Hardware and Software Solutions for Wireless Mesh Network Testbeds

Roberto Riggio, Daniele Miorandi, and Imrich Chlamtac, CREATE-NET

Nicola Scalabrino and Enrico Gregori, Italian National Research Council (CNR) — IIT

Fabrizio Granelli, University of Trento

Yuguang Fang, University of Florida

The work of R. Riggio, N. Scalabrino, D. Miorandi, E. Gregori and I. Chlamtac was partially supported by MIUR in the framework of the Wireless Mesh Network for Next-Generation Internet (WING) project. The work of Y. Fang was partially supported by the U.S. National Science Foundation under grant CNS-0721744 and the National Science Council (NSC), R.O.C., under the NSC Visiting Professorship with contract number NSC-96-2811-E-002-010. The work of F. Granelli was partially funded by MIUR in the framework of the Wireless 802.16 Multi-Antenna Mesh Networks (WOMEN) project.

ABSTRACT

Wireless mesh networks are currently emerging as a promising technology for broadband ubiquitous networking. Typical applications range from home broadband Internet access and community networking to wireless metropolitan area networks. Many companies are currently active in this field with proprietary solutions mostly based on the IEEE 802.11 family of standards. At the same time, there are considerable efforts in academia aiming to provide real-world prototypes and testbeds based on open source software and off-the-shelf technologies. This article provides a survey on the most relevant hardware and software platforms that can be used to build a WMN testbed.

INTRODUCTION

Wireless mesh networks (WMNs) [1, 2] are currently receiving considerable interest in both industrial and academic environments. Many companies are active in this field, developing solutions mostly based on the IEEE 802.11 family of standards [3]. However, while commercial deployments are characterized by proprietary approaches, there are significant efforts in the academic world to provide real-world prototypes and testbeds based on open source software and off-the-shelf technologies. In parallel, as described later in the article, relevant efforts from the IEEE 802 working group are aiming to support the networking paradigm within the well-known IEEE 802.11 and IEEE 802.16 standards.

Shielded from the hazards of a live or production environment, testbeds provide rigorous, transparent, and replicable testing conditions. Measurements run over testbeds can be exploited by the scientific community in order to evaluate the performance of newly developed protocols, providing important guidelines for the design of innovative solutions.

In this article we provide a survey on the most relevant hardware and software platforms that can be used to build a WMN testbed. It is not the authors' intention to provide an exhaustive survey on all platforms suitable for a WMN deployment; instead, we concentrate on open source software and off-the-shelf devices. Unlike [1, 2], where the authors survey existing technologies and research challenges in the WMN scenario, our goal is to provide guidelines, useful for both researchers and practitioners, on the state-of-the-art hardware and software solutions for engineering a WMN testbed.

The remainder of this article is organized as follows. We introduce the WMN paradigm, including network protocols, architectures, and current standardization efforts. We survey hardware platforms and operating systems suitable for WMN deployments. Software solutions implementing a WMN with layer 3 and layer 2.5 routing are reported, respectively. Particular emphasis has been given to implementations based on open source code. Finally, we conclude the article with some remarks on WMN testbed engineering.

WIRELESS MESH NETWORKS

A WMN [1, 2] consists of a set of communication nodes, interconnected via wireless links possibly using multiradio technologies. It allows for continuous connections and reconfiguration around broken or blocked paths by "hopping" from node to node until the destination is reached. WMNs share many features with the conventional ad hoc networking paradigm, particularly self-healing and self-configuring capabilities. Although WMNs can serve as standalone communication systems for disaster recovery or public safety, in this article we focus on *access network* applications. In this scenario a distinction exists in terms of logical roles supported by the physical devices:

- *Relay*: building the multihop wireless backhaul by establishing wireless links between nodes
- *Gateway*: interfacing the WMN with another network, typically the Internet
- *Access point*: providing wireless connectivity to clients
- *Client*: gaining network access for end users

Nodes providing relaying/access functionality are generally computationally powerful devices with no constraints on power consumption and possibly supporting multichannel communication. These

 0163-6804/08/$25.00 © 2008 IEEE

nodes are generally called *mesh routers* as opposed to end-user devices, generally referred to as *mesh clients*. Mesh routers can also act as gateways, providing the WMN with Internet connectivity.

