

# Deployment and Evaluation of a Wireless Mesh Network

Ali Hamidian<sup>(1)</sup>, Claudio E. Palazzi<sup>(2)</sup>, Tin Y. Chong<sup>(3)</sup>, Juan M. Navarro<sup>(3)</sup>, Ulf Körner<sup>(1)</sup>, Mario Gerla<sup>(3)</sup>

<sup>(1)</sup>Department of Electrical and Information Technology, Lund University,  
Box 118, SE-221 00 Lund, SWEDEN  
(e-mail: ali.hamidian@eit.lth.se, ulf.korner@eit.lth.se)

<sup>(2)</sup>Dipartimento di Matematica Pura e Applicata, Università degli Studi di Padova,  
Via Trieste, 63 - 35131 Padova, ITALY  
(e-mail: cpalazzi@math.unipd.it)

<sup>(3)</sup>Department of Computer Science, University of California, Los Angeles,  
3732F Boelter Hall, 420 Westwood Plaza, Los Angeles, CA, USA 90095-1596  
(e-mail: chongtin@ucla.edu, jmnavarro@ucla.edu, gerla@cs.ucla.edu)

**Abstract**—Wireless Mesh Networks (WMNs) have the potentialities to greatly penetrate the marketplace in the next decade. They can be considered a hybrid between wireless infrastructure and ad-hoc networks; no other network infrastructure is needed other than the mesh points, thus providing flexibility in building and expanding the network, allowing for automatic discovery of neighboring nodes, increased reliability and redundancy. In this work, we discuss how WMNs can be practically employed to support communications in an office-wide area. To this aim, we have deployed a real WMN testbed utilizing state-of-the-art technology and analyzed the performance of this architecture when supporting multihop heterogeneous traffic.

**Keywords**- wireless mesh network; testbed; performance evaluation; QoS

## I. INTRODUCTION

Wireless Mesh Networks (WMNs) are the next step in the evolution of wireless architecture, delivering services for a large variety of applications in personal, local, departmental, campus, and metropolitan areas. Unlike Wireless Local Area Networks (WLANs), mesh networks are self-configuring systems where each Access Point (AP) can relay messages on behalf of others, thus increasing the communication range and the available bandwidth. Moreover, in regular WLANs, the (wireless) AP has to be wired to the infrastructure; this is a paradox overcome by WMNs, where APs can be connected to the rest of the network by wireless radio links only. Other key advantages of WMNs include ease of installation, no cable cost, automatic connection among nodes, network flexibility, discovery of newly added nodes, redundancy, and self-healing reliability.

These characteristics make a WMN an IEEE 802.11-based network of wireless nodes that can be considered a hybrid that tries to combine the functionalities of both WLANs and Mobile Ad-hoc Networks (MANETs). For these reasons, WMNs have gained the interest of researchers all around the world. Indeed, WMNs are an important technology for distributed applications that cannot rely on a

fixed infrastructure, but that may require instant deployment, dynamism, self-organization, and self-configuration.

However, full potentialities and limits of this new technology have still to be explored. To this aim, we have created a testbed for multihop WMNs using the Mesh Connectivity Layer (MCL), which is part of the Microsoft Mesh Toolkit. The ease of use of this off-the-shelf technology makes it suitable to quickly deploy WMNs in various contexts. Evaluating this technology in a real testbed is particularly interesting since its features make it a state-of-the-art, readily available technology. Therefore, both researchers and practitioners may consider MCL to deploy their WMNs. For instance, data collected by our experiments may be of interest to network designers that want to know the effective capabilities of a WMN before investing on its deployment, or may be used to integrate or validate data collected through networking simulations (for instance, when using ns-2 [1]).

More in detail, our testbed can be considered for a scenario such as a home environment, an office, or a small department. In this context, heterogeneous, multimedia applications are generally run (e.g., FTP, VoIP, video streaming); we have hence considered various possible flows running on our testbed, delineating the potentialities and the limits of today's WMN technology.

In essence, the main contribution of this work is twofold: reporting on our experience in deploying a real WMN testbed and performing a preliminary analysis of the performance of this kind of technology in supporting heterogeneous multimedia applications.

The rest of this paper is organized as follows. In the following section, we provide some background information on WMN technology. In Section III, we review the state-of-the-art in WMN testbeds. Section IV is devoted to presenting the main tools and the architecture of our testbed. Our testbed deployment is described in Section V and the outcomes of some experiments we have performed through it are discussed in Section VI. Section VII presents our preliminary Quality of Service (QoS) experiments with IEEE 802.11e technology. Finally, in Section VIII we conclude the paper and identify future directions for this work.

