as on-demand protocols create a route from source to destination only when needed, i.e. when there is actual data to be sent. This scheme uses network flooding to find the routes [2][5]. This protocol scheme is suited for mobile ad-hoc networks where there are frequent topological changes due to the mobility of routers [1] [5]. According to [5], flood based route discovery provides high network connectivity and low message overhead. More importantly, the method does not waste bandwidth by propagating control packets when it is not necessary [1]. This scheme, however, leads to higher latency on the network because of route discovery. [2] Argues that reactive protocols are more suitable for a network with static traffic patterns whilst proactive protocols suit dense networks with bursty traffic patterns [2].

Ad-hoc on demand distance vector (AODV) is one of the popular reactive protocols and hence creates routes on demand. AODV has single path routing and is based on hop-by-hop routing [2]. Single path routing means that a node can only have one path towards a destination [2]. The AODV routing table only stores information about the best next-hop towards a destination [2]. Sequence numbers are used to ensure loop-free routes and to ensure the freshness of the routing information [6] [7]. The AODV protocol uses unicast, broadcast, as well as multicast for communication on the network. It uses broadcast to flood route requests, then the intermediate nodes and the destination nodes send a unicast route reply [7]. There are multicast groups where a multicast of sequence numbers takes place [6]. On-demand multipath distance vector routing (AOMDV) was developed to alleviate link failures and link breaking suffered in AODV by using multipath routes [6] [7].

Dynamic source routing (DSR) and SrcRR are other reactive protocols which are based on source routing. DSR as defined by Jonhson et al. is a simple and efficient routing protocol designed specifically for multihop wireless ad hoc networks with mobile nodes [8]. DSR has two fundamental mechanisms: route discovery and route maintenance. These two components allow the nodes to self-discover and self-maintain routes to dynamic destinations in the ad hoc network. SrcRR is based on DSR but differs because it uses ETX routing metric (see next section).

Other on demand routing protocols are based on link reversal routing (LRR). LRR suits ad hoc networks due to their ease on adaptability and scalability with more emphasis on fast changing topology networks [9].

Associativity Based Routing (ABR) is a source-initiated reactive protocol. It is a bandwidth efficient distributed routing protocol used in ad hoc networks [10]. ABR uses periodic beacons to let neighbours know about other neighbours' existence. Another beacon based reactive protocol is the Signal Stability Routing protocol (SSR). This protocol selects routes based on signal strength and nodes location stability.

In proactive protocols, each node in the network maintains a table containing routing information of the entire network. Each node then periodically broadcasts control packets (hello packets) to the whole network to let other nodes know about its existence. The routing information is periodically updated to maintain the adequacy of the routing information and thus the network will always be up to date with respect to topological changes. The biggest advantage of this scheme is the minimization of route discovery delay and consequently lower latency in delivering a packet. However because of the periodic updates of control messages that get propagated through the entire network, the overhead increases. Thus, bandwidth consumption also rises. Proactive protocols are also known as table-driven protocols and consume memory space.

Optimized link state routing (OLSR) protocol is a proactive protocol based on a link state algorithm. OLSR's objective is to reduce the size of the control packets as well as the overhead cost by broadcasting control packets [4]. This protocol is an optimization of the link state protocol for mobile ad-hoc networks [4]. It uses a hop by hop routing metric. Multipoint relays (MPR) are the key concepts in OLSR. MPRs are the subsets of the neighbours of which a node uses to forward broadcast messages. MPRs reduce duplicate retransmission in the same region and thus minimize flooding overhead [4].

Destination-Sequenced Distance Vector (DSDV) is also a proactive routing protocol developed by Perkins et al. based on the classic Bellman-Ford routing algorithm [11]. Global State Routing (GSR) is a link state MAC-efficient protocol similar to DSDV. The main goal of this protocol is to address the shortcomings endured in many LS (link state) protocols such as flooding of routing messages. Thus GSR controls the size and the number of the control packets in order to achieve optimized MAC throughput.

The Wireless Routing Protocol (WRP) is a proactive distance vector routing protocol aimed at maintaining routing information on the network [12]. Each node in the network maintains four routing tables: distance table, routing table, link cost table and message retransmission list (MRL) table [11] [12].