TECHNOLOGIES

In principle WMNs could interface, through suitable gateway nodes, with networks based on different radio technologies (third generation [3G], WiFi, Bluetooth, WiMAX, etc.). However, most actual solutions, in both academic and business environments, heavily rely on the IEEE 802.11 family of standards. This is, to a large extent, because of both the availability of low-cost equipment on the market and the ad hoc features already present in such protocols, making it possible to obtain a mesh configuration with some rather simple modifications. This section describes the most relevant standardization efforts by IEEE in order to support the wireless mesh networking paradigm in its 802 family of protocols.

IEEE 802.11 — The IEEE 802.11s Task Group plans to integrate mesh networking services and protocols within the 802.11 medium access control (MAC) layer. The resulting systems will be compatible with IEEE 802.11 infrastructure mode. In this standard, peer-to-peer layer 2 (L2) links among multiple IEEE 802.11 mesh points can be established to enable direct or multihop data delivery for higher throughput and range extension.

IEEE 802.16 — The IEEE 802.16 first release accounts for a scenario with no mobility and operations in licensed frequency bands ranging from 10 to 66 GHz. Later amendments (IEEE 802.16-2004) extend the standard to non-line-of-sight applications in the 2–11 GHz band. Additional releases encompass mobility (IEEE 802.16e) and improved quality of service (QoS) (IEEE 802.16g). Multihop relaying will be provided by IEEE 802.16j.

ROUTING

WMNs share a number of features with ad hoc networks [4]. In particular, WMNs are characterized by self-organization and self-healing capabilities, and exploit multihopping to build a wireless backhaul for delivering Internet connectivity to end users. As a result, many routing protocols developed for mobile ad hoc networks (MANETs) have been adapted to fit mesh environments. Particular attention has been devoted to the introduction of novel routing metrics capable of achieving better performance in outdoor deployments by considering the wireless channel characteristics [5]. Routing protocols developed for MANETs are generally classified as *proactive*, *reactive*, and *hybrid*. This section summarizes the main features of each category. For a comprehensive survey, readers are referred to [4].

Proactive Routing — Proactive protocols attempt to continuously evaluate all the routes within a network so that when a packet needs to be forwarded, the route is already known and ready to use. Early applications of proactive routing schemes were based on distance vector routing (DVR) protocols exploiting the distributed Bellman-Ford (DBF) algorithm for computing the shortest path in a weighted graph representing the network. Destination-Sequenced Distance-Vector (DSDV) [6] is a routing protocol for ad hoc networks based on the DBF algorithm. As opposed to DVR protocols, link state routing (LSR) protocols react more quickly to connectivity changes. Network traffic is also lower because only information about neighbors is circulated instead of the entire routing table. The main disadvantage of LSR is that it requires more storage and computing resources than DVR. The need to improve convergence performance and reduce control traffic led to the development of improved path finding algorithms that combine the features of DVR and LSR protocols. Optimized Link State Routing (OLSR) [4] is an example of such a protocol. OLSR is an optimized version of traditional link state routing protocols such as Open Shortest Path First (OSPF). It uses the concept of multipoint relays (MPRs) to efficiently disseminate link state updates across the network. Only the nodes selected as MPRs are allowed to generate link state updates.

Reactive Routing — Reactive routing protocols invoke a route discovery procedure on demand. Reactive route discovery is usually based on a query/reply exchange, where a flood-based process is used to reach the desired destination. The main disadvantages of such an approach are:

- The initial delay for route discovery
- The potential scalability problems related to the use of flooding

The Dynamic Source Routing (DSR) [7] and Ad Hoc On-Demand Distance Vector (AODV) [8] protocols can be used to unicast the route reply back to the querying source along a path constructed during the route query phase. In the case of DSR the routing information is accumulated in the query packet, and the complete sequence of nodes on a path to the destination is recorded and returned to the source to be used for source routing of the actual user data. AODV, on the other hand, distributes the discovered route in the form of next-hop information stored at each node in the route.

Hybrid — Protocols that belong to this class leverage both proactive and reactive techniques in order to determine the best path between any pair of nodes. The Hazy Sighted Link State (HSLS) routing protocol [9] was designed to scale in networks with over thousands of nodes, where it outperforms most of the best-known routing algorithms. The protocol exploits both proactive and reactive link state routing to limit network updates in space and time. Unlike traditional methods, HSLS does not flood the network with link state information and attempts to cope with moving nodes that change connections with the rest of the network.

HARDWARE PLATFORMS

From the hardware viewpoint, a wireless mesh router consists of a computer, one or multiple wireless network interface controllers (NICs), an enclosure, an antenna, and all the necessary cable and mounting equipment. A more detailed

A WMN consists of a set of communication nodes, interconnected via wireless links using possibly multi-radio technologies. It allows for continuous connections and reconfiguration around broken or blocked paths by "hopping" from node to node until the destination is reached.