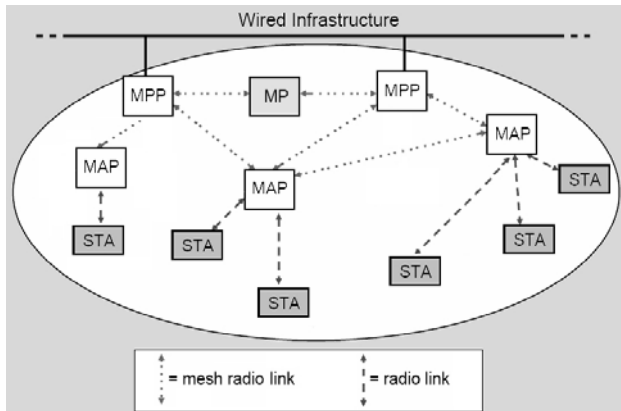


Figure 1. Wireless Mesh Network architecture.

## II. WMN BACKGROUND

A WMN is an IEEE 802.11-based network whose architecture makes it a hybrid wireless network between a WLAN and a MANET. In essence, a WMN is able to extend the coverage of the network infrastructure by multihop wireless connections between APs. The APs in a WMN can hence be detached from any wired infrastructure while being connected to each other through wireless links.

As shown in Fig. 1, nodes in a WMN can be classified into four main categories:

- *Mesh Point (MP)* – A device that provides mesh services and that may be a dedicated infrastructure device or a regular user device with the ability to fully participate in the operation of a mesh network.
- *Mesh Access Point (MAP)* – A special MP that also provides AP services.
- *Mesh Portal Point (MPP)* – A special MP that serves as a gateway to a wired network; it supports transparent bridging, address learning and bridge-to-bridge communication.
- *Station (STA)* – A completely mobile user device that does not participate in mesh services; it can communicate with other stations via an AP, a MAP, or an MPP.

In essence, WMNs are composed of MPs that facilitate the connectivity and intercommunication of wireless clients through multihop wireless paths. WMNs may be connected to the Internet through an MPP, whereas the MPs function as wireless bridges within the WMN. Therefore, if we compare WMNs and legacy ad-hoc networks (e.g., MANETs), end hosts and routing nodes are distinct in a WMN, whereas every node can play both roles in ad-hoc networks. Furthermore, MPs in WMNs are often stationary and not power-constrained, thus freeing routing protocols from the burden of dealing with mobility and power constraints.

A peculiar feature of WMNs is that, if the source and the destination are not in the same Basic Service Set (BSS) domain, the source MAP does not forward packets to all the MAPs in the Extended Service Set (ESS); instead, the packets are sent via MPs to reach the destination [2]. A WMN can be viewed as a multihop, ad-hoc, packet

switching, and forwarding network between MPs in the same ESS. The Wireless Distribution System (WDS) uses an extension of the IEEE 802.11 MAC/PHY, named IEEE 802.11s, to provide a protocol for self-configuring paths among MPs in a multihop topology, supporting broadcast, multicast, and unicast traffic [3].

Generally, the primary purpose of the WMN is to create a low-cost, easily deployable, high performance wireless coverage throughout the building, eliminating radio frequency dead-spots. WMNs can be useful network architecture where Ethernet cabling does not exist or its installation is economically prohibitive. Examples include small and large offices, manufacturing plants, university campuses, government buildings, and health care centers/hospitals.

The aim of WMN technology is to provide capabilities that can facilitate the deployment of multihop wireless networks with access to the Internet. Yet, an evaluation of the performance of this new technology is needed; to this aim, we have created a testbed for multihop WMNs using state-of-the-art, off-the-shelf technology. Our experiments shed new lights on the real potentialities and limits of WMNs employing today's technology.

## III. RELATED WORK

Other researchers around the world are currently building WMN testbeds. In this section we review the most interesting among these attempts.

The University of California, Santa Barbara Mesh Testbed is an experimental wireless mesh network consisting of nodes equipped with multiple IEEE 802.11a/b/g wireless radios and distributed on various floors of a campus building [4]. The aim of this testbed is that of testing new protocols supporting robustness in multihop wireless networks.

The Broadband and Wireless Network (BWN) Lab at Georgia Institute of Technology has recently built a WMN testbed [5], [6]. The testbed includes 15 IEEE 802.11b/g-based mesh routers, some of which are connected to the Internet. The testbed is distributed among various rooms on a single floor and laptops are used as mobile stations. The main aim of this WMN testbed is that of investigating the effects of inter-router distance, backhaul placement, and clustering; recent research has been devoted to study adaptive protocols for transport layer, routing and MAC layers, and their cross-layer design.