Fisheye State routing (FSR) is a LS routing protocol inspired by the fish-eye technique created to reduce the size required for graphical data representation [13]. Clusterhead Gateway Switch Routing (CGSR) is a cluster based proactive protocol which uses DSDV. Nodes are grouped into clusters where cluster-heads are elected.

The BATMAN algorithm (described in Section I) is also a proactive protocol. However, it experiences serious flaws in dealing with asymmetric links. BATMAN advanced, referred to as BATMAN-adv, is a Layer 2 protocol introduced to overcome this setback by using a Transmit Quality (TQ) algorithm. BATMAN-adv consists of two fundamental functions: receiving link quality (RQ) and transmit link quality. Receiving link quality deals with the probability of transmitting a packet successfully towards a node [14]. The transmitting link illustrates the probability of transmitting a packet successfully towards a neighbor [14]. TQ is the most important because RQ does not influence the routing decision. RQ is determined by the by the number of received OGMs. Echo link quality (EQ) is the number of the rebroadcasted OGMs from neighbors. TQ is calculated by dividing the EQ by the RQ i.e. $TQ = EQ/RQ$ [14].

Hybrid protocols exhibit the behavioural design of the two above mentioned protocols. Hybrid protocols are very challenging because the switch from one protocol to another needs to be very sharp. However, this is still a major concern and thus hybrid protocols are still theoretical rather than practical due to their complex implementation [1].

MeshDv is a hybrid protocol which uses the combination of proactive route computation for the routers and on-demand path request for clients [15]. The proactive route is based on the destination-sequenced distance vector (DSDV) protocol [15].

Zone Routing Protocol (ZRP) is a zone based hybrid protocol. ZRP proactively maintains routing information for the local neighbourhood, referred to as the routing zone. It reactively acquires routes to destinations that are outside the routing zone. Zone-based Hierarchical Link State (ZHLS) routing protocol is another zone based hybrid routing protocol, and is based on global positioning system (GPS). Other hybrid protocols includes SHARP (Hybrid Adaptive Routing Protocol) and HARP (Hybrid ad hoc routing).

B. Routing metrics

Routing protocols use metrics to select the best routing path. Several situation-aware routing metrics have been proposed, as well as applied, in many routing protocols.

The hop count routing metric counts the number of hops between a sender and its destination. Hop count is commonly used in routing protocols such as AODV, DSR and DSDV [5]. Hop count is simple to compute when compared to other metrics, and this is the main reason it has been preferred by many routing protocols. However, hop count does not consider packet loss or bandwidth, and hence results in low throughput [5].

Expected transmission count (ETX) is a situation-aware metric which considers the number of MAC layer transmissions needed to successfully deliver a packet through a link [5] [16]. The ETX metric captures the effects of packet loss and path length. Each node broadcasts probe packets to its neighbors and they send a back a reply/report [16]. The metric is calculated by the number of probe packets received by its neighbor in both directions [12]. ETX is isotonic, thus ensures easy calculations of minimum weight paths [5]. The ETX metric does not consider bandwidth, interference, or the link transmission variance [5].

The Expected transmission time (ETT) metric was developed to overcome the shortcomings of ETX and hence it is an optimization of ETX. ETT takes bandwidth and link transmission difference into consideration for its path selection computation. The ETT of a link is measured by the expected Layer 2 durations it takes to successfully transmit a packet through that link [5]. However, since ETT uses a single path channel interference (both inter-flow and intra-flow), this remains a major drawback in ETT, like in ETX [5]. Intra-flow interference is interference between intermediate routers sharing the same path while inter-flow is between neighboring routers competing for the same channel.

Weighted cumulative ETT (WCETT) was designed to overcome the shortcomings of both ETX and ETT in order to reduce intra-flow interference [5]. This is done through the use of multi path channels. However, since it is not isotonic, it has not been used by any algorithm [5]. Another drawback of WCETT it does not consider inter-flow interference and its effects. The metric of interference and channel-switching (MIC) addresses the shortcomings of WCETT by considering inter-flow interference and as well as solving the some of the non-isotonic effects.

C. Moving averages

The idea of using moving averages is not new. It is commonly used in economic systems for computing and plotting stock markets. Moving average (MA) is an arithmetic result calculated by averaging a number of past data points [17]. A simple moving average (SMA) is calculated using the mean of a given set of values. The sum of the set is divided by the number of elements in that set. It is similar to a statistical computation of a mean yet different by the fact that only a recent n number of data values are considered.