The Orbit project aims at building a laboratory testbed designed to achieve reproducibility of experimentation. Orbit exploits a large two-dimensional grid of 400 nodes, each of which is equipped with two IEEE 802.11 radios.

checklist heavily depends on research directions and reference deployment scenarios in terms of network type and size, expected users, and budget. Finally, a suitable operating system (OS), which manages the hardware and software resources of a mesh node, must be chosen. It is worth pointing out that the choice of OS should not be the driving factor in testbed development. Instead, it must act as the "glue" between the hardware and software requirements.

Companies like BelAir, Tropos, and Strix provide network engineers with vertical solutions for wireless mesh networking, from network planning to network management. However, such solutions are based on proprietary technologies and adopt radically different approaches and protocols, making interoperability problematic.

This section reviews the most relevant hardware platforms for WMN development focusing on platforms based on the well recognized standards preferably supporting open source OSs. The latter requirement provides the testbed designed with a higher level of control over the network parameters.

WIRELESS AP/ROUTERS

Although limited by radio capability due to their small antennas and low-power WiFi cards, many home wireless routers can be successfully exploited as wireless mesh routers. Being characterized by costs between €80–100 (at the time of this writing), including the radio card, these devices provide the cheapest solution for wireless mesh networking. However, due to their limited resources, embedded devices cannot generally run a compiler or a development environment. In such a scenario cross compilation is the only possible way to build programs. Cross compilation is a process by which a compiler generates binary program files that are not meant to be executed on the local system (typically an x86-compatible PC), but rather on a different platform (the embedded device). For example, OpenWRT is a GNU/Linux-based distribution for embedded devices that provides an integrated framework for cross compiling software packages, allowing users to generate customized firmware to be loaded into the wireless router.

PERSONAL COMPUTERS/EMBEDDED PCS

Personal computers based on the Intel x86 architecture provide high flexibility in terms of choices of components that are widely available and may be selected according to specific needs. However, being designed primarily as general-purpose computers, these platforms may be expensive to install and manage, and are not suitable for a 24/7 service. For example, many PCs have moving parts (hard disk and fans), and their outdoor deployment could be challenging due to both the lack of suitable (waterproof) enclosures and power consumption requirements. On the other hand, they are easier to program and offer greater flexibility. A wide range of OSs is available for such platforms, like all Linux/Unix variants, BSD, and the Microsoft OSs.

Being based on the x86 architecture, embedded PCs combine the advantages of wireless routers and PCs. Custom and off-the-shelf hardware (PCI, MiniPCI, etc.) is available, and final systems are characterized by a wide performance range. No cross compilation is required, and standard development tools and OSs can be used. Finally, outdoor deployment is made easier by tailored waterproof enclosure, power over Ethernet support, and the absence of any moving parts.

Soekris Engineering provides a line of x86-based embedded computers. For example, the Soekris net4826-50 board is equipped with a 266 MHz CPU, 128 Mbytes of RAM, one 100 Mb/s Ethernet port, and two MiniPCI sockets. Power can be provided through power over Ethernet. Such a system can be equipped with one or two wireless NICs, providing a cheap yet powerful wireless router application platform. Many OSs are available for these platforms, ranging from open source systems like all Linux/Unix variants and BSD to commercial real-time solutions. Embedded platforms based on non-x86 CPUs (e.g., Routerboard) provide a better price/performance ratio with the drawback of requiring cross compilation.

TIME-SHARED TESTBED FACILITIES

The Orbit project[1] aims at building a laboratory testbed designed to achieve reproducibility of experimentation. Orbit exploits a large two-dimensional grid of 400 nodes, each of which is equipped with two IEEE 802.11 radios. Nodes can be interconnected into user-defined topologies with reproducible wireless channel conditions. Users are allowed to load a custom OS together with any modified system software and applications needed to run their experiments. An extensive library of measurement tools and experimental setups is available.

LAYER 3 SOLUTIONS

In this section we survey the main implementations of layer 3 routing protocols for multihop wireless networks. Implementations are grouped by protocols employed, and have been selected using code maturity, license, and exploitation in real-world testbeds as classification criteria.

AD HOC ON-DEMAND DISTANCE VECTOR (AODV)

The AODV-UIUC project developed a library (Ad Hoc Support Library, ASL) that can be exploited to implement on-demand or reactive ad hoc routing protocols. The library works in user space on GNU/Linux systems, and is provided together with a small loadable kernel module. In order to show the capabilities of the framework, AODV has been implemented using the ASL. A similar design has been used by the AODV-UCSB and AODV-UU implementations. Both exploit the same kernel interaction part, differing only in the logic implementation of the AODV protocol, which is done in user space.