MAP at Purdue is an experimental WMN testbed composed by several nodes capable of running in both 802.11a and 802.11b/g mode. The purpose of the testbed is mainly that of studying routing problems and solutions to create high throughput routes [7], [8]. Furthermore, MAP includes a mobile ad-hoc network testbed consisting of 5 laptops and 16 Compaq IPAQ PDAs.

A general purpose WMN testbed using Intel IXP425 series XScale network processors as routers and iPAQ as clients has been built at Carleton University [9]. Two WLAN network interfaces are installed on the two Mini-PCI slots; one is a Prism 2/2.5 card, which supports IEEE 802.11b, and the other is an Atheros card, which supports IEEE 802.11a/b/g.

The main aim of our testbed is that of evaluating state-of-the-art, off-the-shelf technology for WMNs. In particular, as opposed to the aforementioned testbeds, our aim is to evaluate interactive and multimedia applications over a WMN. Our testbed is composed by one server, one MPP, three MPs, one MAP and a variable number of stations. Another important difference between the aforementioned testbed and ours is the adopted core technology: we utilize Microsoft's MCL and network cards supporting the IEEE 802.11a/b/g/e standards. In particular, Wi-Fi Multimedia (WMM) capabilities of the network interface that we have employed allow us to evaluate Quality of Service (QoS) solutions; indeed, this kind of solutions is particularly interesting for interactive and multimedia applications.

#### IV. AN OFF-THE-SHELF MESH NETWORK ARCHITECTURE

In this section we describe the main software tool utilized for our testbed and motivate our choice.

Microsoft's MCL tool allows for the deployment of a WMN using any wireless card [10]. Simply, as a native Windows driver, upon installation the host system can see a virtual network adapter that allows for direct connectivity to the wireless mesh network. More specifically, MCL is an interlayer protocol, architecturally located between the network layer and the link layer complementing them in a transparent way. System changes are hence minimized, allowing for the use of regular technologies and protocols, while introducing the mesh connectivity feature.

Another interesting feature of MCL is that its routing protocol, Link Quality Source Routing (LQSR), is a modified version of Dynamic Source Routing (DSR [11]) and its basic functionalities are as follows:

1. it identifies all the MPs in a WMN and assigns relative weights to the links among the nodes;
2. it determines the channel, the bandwidth, and the loss rate for every possible link and spreads this information to all nodes;
3. the aforementioned information is exploited to compute a routing metric called Weighted Cumulative Expected Transmission Time (WCETT [12]), which defines the best data transmission path from a given source to a given destination;
4. if the optimum path between a particular source and destination changes, the route is modified accordingly, without interrupting the link between the nodes.

Finally, MCL is surprisingly provided by Microsoft as an open source tool, allowing anyone to modify its code and testing alternative solutions to any of its components.

For all of these characteristics, MCL can be considered as a tool for quick and easy deployment of WMNs; in other words, an off-the-shelf option for mesh networking. This is also the main reason why we have used this tool for our work: we wanted to create an evaluation testbed of current technologies that any house owner or office director can choose to quickly deploy a WMN and, on top of it, analyze the behavior of most popular applications.

#### V. WMN TESTBED DEVELOPMENT

In this section we describe the experimental testbed that we have set up to evaluate a WMN supporting interactive multimedia applications. We have built our WMN testbed at Boelter Hall, UCLA campus, in the immediate vicinity of the Network Research Laboratory. Boelter Hall is a square-shaped building with an open area in the middle. The map in Fig. 2 provides a bird-eye view of our network topology setup, whereas Fig. 3 shows two pictures of the actual testbed taken from different angles.

A total of five mesh-capable devices are part of our WMN; three MPs, one MAP, and one MPP. The MPP is connected to an FTP and video streaming server, whereas a variable number of STAs (clients) are connected to the MAP. The MPs on the mesh backbone are operating on channel 11, while the STAs are communicating with the MAP on channel 1. The rationale behind this choice is that of keeping these two channels far from each other so as to decrease the inter-carrier interference that could affect experimental results and their clarity.

The mesh backbone is implemented with Dell Latitude D610 Review laptops (Pentium M 760 2.00 GHz, 512 MB RAM), whereas the STAs are Dell laptops with Pentium III CPU and 128 MB of RAM. All nodes utilize ZyXEL AG-225H as Network Interface Card.