Exponential moving average (EMA) is another type of MA that gives more weight to recent values in order to make it more responsive newer information [17]. In relation to routing protocols more weight is applied to more recent OGMs, for example, for precise current link quality estimation. In stock market analysis, EMA uses the formula: $EMA = (P*\alpha) + (\text{previous } EMA* (1-\alpha))$ where P = current price, α = smooth factor = $2/(1+N)$ and N is the number of time periods. [18] applied this in BATMAN protocol in terms of round trip-time formulated as: $estimateRTT = (1-\alpha)* estimateRTT + \alpha*sampleRTT$ where sampleRTT is the RTT measured with the last packet, α is a smoothing operator with a constant value of~0.125m [18].

Weighted moving average (WMA) is another type of MA that also gives more weight to recently received data values. WMA multiplies the most recent value with its sequence value and monotonically decreases with iterations. For example, given a set of 10 values, WMA would multiply the value at index 10 with 10 and value at 9 with 9 and so forth.

D. Routing in mobile phones

BATMAN has been successfully deployed on a new routing device called the mesh potato by Village Telco Project (www.villagetelco.org). A mesh potato is a wireless access point combined with an Asynchronous Telephone Adapter (ADT) suitable for a rural network. The Village Telco team has also successfully ported the BATMAN stack on a selection of Android mobile phones. We aim to use such phones, known as Batphones, as mobile nodes in our hybrid WMN.

III. METHODS

This section describes how we optimized the BATMAN protocol to make situation-aware routing decisions based on link quality. We also describe the experimental design to compare baseline performance to that of the modification.

Given the mobility of mobile nodes, rapid topological changes in a hybrid mesh network are inevitable. Thus, the ideal approach is to take the current network situation into consideration when making routing decisions. In BATMAN, the best link is measured by the highest number of OGMs received from the destination over a current sliding window. Much can happen within a second in an ad hoc wireless network. Any link with a sliding window that records a lot of OGMs at the beginning and fewer at the end due to superior link strength at the beginning stands a chance of being the best as opposed to the one that records a lot towards the end but fewer in total. For example, suppose one has a sliding window of 10, link L1 records [1111100000] with 5 OGMs at the front, and link L2 [0000001111] with 4 OGMs seen at the end. BATMAN will chose L1 as the best next hop because of the higher number of OGMs, but actually, the current best option would be L2 because the most OGMs have arrived there more recently.

Our method prioritizes the recently received OGMs in the sliding window, and would therefore correctly choose L2 over L1. We sum the indices on which OGMs were recorded in a given window. From the example above, we would have link L1: 1+2+3+4+5 = 15 and link L2: 7+8+9+10 = 34. This is a more accurate numeric representation describing the current situation of the two links.

This section explains the experimental design and procedure to evaluate our BATMAN modification. We created a mesh network composed of four nodes as shown in Fig. 2. All nodes ran Linux version 2.6.32-31 with 802.11bg network cards. We used BATMAN advanced version BAD2010.1.0. Note that node B is the server, and node C is a laptop that we can move around during the tests.

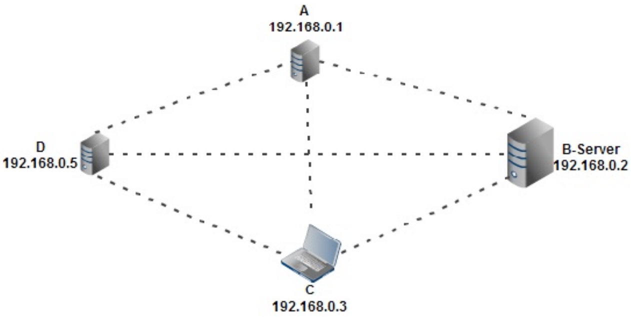

Fig. 2: The experimental test bed.

The experiment was designed to compare the performance of unmodified BATMAN-adv with our modified version on a dynamic mesh network with and without congestion. Our main objective is to show that situation-aware routing is viable and effective in a hybrid WMN. The test parameters examined were jitter, packet loss and throughput. We assume that jitter also covers latency.