Unlike previous implementations, Kernel-AODV, developed by the U.S. National Institute of Standards and Technology, moves all the routing logic into kernel space; therefore, no user-space daemon is needed. Such an approach improves the performance in terms of packet handling, because no packets are required to traverse from kernel to user space. This implemen-

[1] http://www.orbit-lab.org/

tation is also known to support multiple interfaces and has basic multicast capabilities.

A Java implementation of AODV called JAdhoc is provided by the ComNets Department of the University of Bremen. It uses the Java packet capture library to monitor the interfaces in user space. Current code is known to work in GNU/Linux, Zaurus, and Windows. Security extensions have been added in a recent release.

DYNAMIC SOURCE ROUTING

DSR-UU, developed by Uppsala University, provides a dynamic source routing (DSR) implementation for both GNU/Linux and the ns-2 network simulator. DSR-UU implements a virtual network interface that enables a DSR network to coexist with a regular single-hop ad hoc network at the same time. DSR-UU implements a link cache that supports multiple routing metrics. However, at the time of this writing, only minimum-hop-count routing is supported.

A kernel-level implementation of the DSR protocol is provided by Rice University with the Monarch project. Routing logic is implemented through extensive modifications of the IP stack. At the time of this writing, the project is in a pre-alpha release, and is available for the information and use of other network researchers.

HAZY SIGHTED LINK STATE

Hazy Sighted Link State (HSLS) has been implemented by the Champaign-Urbana Community Wireless Network (CUWiN) for the NetBSD platform. The CUWiN foundation aims at fostering the development of community-owned networks exploiting open source technologies. In order to enhance the portability to other OSs, the routing logic has been implemented as a daemon running in user space. Expected transmission time is used as the routing metric.

OPTIMIZED LINK STATE ROUTING

A cross platform implementation of the OLSR protocol supporting GNU/Linux, MacOS, and the various children of BSD is provided by the University of Oslo as a part of the OLSR daemon (Olsrd) project. Olsrd supports a plug-in interface based on dynamically linked libraries for extending OLSR functionality.

QOLSR is a QoS extension introduced to the OLSR protocol by LRI Laboratory at the University of Paris. Link state information generated by MPRs is exploited in order to provide optimal paths based on applications' QoS requirements. QOLSR does not require any change to the format of IP packets; thus, any existing IP stack can be used, and the protocol only interacts with kernel routing table management.

LAYER 2.5 SOLUTIONS

In this section we review the software platforms implementing a WMN with routing performed at layer 2.5.

ROOFNET

Roofnet is an experimental IEEE 802.11b-based WMN consisting of about 50 nodes located in Cambridge, Massachusetts, installed and operated by the Massachusetts Institute of Technology. The network participants are volunteers who host in their apartments the equipment required to implement a mesh node. Each node is equipped with a single 802.11b wireless card. Roofnet routes packets using SrcRR [10], a protocol inspired by DSR. The original protocol has been modified extensively, mainly for supporting additional link-quality metrics. SrRR uses estimated transmission time as the routing metric [11].

MCL

The mesh connectivity layer (MCL) is a loadable Microsoft Windows driver that implements an interposition layer between the link and network layers of the standard International Organization for Standards open systems interconnection (ISO/OSI) model. MCL routes packets using a modified version of DSR called Link Quality Source Routing (LQSR) [11]. LQSR uses a routing metric called weighted cumulative expected transmission time capable of selecting channel-diverse paths in multiradio environments. An indoor testbed based on MCL software is introduced in [11]. The testbed is exploited to compare the performance of quality-based metrics against minimum-hop routing.

FINAL CONSIDERATIONS AND REMARKS

As a promising technology for ubiquitous wireless network access, WMNs are required to support a wide range of benchmarks and application scenarios. Being a testbed, the ideal playing field to develop and validate innovative solutions, its design should be driven by both research trends and requirements imposed by the application scenarios. This section complements the previously discussed issues by analyzing the most relevant trade-offs involved in designing a testbed. Table 1 summarizes the trade-offs to be considered by both academia and industry in building a WMN testbed. However, we should remember that none of the following issues should be considered closed, but rather open topics for further investigation.