Distances among nodes have been chosen so that non-neighboring nodes cannot communicate directly with each other. For instance, MP1 in Fig. 2 cannot communicate directly with MP3 unless the packets are routed through

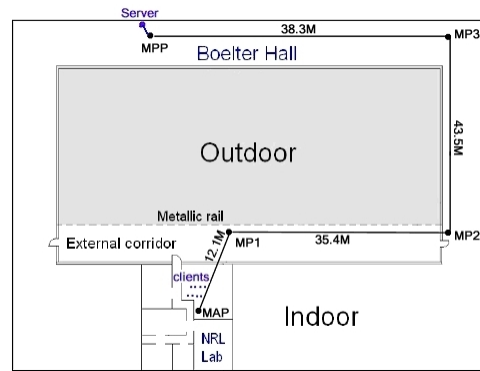


Figure 2. Testbed map.

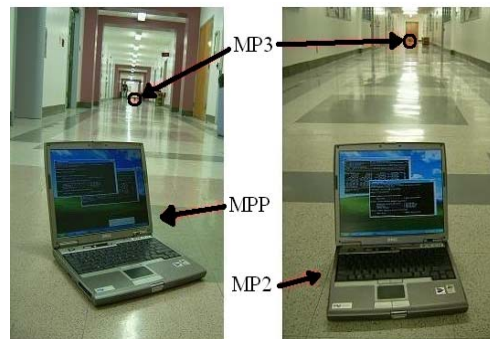


Figure 3. Testbed snapshots.

MP2. To this aim we have run preliminary tests through tools such as ping and traceroute to find the distances at which non-neighboring nodes could not reach each other: to avoid data packets using a decreased number of hops and bias our results.

In our tests different multimedia applications are run solo or together to evaluate their performance on the WMN. We considered VoIP streams, each corresponding to 64 kbps of voice traffic (G.711 voice codec [13]); each generated voice packet carries a payload of 160 bytes. Moreover, we have also considered video stream flows: an MPEG video with a bit rate varying between 218 and 456 kbps.

The VoIP traffic was generated using the Distributed Internet Traffic Generator (D-ITG) [14], while we used VLC media player [15] for video streaming using RTP, and FileZilla [16] for file transfer using FTP. We consider different kind of applications in our experiments but we keep the focal point on the performance achieved by multimedia applications, as they represent the most interesting and challenging service for the considered scenario.

## VI. THE TESTBED IN ACTION

In this section we report on preliminary experimental outcomes achieved through our WMN testbed. The purpose of these experiments is both to assess the functioning of our testbed and to do a preliminary evaluation of the performance of heterogeneous flows over it.

As our first experiment, we run a single FTP/TCP download session over our WMN, varying the number of hops that packets have to traverse from the source (the FTP server) to the destination (the FTP client); the client has to download a 17.3 MB file. The client was positioned as depicted in Fig. 2 and engaged with the MAP, whereas the position of the server varied: on MAP to have the 1-hop evaluation, on MP1 to have the 2-hops evaluation, on MP2 to have the 3-hops evaluation, etc. When not differently stated, the server was located where shown in Fig. 2

Results of this experiment are reported in Fig. 4. As the figure shows, the time to download the file increases linearly with the number of hops the flow has to traverse. This is not surprising and indicates that our testbed is performing correctly. Indeed, it is well-known in scientific literature that the available data rate for TCP-based flows decreases for each wireless hop until becoming unable to support any application after a certain “ad hoc horizon” [17]-[19].

As a second experiment, we consider a scenario with several users simultaneously engaged in voice chatting with each other and one background FTP session between two nodes that are at 2 hops of distance from each other.

In this scenario we study the impact of changing the number of hops traversed by the VoIP streams; clearly, the more hops the VoIP streams traverse, the more is the impact on their performance. This can be seen in Fig. 5, which shows the average packet reception rate (PRR), i.e., the goodput in terms of packets/s successfully received at the destination, as experienced by the 10 VoIP streams when varying the number of hops. The PRR of the VoIP streams decreases when the number of hops increases. This has a

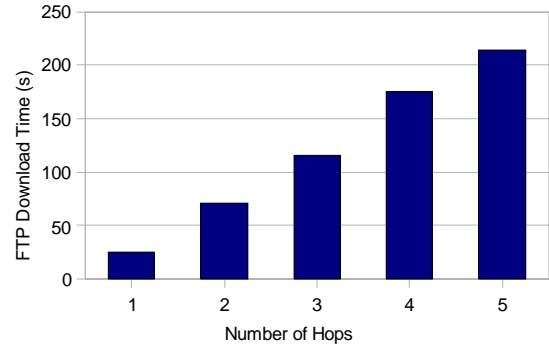


Figure 4. Download time for a 17.3 MB file through FTP.