We installed Iperf on all of the nodes to conduct the tests. Node B was used as a server (receiver) whilst the others were clients (senders). We configured Iperf to send packet flows representing voice packets to the server. We set a transfer interval of 60 seconds with a report back of 10 seconds. This was run 10 times for each parameter (herein referred as 10 flows). During the transfer interval, Iperf sent about 4000 UDP (User Datagram protocol) packets, about 665 each 10 seconds, with a maximum size of 1500 bytes. The parameters were tested with a selection of transfer rates and buffer sizes. The default settings were 1MB/s (megabytes per second) of bandwidth and 41 KB (kilobytes) for the buffer size. The transfer rate was regulated over 1MB, 100MB and 150MB speeds whilst buffer size was varied over 41KB, 31KB and 11KB. The first comparison combination consisted of all the transfer speeds with the default buffer size of 41KB. The second comparison combination applied the buffer size variations to the default transfer rate of 1MB/s. Lastly, the 150MB/s rate was applied to the 11KB buffer size to achieve maximum congestion of the compared rates and buffer sizes.

IV. RESULTS AND DISCUSSION

The results of jitter, packet loss and throughput comparisons in these combinations are presented in Table 1, Table 2 and Table 3, respectively. We measured average jitter, packet loss and throughput with the rate/buffer size combinations mentioned above. Fig. 3 and Fig. 4 are illustrative examples of packet loss only.

Jitter						
	1M 41kB	100M 41kB	150M 41kB	150MB 11KB	1MB 31KB	1MB 11KB
BATMAN-ADV						
A	32.06	45.24	48.35	45.19	20.04	17.47
C	27.42	41.62	43.43	28.08	13.45	20.43
D	36.91	53.61	54.78	37.83	28.85	32.39
BATMAN-ADV modified						
A	150.62	38.59	336.20	109.30	59.75	101.80
C	0.25	102.60	142.00	179.50	258.40	131.30
D	216.58	58.18	52.80	45.80	50.72	24.24

Table 1: Jitter comparisons.

Packet Loss						
	1MB 41kB	100M 41kB	150M 41kB	150MB 11KB	1MB 31KB	1MB 11KB
BATMAN-ADV						
A	3.80	0.77	1.13	6.07	3.13	2.77
C	6.54	6.01	5.53	4.30	4.90	5.28
D	1.33	1.12	1.26	0.35	0.41	0.51
BATMAN-ADV modified						
A	7.92	3.21	1.73	2.02	2.25	4.03
C	9.45	1.75	2.12	1.267	2.09	1.67
D	8.58	0.94	2.04	5.05	1.54	0.43

Table 2: Packet loss comparisons.

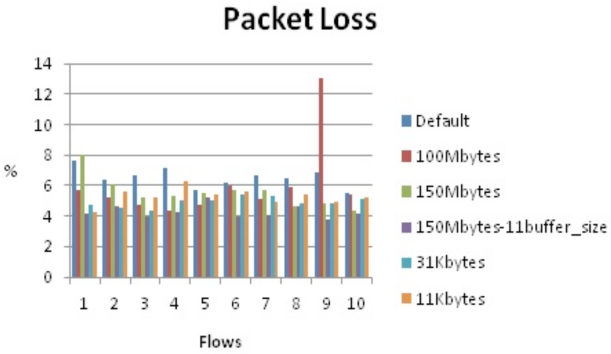

Fig. 3: Packet loss from node C to B in a congested scenario using BAMTAN-adv original.

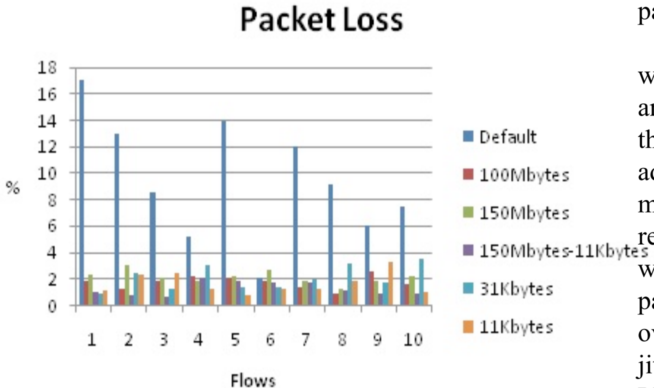

Fig. 4: Packet loss from node C to node B in a congested scenario using the modified BAMTAN-adv.