HARDWARE PLATFORM

A WMN designer must consider issues ranging from platform selection and node deployment, to the selection of a suitable software framework for efficient and useful testbed operation. We have already analyzed the most relevant platforms for WMN development. However, additional study is required for more proper choices of both wireless interfaces and antennas.

Wireless Interfaces — Different IEEE 802.11 chipsets are available on the market. While for indoor deployments NICs characterized by a low transmission power should be preferred in order to minimize interference, outdoor deployments require higher transmit power and receiver sensitivity. Common IEEE 802.11 NICs are characterized by an output power of 30 mW (15 dBm), while an access point (AP) can reach 100 mW (20 dBm). By operating in the industrial, scientific, and medical (ISM) bands, WMNs based on IEEE 802.11 technology can exploit frequency bands located around 2.4 and 5 GHz. However,

Roofnet is an experimental IEEE 802.11b-based WMN consisting of about 50 nodes located in Cambridge, Massachusetts, installed and operated by the Massachusetts Institute of Technology. The network participants are volunteers who host in their apartments the equipment required to implement a mesh node.

	Industrial testbed	Academic testbed
Target	Prototype/final product	Proof of concept
Software framework	Focus on code maturity and stability (e.g., OLSRD, AODV-UCSB, AODV-UU). Kernel space implementations (e.g., Kernel AODV) preferred when performance is a major requirement.	Focus on ease of development and software flexibility. User space implementation preferred due to shorter development cycle (e.g. AODV-USCB, AODV-UU, OLSRD, Roofnet).
Hardware platform	Focus on ease of deployment and management management features. Embedded platforms (e.g., Soekris) preferred due to better price/performance ratio.	Focus on ease of development and hardware accessibility. x86 platforms (both embedded and nonembedded) provide a flexible yet cost-effective solution without the need for cross compilation.
License	Permissive license (e.g., BSD, MIT) preferred in order to allow proprietary commercial exploitation.	Both permissive and copyleft (e.g., GPL) licenses are usually suitable.

Table 1. *Design trade-offs for a WMN testbed.*

each range has different characteristics. While the lower frequencies typically exhibit better range, the higher frequencies have shorter range and are subject to greater attenuation from solid objects, but usually present a lower level of background noise.

Wireless NICs that allow control of low-level (physical) parameters should be preferred. As an example, NICs based on the Prism 2.5 chipset allow the control of parameters such as bit rate and carrier sense thresholds, and provide transmission feedback for unicast frames that are not successfully delivered. Atheros-based NICs expose raw 802.11 frames to the driver, allowing control over most of the node's functionality at the application level; for example, it is possible to get per-packet signal and noise readings, and send broadcast frames at arbitrary rates. Such features are exploited, for example, in Roofnet in order to compute the expected transmission time link metric. It is worth noting that most bundled solutions (e.g., the Intel Centrino platform) do not support such advanced features. An extensive comparison of open source drivers and supported chipsets is available at [12].

Antennas — In order to maximize the overall performance of a WMN, a careful selection of antennas and node placement is needed. Omnidirectional antennas, like dipoles, are used when coverage in all directions is required. On the other hand, directional antennas focus more energy in one direction and expend less energy in all other directions. As the gain of a directional antenna increases, the angle of radiation usually decreases, providing a greater communication distance, but with a reduced coverage angle. Directional antennas can increase network throughput by both reducing the exposed terminal problem and enabling sectoral coverage. However, they raise considerable challenges at the MAC layer, such as more hidden nodes. Routing protocols also need to take into account the selection of directional antenna sectors.

Routing Framework

The most relevant implementations of routing protocols for WMNs are summarized in Table 2. The implementation of multihopping requires considerable modifications to the networking stack, often involving kernel space programming. Moving the networking stack into user space libraries offers considerable advantages over kernel space development in terms of both faster development cycle and easier debugging at the expense of performance reduction. Due to such considerations, many academic research testbeds exploit routing protocol implementations running in user space.

A hybrid approach is the Click modular router [13] on which the MIT Roofnet testbed is based. A Click router is built by assembling several packet processing modules, called *elements*, forming a directed graph. Each element is in charge of a specific function such as packet classification, queuing, or interfacing with networking devices. Click comes with an extensive library of elements supporting various types of packet manipulations. Such a library enables easy router configuration by simply choosing the elements used and the connections among them. Finally, a router configuration can easily be extended by writing new elements. The Click modular router is available as both a Linux kernel module and a user space driver, allowing straightforward porting of a user space implementation to kernel space.