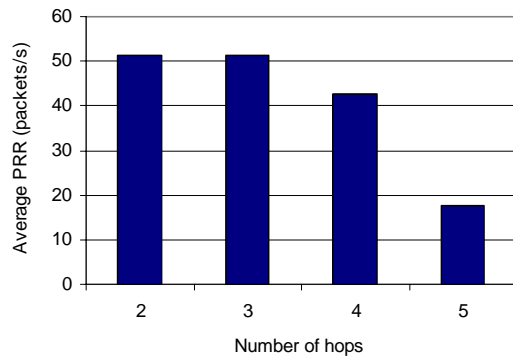


Figure 5. Average packet reception rate (PRR) experienced by 10 VoIP streams with a background FTP flow.

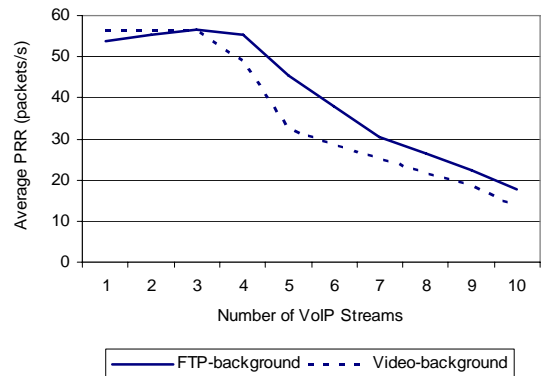


Figure 6. Average packet reception rate (PRR) experienced by a variable number of VoIP streams with a background FTP or video flow.

devastating effect on the perceived quality of the VoIP application: voices will result severely scattered if the distance between the participants is larger than 3 hops.

As third experiment, we consider a testbed configuration where the background traffic is either represented by an FTP flow or by a video stream. In both cases, this background traffic is traversing 2 hops between the FTP/video streaming server (connected to MPP) and MP2. Simultaneously, a

variable number, from 1 to 10, of VoIP streams traverse all 5 hops of the WMN. Again, the PRR for the VoIP streams is measured and its values reported in Fig. 6. As expected, the performance worsens when increasing the number of simultaneous VoIP sessions as the various flows interfere with each other and with the background traffic. By looking at the chart we can conclude that with the considered configuration only up to 3-4 simultaneous VoIP streams can be effectively supported over the WMN.

## VII. PRELIMINARY QOS EXPERIMENTS

The notion of having prioritized traffic on a WMN seems a viable solution for our purposes. Wireless MultiMedia (WMM) technology, a subset of the Enhanced Distributed Channel Access (EDCA) specified in IEEE 802.11e, provides QoS support to IEEE 802.11 wireless networks. QoS technologies can be used to prioritize delay-sensitive real-time traffic (i.e., voice or video), while throttling delay-insensitive traffic (i.e., bulk data transfers).

In WMM, four Access Categories (ACs) are used to prioritize traffic at the Medium Access Control (MAC) sublayer: AC\_VO (for voice traffic), AC\_VI (for video traffic), AC\_BE (for best effort traffic), and AC\_BK (for background traffic). However, WMM does not provide guaranteed QoS, but only service differentiation. This service differentiation is realized by assigning different values to the following contention parameters:

1. *AIFSN (Arbitration InterFrame Space Number)*: denotes the number of time slots to defer, after SIFS duration, before either starting a transmission or invoking a backoff procedure. By assigning a low value to AIFSN, an AC is given a high priority.
2. *CWmin* and *CWmax*: denote the minimum and maximum Contention Window (CW) size used to calculate the number of time slots that have to be waited before accessing the medium. By assigning low values to CWmin and CWmax, an AC is given a high priority.
3. *TXOP (Transmission Opportunity) limit*: denotes the maximum duration of a TXOP, which is a bounded time interval during which a station has the right to transmit. By assigning a high value to TXOP limit, an AC is given a high priority as it will be able to transmit more packets. If this parameter is set equal to zero, stations are allowed to transmit one frame only; otherwise, stations can transmit multiple frames as long as the duration of the transmission does not extend beyond the TXOP limit.