Throughput						
	1MB 41kB	100M 41kB	150M 41kB	150MB 11KB	1MB 31KB	1MB 11KB
BATMAN-ADV						
A	0.08	0.09	0.08	0.09	0.09	0.08
C	0.09	0.09	0.09	0.09	0.09	0.86
D	0.08	0.09	0.08	0.08	0.08	0.08
BATMAN-ADV modified						
A	0.07	0.06	0.09	0.10	0.07	0.05
C	0.11	0.05	0.05	0.09	0.06	0.06
D	0.06	0.09	0.08	0.09	0.09	0.09

Table 3: Throughput comparisons.

The results show that our metric is well suited for unstable and dynamic networks under strenuous circumstances. The variation of packet latency across a network, known as jitter or packet delay variation (PDV) shows a significant difference between protocol sets. The BATMAN-adv original shows the best (low) PDV of less than 55ms across all variation settings as show in Table 1. The PDV is consistent irrespective of the transfer rate or the buffer size. Node C, which had mobility throughout the tests, exhibits an overall average of 30.80ms across all variation settings while nodes A and D are 38.18 and 42.40, respectively. On the other hand, the modified BATMAN-adv lacks consistency as some points rise abruptly, reaching 336.2ms, while the lowest is 24.24ms in line with the original protocol. PDV consistently increases and appears to do so independently of the variation settings.

The average packet loss results of BATMAN-adv original appear inconsistent in the baseline measurement. The average across all variation settings (i.e. from default 150MB-11KB see Table 2) exhibits some inconsistencies as compared to our modified version which stands at an average of 3.2% for all the links (A = 3.52, C = 3.05 and D = 3.09). BATMAN original has values: A = 2.94, C = 5.42, C 0.83. The most distinctive and significant factor in this case is the consistency of packet loss for BATMAN-adv across all settings while the modification shows reduction as per variation settings. At default, the average packet loss on the three links is about 8%. The loss rate then reduces proportionally to the transfer rate and buffer size. This shows that situation-aware routing metrics perform well on large and inconsistent networks with congested links. The results show no practical relation between jitter/PDV and packet loss.

Unlike packet loss, the consistency in PDV correlates well with the consistency in throughput as shown in Table 1 and Table 3. The average throughput is also independent of the variation settings. The average throughput in BATMAN-adv is consistently at 0.08 MB/s. On the other hand our modified version tends to fluctuates a bit. The maximum recorded throughput in flow for BATMAN-adv is 0.09MB/s while our modified version could reach 3MB/sec in a particular flow but due to its fluctuation tendency, the overall average amounts to 0.76MB/sec. We observe that jitter/PDV and throughput are correlated, i.e. consistent PDV results in a consistent throughput. In terms of throughput, both protocol versions are at par with each.

V. CONCLUSION AND FUTURE WORK

Situation-aware routing seeks to improve QoS on hybrid wireless mesh networks by making routing decisions based on the current situation of the network. BATMAN-adv counts OGMs received as a link quality measurement. We apply a prioritization technique to calculate the link quality metric. We give the more recently received OGMs more weight in deciding the link quality by summing their indices in a given window rather than counting their quantity. Therefore, more recently received OGMs contribute more to the metric in order to give a more precise indication of the current state of a link.

The results show little relation between jitter/PDV and packet loss. Jitter is, however, proportional to throughput. The average throughput achieved on both protocols was almost the same but we noticed that the throughput on our modified version increases as the network grows, and therefore appears to be scalable. Our protocol modification suffered from packet loss at low bandwidth rates but this reduces as the transfer rate increases and buffer size shrinks, i.e. it performs well with congestion. We can infer that increasing the transfer rate with a smaller buffer size is poorly handled by BATMAN-adv original while our modification performs admirably. We conclude that our situation-aware protocol modification shows potential to address issues pertaining to scalable and congested static mesh networks with mobile nodes.

There are some limitations to our experimental design, which possibly had negative impact on the results. The test bed was in a single computer laboratory room with only four nodes. The distance between the nodes was small. Also, there are several other wireless networks accessible in the same room. Although we tried our best to confine our network to free different channel spread, the noted inconsistent packet loss could possibly result from network interference.