Routing Protocols

As stated earlier, proactive protocols maintain a list of all destinations and routes while reactive protocols discover routes on demand when a packet needs to be forwarded. Such behavior makes proactive routing less suitable for WMNs or in general for networks characterized by low churn rates.[2] Moreover, proactive protocols aim at computing routes between any pair of nodes participating in networking,[3] while many reference scenarios for WMNs (e.g., access network) are characterized by a low percentage of intramesh traffic and a high percentage of outgoing traffic. Thus, on-demand route discovery can result in much less traffic than the standard proactive approach, especially when innovative route maintenance schemes are employed. However, the reliance on flooding that characterizes reactive protocols may still lead to considerably high control traffic in mobile networking environments. Moreover, the route discovery process

Name	Protocol	Platform	Implementation	Routing layer
AODV-UCSB	AODV	GNU/Linux	Kernel module w/ user space routing logic	3
AODV-UIUC	AODV	GNU/Linux	User space	3
AODV-UU	AODV	GNU/Linux	Kernel module w/ user space routing logic	3
CUWiN	HSLS	NetBSD	Kernel space	3
DSR-UU	DSR	GNU/Linux	User space	3
JAdhoc	AODV	GNU/Linux, Windows, Zaurus	User space	3 3
Kernel-AODV	AODV	GNU/Linux	Kernel space	3
MCL	LQSR (DSR-like)	Windows	Loadable Windows driver	2.5
Monarch Project	DSR	FreeBSD	Kernel space	3
OLSRD	OLSR	GNU/Linux, Windows, MAC OS X, *BSD	User space	3
QOLSR	OLSR	GNU/Linux	User space	3
Roofnet	SrcRR (DSR-like)	GNU/Linux, *BSD	User space	2.5

Table 2. *Comparison of routing platforms for WMNS.*

may be subject to significant delays due to the large volume of control traffic generated.

PROTOCOL ARCHITECTURE

Routing can be provided either at level three of the ISO/OSI networking stack as a modification of standard IP or by adding an interposition layer between the data link and network layers. In the latter solution (usually referred to as layer 2.5 routing), the multihop backhaul is transparent to the upper networking stack, making the WMN appear as just another Ethernet link. On the other hand, such an approach introduces additional encapsulation and processing overhead as a result of the header and checksum required by the interposition level, respectively. This implies a slight degradation in overall performance in terms of both throughput and latency.

Cross-layer design can also be considered an interesting research direction. Such an approach exploits interactions between layers in order to optimize network efficiency. For example, multiuser diversity is exploited in [14] in order to increase network throughput. In this approach each user provides its "instantaneous" channel condition. This information can be utilized by the scheduling algorithm in order to take advantage of channel variations by giving priority to users with instantaneously better channel quality.

LICENSING

No currently available routing framework may be considered the final answer for building a WMN testbed. An open source license that makes the source code available under terms that allow modification and redistribution may therefore speed up research in this specific field. In this scenario *permissive licenses* (e.g., BSD and MIT) have fewer restrictions than other free software licenses, such as the GPL, which require copies and derivatives of the source code to be made available on the same terms as the original code. While for an academic research testbed this may not be a major problem, for an industrial testbed a routing framework released under a permissive license may be preferable in that it allows a higher degree of freedom in the distribution of the final product [15].

REFERENCES

[1] R. Bruno, M. Conti, and E. Gregori, "Mesh Networks: Commodity Multihop Ad Hoc Networks," *IEEE Commun. Mag.*, vol. 43, no. 3, Mar. 2005, pp. 123–31.

[2] I. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," *Elsevier Comp. Networks*, vol. 47, no. 4, Mar. 2005, pp. 445–87.

[3] IEEE 802.11, "IEEE Standards for Information Technology — Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Network — Specific Requirements — Part 11: Wireless lan Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[4] Chlamtac, M. Conti, and J. Liu, "Mobile Ad Hoc Networking: Imperatives and Challenges," *Elsevier Ad Hoc Networks*, vol. 1, no. 1, Jan. 2003, pp. 13–64.

[5] R. Draves, J. Padhye, and B. Zill, "Comparison of Routing Metrics for Static Multi-Hop Wireless Networks," *Proc. ACM SIGCOMM*, Portland, OR, 2004.

[6] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Proc. ACM SIGCOMM*, London, U.K., 1994.

[7] D. Johnson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPV4," IETF RFC 4728, Feb. 2007, http://www.ietf.org/rfc/rfc4728.txt

[8] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF RFC 3561, July 2003, http://www.ietf.org/rfc/rfc3561.txt

[9] C. Santivanez and R. Ramanathan, "Hazy Sighted Link State (HSLS) Routing: A Scalable Link State Algorithm," BBN, tech. rep. 1301, Aug. 2001.