Default values for these parameters are shown in Table I. In order to prioritize a traffic stream, the Differentiated Services Code Point (DSCP) value must be modified by the application. This value is contained in the 6 least significant bits in the Differentiated Services (DS) field, which supersedes the Type of Service (TOS) field in the IPv4 header and the Traffic Class field in the IPv6 header. Once the DSCP value is set in the IP header, the packet is delivered to the MAC sublayer, where it is placed into a priority queue by WMM. To this aim, Table II shows the basic prioritization mapping between DSCP/TOS and ACs.

TABLE I. DEFAULT EDCA PARAMETERS

AC	CWmin	CWmax	AIFSN	TXOP Limits (ms)
AC_BK	15	1023	7	0
AC_BE	15	1023	3	0
AC_VI	7	15	2	3.008
AC_VO	3	7	2	1.504

TABLE II. MAPPING OF WMM PRIORITIES

Type	WMM_AC	Priority	TOS	DSCP
best effort	AC_BE	0	0	0
background	AC_BK	1	32	8
background	AC_BK	2	64	16
best effort	AC_BE	3	96	24
video	AC_VI	4	128	32
video	AC_VI	5	160	40
voice	AC_VO	6	192	48
voice	AC_VO	7	224	56

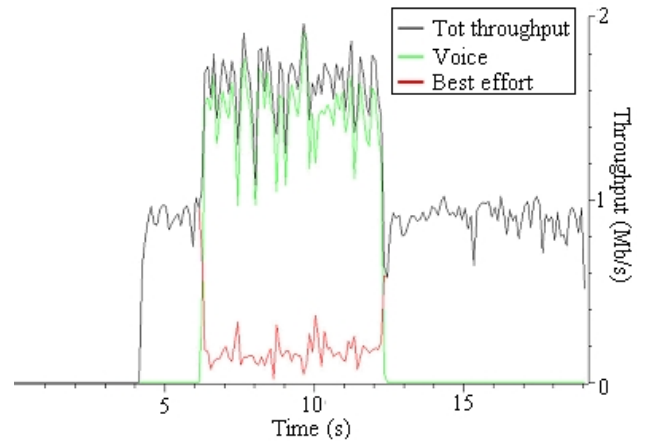


Figure 7. WMM experiments with voice (green) and best effort (red) streams. The black curve shows the sum of the throughput of both streams.

We have hence run a preliminary QoS experiment with integrated Intel PRO/Wireless 3945ABG Network Connection wireless cards on three laptops (say A, B, and C) operating in ad hoc mode. In the experiments, we used the default values of the wireless cards, i.e., channel 11, 802.11b/g, ad hoc power management disabled, WMM enabled, HD mode disabled (a feature reducing interference in environments with several nearby APs), CTS-to-self enabled, power management optimized for maximum battery life, balanced setting between roaming and performance, throughput enhancement disabled, and highest transmit



power optimized for maximum coverage. The outcome has been captured using the network protocol analyzer WireShark and is shown in Fig. 7. Basically, two streams are simultaneously present on the channel: a best effort stream (the red curve) and a prioritized voice stream (the green curve). The black curve denotes the sum of the throughput of both streams.

In this scenario, the three stations are positioned in a straight line, equally distanced, with B in the middle. Both nodes A and C send data to B. First, we start a best effort stream from node A to B starting at 4 s and going on until 6 s. Since this is the only active stream during this period, its throughput curve (in red) overlaps with the curve representing the total throughput (in black). We can see that the throughput of the best effort stream is around 1 Mbps. Then, at 6 s, a voice stream is started from node C to B. This is depicted by the green curve going on approximately from 6 s to 12 s. The figure shows that the throughput of the best effort stream falls to around 200 kbps while the voice stream gets around 1.5 Mbps. This is obviously because WMM succeeds in prioritizing the high-priority voice stream and transmit its data frames using AC\_VO. The low-priority best effort stream, on the other hand, is transmitted using AC\_BE. This results in longer waiting times before trying to access the wireless medium and no possibility to transmit more than a single data frame; or in other words, lower throughput. At 12 s, the voice stream is stopped and the best effort stream recaptures the available bandwidth of the channel, again climbing up to 1 Mbps.

### VIII. CONCLUSION

In this work, we have reported on the deployment of a real wireless multihop testbed utilizing WMN technology based on Microsoft MCL. We have hence created our testbed using open source software and hardware technology that are available to any house owner or office director that decided to exploit the advantages of a WMN.

Furthermore, we have utilized our testbed to provide an evaluation of the factual capability of this technology in supporting multimedia applications, which represent the most popular and challenging applications among users. Based on results presented in this paper, we can claim that today's WMN technology is promising even if performance quickly deteriorates when increasing the number of wireless hops between the source and the destination, becoming hardly satisfying after three wireless hops. Yet, a wise design of the network architecture that limits the number of consecutive wireless hops may generate a WMN able to satisfy the users' needs.