In the future, we would like to see how the protocol performs under a wider range of traffic patterns, and also in a more geographically spread mesh network (with more nodes). Our initial plan was to use mobile phones as the mobile nodes but could not because BATMAN-adv has not yet been ported to the mobile phone. Hence our future work can orient toward that scenario.

VI. ACKNOWLEDGEMENTS

We thank Telkom, Cisco, and the THRIP (Technology and Human Resources for Industry Partnership) initiative of the South African Department of Trade and Industry for financial support via the Telkom Centre of Excellence (CoE) programme. THRIP funding is managed by the National Research Foundation (NRF). Any opinion, findings and conclusions or recommendations expressed in this material are those of the authors and therefore the NRF does not accept any liability in regard thereto.

VII. REFERENCES

[1] M. Abolhasan, B. Hagelstein, and J. C. P. Wang, "Real-world performance of current proactive multi-hop mesh protocols," in *Communications, 2009. APCC 2009. 15th Asia-Pacific Conference on*, p. 44–47.

[2] R. Pinto, *WMM-Wireless Mesh Monitoring*, Technical report, INESC-ID, 2009.

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

[4] A. Munaretto, H. Badis, K. Al Agha, and G. Pujolle, "QoS-enhanced OLSR protocol for mobile ad hoc networks," in *Proceedings of ANWIRE 1st International Workshop, Glasgow Scotland*, 2003, p. 171–183.

[5] Y. Yang, J. Wang, and R. Kravets, "Designing routing metrics for mesh networks," in *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[6] M. K. Marina and S. R. Das, "Ad hoc on-demand multipath distance vector routing," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3, p. 92–93, 2002.

[7] E. Royer and C. Perkins , "An Implementation Study of the AODV Routing Protocol.", *In Proceedings of the IEEE Wireless Communications and Networking Conference*, vol. 3, 2000, pp. 1003-1008.

[8] D. B. Johnson, D. A. Maltz, J. Broch, and others, "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks," *Ad hoc networking*, vol. 5, p. 139–172, 2001.

[9] M. Vainio, "Link reversal routing," *Nokia Group*.

[10] S. A. Shaar, F. A. M. Masoud, A. Murad, R. Al-Shalabi, and G. Kanaan, "Analysis of Enhanced Associativity Based Routing Protocol," Journal of Computer Science, vol. 2, p. 853–858, 2006.

[11] B. Divecha, A. Abraham, C. Grosan, and S. Sanyal, "Impact of node mobility on MANET routing protocols models," *Journal of Digital Information Management*, vol. 5, no. 1, p. 19, 2007.

[12] E. M. Royer and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *Personal Communications, IEEE*, vol. 6, no. 2, p. 46–55, 1999.

[13] G. Pei, M. Gerla, and T. W. Chen, "Fisheye state routing in mobile ad hoc networks," in *ICDCS Workshop on Wireless Networks and Mobile Computing*, 2000, p. D71–D78.

[14] S. Wunderlich, M. Lindner and W. Tsai "B.A.T.M.A.N advance" http://www.open-mesh.org/wiki/batman-adv.

[15] L. Iannone and S. Fdida, "Meshdv: A distance vector mobility-tolerant routing protocol for wireless mesh networks," in *IEEE ICPS Workshop on Multi-hop Ad hoc Networks: from theory to reality (REALMAN)*, 2005, p. 103–110.

[16] M. Caleffi and L. Paura, "Bio-inspired link quality estimation for wireless mesh networks," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009*, 2009, p. 1–6.

[17] C. Murphy, "Moving Averages." http://investopedia.com/inv/pdf/tutorials/MovingAverages.pdf. .

[18] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A multi-radio unification protocol for IEEE 802.11 wireless networks," 2004.

Hlabishi I. Kobo obtained a BSc Honours degree in Computer Science from the University of the Western Cape (UWC) in 2009. He is presently studying towards an MSc degree with the Bridging Applications and Networks Group (BANG) at the same institution. His main research interest is now wireless mesh routing protocols on mobile phones.

William D. Tucker is a Senior Lecturer of Computer Science at UWC and leads BANG research there. His research interest includes Internet Protocol (IP) networks and their applications in developing regions.

Xiaoming "Andy" Liu obtained an MSc degree in Computer Science from UWC in 2008. He is now a PhD student interested in IP networks. His research interests include intelligent wireless network management with situation awareness and AI methods, distributed computing, and time series modeling.