[10] J. Bicket *et al.*, "Architecture and Evaluation of an Unplanned 802.11b Mesh Network," *Proc. ACM MOBI-*

[2] *Number of nodes that leave the network during a specified time period divided by average total number of nodes over that same time period.*

[3] *Albeit in OLSR only nodes selected as MPRs are responsible for forwarding control traffic.*

COM, Cologne, Germany, 2005.
[11] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," *Proc. ACM MOBICOM*, Philadelphia, PA, 2004.
[12] "Seattle Wireless," http://www.seattlewireless.net/
[13] E. Kohler *et al.*, "The Click Modular Router," *ACM Trans. Comp. Sys.*, vol. 18, no. 3, Aug. 2000, pp. 263–97.
[14] J. Wang *et al.*, "Omar: Utilizing Multiuser Diversity in Wireless Ad Hoc Networks," *IEEE Trans. Mobile Comp.*, vol. 5, no. 12, Dec. 2006, pp. 1764–79.
[15] H. Wang and C. Wang, "Open Source Software Adoption: A Status Report," *IEEE Software*, vol. 18, no. 2, Mar. 2001, pp. 90–95.

BIOGRAPHIES

ROBERTO RIGGIO (roberto.riggio@create-net.org) received a Laurea (M.Sc.) degree (summa cum laude) in telecommunication engineering from the University of Reggio Calabria, Italy, in 2004. He received a Ph.D. degree in telecommunication science from the University of Trento, Italy, in 2008, with a thesis entitled *Cross-Layer Design in Wireless Mesh Networks*. From May 2007 to November 2007 he was a visiting researcher at the Wireless Networks Laboratory (WINET) at the University of Florida. His main research interests lie in the field of wireless mesh networks with a particular focus on cross-layer design and QoS.

NICOLA SCALABRINO (nicola.scalabrino@iit.cnr.it) received Laurea and Master's degrees from the University of Pisa, Italy, both in 2004. During his Ph.D. path at the University of Pisa he cooperated with both the Institute of Informatic and Telematic (IIT) of the Italian National Research Council (CNR) and the CREATE-NET research center in Trento. His research interests focus on wireless networks with emphasis on MAC and routing protocols for IEEE 802.16 and mesh networks. Currently, he is working at a company called Thales.

DANIELE MIORANDI (daniele.miorandi@create-net.org) is head of the Pervasive Area at CREATE-NET. He holds an M.Sc. and a Ph.D. in communications engineering from the University of Padova, Italy. Before joining CREATE-NET in 2005 he held a visiting scholar position at INRIA, Sophia Antipolis, France, and an adjoint researcher position at IEIITCNR, Torino, Italy. He has published more than 60 papers in various international journals and conferences in the networking and computing area. He is currently coordinating the European IP BIONETS (www.bionets.eu), focused on bio-inspired solutions for pervasive computing/communications environment. He serves on the program committees of various leading conferences in the networking field (e.g., IEEE INFOCOM, IEEE ICC, IEEE GLOBECOM), and isone of the co-founders of the ValueTools and Autonomics conferences.

FABRIZIO GRANELLI [SM] (granelli@dit.unitn.it) received a Laurea (M.Sc.) degree in electronicengineering from the University of Genoa, Italy, in 1997, with a thesis on video coding that was awarded with the TELECOM Italy prize, and a Ph.D. in telecommunications from the same university in 2001. Since 2000 he has been an assistant professor of telecommunications with the Department of Information and Communication Technology, University of Trento). In August 2004 he was a visiting professor at the State University of Campinas (Brasil). He is author or co-author of more than 60 papers published in international journals, books, and conferences. He is guest-editor of the *ACM Journal on Mobile Networks and Applications* Special Issue on WLAN Optimization at the MAC and Network Levels and Co-Chair of THE 10th IEEE Workshop on Computer-Aided Modeling, Analysis, and Design of Communication Links and Networks. He was General Vice-Chair of the First International Conference on Wireless Internet,) and General Chair of the 11th IEEE Workshop on Computer-Aided Modeling, Analysis, and Design of Communication Links and Network). He is an Associate Editor of *IEEE Communications Letters* and *IEEE Communications Surveys and Tutorials*. His main research activities are in the field of networking, with particular reference to network performance modeling, MAC, wireless networks, next-generation IP, and video transmission over packet networks. He is Senior Member of IEEE and Vice-Chair of IEEE ComSoc Technical Committee on Communications Systems Integration and Modeling.