Finally, our testbed also allowed us to study many practical issues that inspired us to identify future directions for this work, such as fully integrating QoS solutions in our testbed and testing the performance of the WMN when supporting specific, challenging applications like online games and augmented reality services. Since we are convinced that interactive applications and WMNs can represent a very effective combination, we are motivated in testing specific solutions for these kinds of applications on our testbed. We are now aware that our testbed represents a

useful tool to evaluate complex and realistic scenarios: by integrating WMNs with WMM technology, we are able to provide different priorities to traffic flows at the MAC sublayer, depending on the access category (i.e., voice, video, best effort, and background): this may be an essential feature worth evaluating for complex interactive applications.

We are hence aimed at exploiting the testbed to assess the performance of various challenging and popular applications such as online games, cooperative multimedia management, interactive audio/video, and augmented reality services [20]-[24].

### REFERENCES

- [1] The Network Simulator - ns-2, [Online]. Available: <http://www.isi.edu/nsnam/ns>
- [2] IEEE, "802.11-2007, IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Jun 2007.
- [3] IEEE, "Draft Amendment: ESS Mesh Networking," IEEE P802.11s Draft 2.0, Mar 2008
- [4] K. Ramachandran, K. Almeroth, E. B. Royer, "A Novel Framework for the Management of Large-scale Wireless Network Testbeds," in Proc. of the 1st International workshop on Wireless Network Measurements (WinMee 2005), Trento, Italy, Apr 2005.
- [5] V. C. Gungor, P. Pace, E. Natalizio, "AR-TP: An Adaptive and Responsive Transport Protocol for Wireless Mesh Networks," in Proc. of IEEE ICC 2007, Glasgow, Scotland, UK, Jun 2007.
- [6] K. R. Chowdhury, I. F. Akyildiz, "Cognitive Wireless Mesh Networks with Dynamic Spectrum Access," IEEE Journal on Selected Areas in Communications, vol. 26, no. 1, pp. 168-181, Jan 2008.
- [7] S. M. Das, H. Pucha, K. Papagianakki, Y. C. Hu, "Understanding Wireless Routing Link Metric Dynamics," in Proc. of the 7th ACM SIGCOMM/USENIX Internet Measurement Conference (IMC 07), San Diego, CA, USA, Oct 2007.
- [8] D. Koutsonikolas, J. Dyaberi, P. Garimella, S. Fahmy, Y. C. Hu, "On TCP Throughput and Window Size in a Multihop Wireless Network Testbed," in Proc. of ACM Mobicom International Workshop on Wireless Network Testbeds, Experimental evaluation and Characterization (WiNTECH 07), Montreal, QC, Canada, Sep 2007.
- [9] Carleton University, Wireless Mesh Networking, [Online]. Available: <http://kunz-pc.sce.carleton.ca/MESH/index.htm>
- [10] Microsoft Research, Self-Organizing Neighborhood Wireless Mesh Networks, [Online]. Available: <http://research.microsoft.com/mesh/>
- [11] D. B. Johnson, Y. Hu, D. A. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," RFC 4728, Feb 2007.
- [12] R. Draves, J. Padhye, B. Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in Proc. of MobiCom 2004, Philadelphia, PA, USA, Sep 2004.
- [13] ITU-T Recommendation G.711, "Pulse Code Modulation (PCM) of Voice Frequencies," 1988.
- [14] A. Botta, A. Dainotti, A. Pescapè, "Multi-protocol and Multi-platform Traffic Generation and Measurement," in Proc. of INFOCOM 2007, Anchorage, AK, USA, May 2007.
- [15] VLC media player, [Online]. Available: <http://www.videolan.org/vlc/>
- [16] FileZilla, [Online]. Available: <http://filezilla-project.org/>
- [17] M. Gerla, K. Tang, R. Bagrodia, "TCP Performance in Wireless Multi-hop Networks," in Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99), New Orleans, LA, USA, Feb 1999.

- [18] G. Holland, N. Vaidya, "Analysis of TCP Performance over Mobile Ad Hoc Networks," in Proc. of MobiCom'99, Seattle, WA, USA, Aug 1999.
- [19] C. Tschudin, P. Gunningberg, H. Lundgren, E. Nordström, "Lessons from Experimental MANET Research," Ad Hoc Networks Journal, vol. 3, no. 2, pp. 221–233, Mar 2005.
- [20] S. Ferretti, M. Rocchetti, A. La Penna, "Fast Synchronization of Mirrored Game Servers: Outcomes from a Testbed Evaluation," in Proc. of the International Symposium on Intelligence Techniques in Computer Games and Simulations, Shiga, Japan, Mar 2007.
- [21] S. Ferretti, S. Mirri, L. A. Muratori, M. Rocchetti, P. Salomoni, "Cooperative Multimedia Management for Participative Learning: A Case Study," New Review of Hypermedia and Multimedia, Taylor & Francis, vol. 14, no. 2, pp. 177–197, Dec 2008.
- [22] M. Furini, "Fast Play: A Novel Feature for Digital Consumer Video Devices," IEEE Transaction on Consumer Electronics, IEEE Consumer Electronics Society, vol. 54, no. 2, pp 513–520, May 2008.
- [23] M. A. Alberti, D. Maggiorini, P. Trapani, "A Semiotic Approach to Narrative Manipulation," in Proc. of the 29th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2002), San Antonio, TX, USA, Jul 2002
- [24] A. Ploss, S. Wichmann, F. Glinka, S. Gorlatch, "From a Single- to Multi-Server Online Game: A Quake 3 Case Study Using RTF," in Proc. of ACM ACE 2008, Yokohama, Japan, Dec 2008.

## APPENDIX

We have developed various tools to ease the performance evaluation of mesh networks and technologies. It is our hope that these tools may be valuable to continue WMN research; in public research spirit, we have hence decided to make them available online at <http://www.math.unipd.it/~cpalazzi/mesh.html>.

We briefly describe in the following those that helped us with the toughest part of our experiment: the preliminary QoS evaluation described in Section VII.

- **qostest.exe** – This is a program written in C# that can behave as either a sender or receiver. It sets various socket options to either enable or disable QoS settings, such as the DSCP field in the IP header; it was hence utilized to perform the preliminary QoS evaluation in Section VII. When configured as sender, the sending rate can be specified by calculating the ratio of the packet size to the sending rate (taken as arguments). When configured as receiver, it can receive an indefinite number of packets: a timeout can be specified to eventually block the receive call. Upon timing out, the receiver will print statistics that are specific to a certain run.
- **parsedotnet.pl** – This is a Perl script that parses the output of the qostest.exe program for several runs, and prints summarized statistics. It takes a filename as an argument: it assumes the file contains data for several runs of qostest.exe, and then proceeds to calculate averages based on the run information. The output of this program can then be easily loaded into

tools such as MATLAB to ease the plotting and analysis process.

However, enabling QoS settings via software is not enough: it is necessary to activate the IEEE 802.11e EDCA also via hardware. To this aim, we have to point out that finding a wireless card that could support EDCA in ad-hoc mode has not been an easy task; in fact, most commercial cards have support for EDCA in infrastructure mode only. Even worse, some vendors promise support for EDCA also in ad-hoc mode without fulfilling this promise. It should also be mentioned that the Wi-Fi Alliance has an optional certification testing support for multimedia content over Wi-Fi networks – WMM, but this program checks EDCA support in infrastructure mode only and not in ad-hoc mode. In other words, EDCA support in ad-hoc mode is totally up to the vendor to be implemented. The only wireless cards we have found that could use EDCA in an ad-hoc mode are:

1. Intel PRO/Wireless 3945ABG Network Connection;
2. Intel Wireless WiFi Link 4965AGN.

As stated in Section VII, we utilized the former for our experiments. Once having a wireless card that supports EDCA in ad-hoc mode, this feature has to be enabled in the hardware. To this aim, in the following we report sketch instructions on how to do it.

### 1. Set up an ad-hoc network:

Start > Control Panel > Network Connection > Wireless Network Connection > Properties > Wireless Networks > Add...

- Network name (SSID): MyAdHocNetwork
- Network Authentication: Open
- Data Encryption: Disabled
- Click the box "This is computer-to-computer (ad-hoc) network; wireless access points are not used"

### 2. Set a static IP address:

Start > Control Panel > Network Connection > Wireless Network Connection > Properties > Internet Protocol (TCP/IP) > Properties > Use the following IP address

- IP address: 192.168.10.1
- Subnet mask: 255.255.255.0

### 3. Enable the use of the IP\_TOS socket option:

Start > Run > regedit > HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters > Edit > New > DWORD Value

- Create and set a new DWORD registry value: DisableUserTOSSetting=0

### 4. Enable WMM in ad-hoc mode:

Start > Control Panel > Network Connection > Wireless Network Connection > Properties > Configure... > Advanced > Ad Hoc QoS Mode