YUGUANG FANG [F] (fang@ece.ufl.edu) received a Ph.D. degree in systems engineering from Case Western Reserve University in January 1994 and a Ph.D. degree in electrical engineering from Boston University in May 1997. He was an assistant professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology from July 1998 to May 2000. He then joined the Department of Electrical and Computer Engineering at the University of Florida in May 2000 as an assistant professor, got an early promotion to associate professor with tenure in August 2003 and to a full professor in August 2005. He holds a University of Florida Research Foundation (UFRF) Professorship from 2006 to 2009. He has published over 200 papers in refereed professional journals and conferences. He received the NSF Faculty Early Career Award in 2001 and the Office of Naval Research Young Investigator Award in 2002, and is the recipient of the Best Paper Award at IEEE International Conference on Network Protocols in 2006 and the IEEE TCGN Best Paper Award at the IEEE High-Speed Networks Symposium, IEEE GLOBECOM in 2002. He is also active in professional activities. He is a member of ACM. He has served on several editorial boards of technical journals including *IEEE Transactions on Communications*, *IEEE Transactions on Wireless Communications*, *IEEE Wireless Communications*, and *ACM Wireless Networks*. He was an editor for *IEEE Transactions on Mobile Computing* and currently serves on its Steering Committee. He has been actively participating in professional conference organizations such as serving as the Steering Committee Co-Chair for QShine, the Technical Program Vice-Chair for IEEE INFOCOM 2005, Technical Program Symposium Co-Chair for IEEE GLOBECOM 2004, and a member of the Technical Program Committee for IEEE INFOCOM (1998, 2000, 2003–2009).

ENRICO GREGORI (enrico.gregori@iit.cnr.it) received a Laurea in electronic engineering from the University of Pisa in 1980. In 1981 he joined the Italian National Research Council (CNR) where he is currently a CNR research director. He is currently the director of the CNR institute for Informatics and Telematics (IIT). In 1986 he held a visiting position at IBM Research, Zurich, Switzerland, working on network software engineering and heterogeneous networking. He has contributed to several national and international projects on computer networking. He has authored more than 100 papers in the area of computer networks published in international journals and conference proceedings, and is co-author of the book *Metropolitan Area Networks* (Springer, London 1997). He was the General Chair of the IFIP TC6 conferences Networking 2002 and PWC 2003, and IEEE Pervasive Computing and Communication 2006. He served as guest editor for *Networking* Special Issues on Performance Evaluation and Cluster Computing, and *ACM/Kluwer Wireless Networks Journal*. He is on the editorial board of the *Cluster Computing Journal*, *Computer Networks Journal*, and *Wireless Networks Journal*. His current research interests include ad hoc networks, sensor networks, wireless LANs, QoS in packet switching networks, and evolution of TCP/IP protocols.

IMRICH CHLAMTAC [F] (imrich.chlamtac@create-net.org) is President of CREATE-NET and the Bruno Kessler Professor at the University of Trento, and has held various honorary and chaired professorships in the United States and Europe, including the Distinguished Chair in Telecommunications Professorship at the University of Texas at Dallas, Sackler Professorship at Tel Aviv University, and University Professorship at the Technical University of Budapest. In the past he was with Technion and the University of Massachusetts, Amherst, DEC Research. He has made significant contributions to various networking technologies as scientist, educator, and entrepreneur. He is the recipient of multiple awards and recognitions including as a Fellow of the ACM, a Fulbright Scholar, and the ACM Award for Outstanding Contributions to Research on Mobility and the IEEE Award for Outstanding Technical Contributions to Wireless Personal Communications. He has published close to 400 refereed journal, book, and conference articles, and is listed among ISI's Highly Cited Researchers in Computer Science. He is the co-author of four books, including the first book on local area networks (1980) and the Amazon.com best seller and IEEE Editor's Choice *Wireless and Mobile Network Architectures* (Wiley, 2000). He has widely contributed to the scientific community as founder and chair of ACM Sigmobile, and founder and steering committee chair of some of the leading conferences in networking, including ACM Mobicom, IEEE/SPIE/ACM OptiComm, CreateNet Mobiquitous, CreateNet WiOpt, IEEE/CreateNet Broadnet, IEEE/CreateNet Tridentcom, and IEEE/CreateNet Securecomm conferences. He also serves as founding Editor-in-Chief of *ACM/URSI/Springer Wireless Networks* and *ACM/Springer Journal on Special Topics in Mobile Networks and Applications*.