Effects of physical channel separation on application flows in a multi-radio multi-hop wireless mesh network: An experimental study on BilMesh testbed

Alper Rifat Ulucinar [a], Ibrahim Korpeoglu [a,*], Ezhan Karasan [b]

[a] Department of Computer Engineering, Bilkent University, 06800 Ankara, Turkey
[b] Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Turkey

ABSTRACT

In this paper, we introduce *BilMesh*, an indoor 802.11 b/g mesh networking testbed we established, and we report about our performance experiments conducted on multi-hop topologies with single-radio and multi-radio relay nodes. We investigate and report the effects of using multi-radio, multi-channel relay nodes in the mesh networking infrastructure in terms of network and application layer performance metrics. We also study the effects of physical channel separation on achievable end-to-end goodput perceived by the applications in the multi-radio case by varying the channel separation between the radio interfaces of a multi-radio relay node. We have observed that the difference between TCP and UDP goodput performances together with the delay and jitter performance depends on the hop count. We also observed that assigning overlapping channels with a central frequency separation of 5–15 MHz may render the CSMA mechanism used in 802.11 MAC ineffective and hence reduce the overall network performance. Finally, we provide some suggestions that can be considered while designing related protocols and algorithms to deal with the observed facts.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Wireless mesh networking (WMN) is an active area of research which is believed to be the next step in the evolution of the wireless architecture due to its relatively low cost, flexibility in the hardware and software options, ease of deployment, self-configuration and self-healing properties. Unlike ad hoc networks, infrastructure/backbone and hybrid wireless mesh networks (Akyildiz et al., 2005) employ a wireless mesh backbone composed of mesh routers as an architectural component. And similar to ad hoc networks, this backbone should be self-organizing and self-configuring for scalability, ease of deployment and ease of maintenance. In infrastructure/backbone WMNs (Akyildiz et al., 2005), conventional clients (clients lacking the ability to forward packets on behalf of other nodes) access backhaul services and communicate with each other via the mesh backbone. The mesh backbone, therefore, provides mesh connectivity and routing services in a multi-hop manner for the conventional clients and other mesh clients.

Mesh networking paradigm provides better coverage and better scalability when compared with conventional wireless local area networks due to low deployment and low maintenance costs. Also since the capacity of a communication channel is logarithmically proportional to the signal-to-noise ratio (SNR) by Shannon's channel capacity formulation (Rappaport, 2002), and since increased deployment density implies increased SNR values in general, mesh networking paradigm can provide increased network capacities. Another advantage of the mesh networking paradigm is that it can be applied by modifying layer 3 solely, which makes it possible to apply this paradigm on top of various wireless communications technologies such as Wi-Fi (The Wi-Fi Alliance, 2012), WiMAX (The WiMAX Forum, 2012) or ZigBee (ZigBee Specification, 2008), etc. No new hardware or software below layer 3 is required most of the time, which provides greater flexibility in hardware and software choices, and decreasing costs.

Various non-academic communities have built urban wireless mesh networking infrastructures using low cost commodity hardware and open software. Also many academic groups have reported establishing wireless mesh networking testbeds to research various issues related with the paradigm. Since the mesh networking paradigm is generally applied onto existing MAC and physical layers and is used in conjunction with the widely adopted transport layer protocols, such as TCP, that are not capable of appropriately dealing with packet losses occurring in multi-hop

* Corresponding author. Tel.: +90 3122902599.
E-mail addresses: ulucinar@cs.bilkent.edu.tr (A.R. Ulucinar), korpe@cs.bilkent.edu.tr (I. Korpeoglu), ezhan@ee.bilkent.edu.tr (E. Karasan).

wireless links, researchers are faced with many challenges originating from the MAC and transport layers while designing wireless mesh networks. The multi-hop nature of the wireless mesh backbone and the shared/broadcast nature of the wireless medium also give rise to the well-known hidden and exposed terminal issues. Another important issue arising from the broadcast nature of the wireless medium is that packets of the same multi-hop flow interfere with each other while traversing subsequent links. As we clearly show through experiments conducted in our testbed, this intra-flow interference greatly destabilizes multi-hop flows and reduces achievable goodput.

In 802.11b/g, there are 11 channels where each channel is 22 MHz wide and the central frequencies of consecutive channels are separated by 5 MHz. When the center frequencies of two channels are separated by more than 22 MHz, these channels are considered to be *non-overlapping* (i.e., orthogonal) channels (Mishra et al., 2005a). In 802.11b/g, for two channels to be considered as non-overlapping channels, they should be at least 5 channels away from each other since 5 channels of separation implies that the channel center frequencies are separated by 25 MHz, which is greater than 22 MHz. Otherwise, if two channels are separated by less than 5 channels, they are overlapping. Hence, channels 3 and 8, for example, are non-overlapping whereas channels 1 and 4 are overlapping. There are at most 3 non-overlapping channels (channels 1, 6, and 11) in 802.11b/g that can be used simultaneously.

One common approach when applying mesh networking onto wireless networking technologies which possess multiple overlapping or non-overlapping channels is to make use of multiple channels for adjacent hops. This greatly reduces the hidden and exposed terminal issues though do not completely annihilate them especially when overlapping channels are employed.

In order to be able to use multiple channels with the conventional Wi-Fi radios, one approach is to have the radios hop channels in the course of time (So and Vaidya, 2004; Bahl et al., 2004). However, this approach requires temporal synchronization between the transmitter and receiver radios because the transmitter and the receiver must be operating on the same channel simultaneously to be able to communicate with each other. Another problem with this approach is the latency introduced to the system while switching from one channel to another.

Another approach to employ multiple channels on consecutive hops is to use nodes equipped with multiple radios (Adya et al., 2004). Each radio of the node can be configured to operate on a different channel so that packets arriving in the multi-radio node on one channel may depart the node on a different channel. Although currently available 802.11 b/g hardware does not comprise multiple radios, it is possible to build a *logical* multi-radio node out of two or more single radio modules. This is the approach we pursue and further details of this approach are discussed in Section 3.

In this paper, we introduce our experimental mesh network testbed, called *BilMesh*, and report about our results obtained using this testbed to investigate the effects of physical channel separation on the performance of a wireless mesh network consisting of single-radio and multi-radio nodes. We provide details about how a multi-radio mesh network that supports ad hoc routing can be built and configured using commodity hardware and software, together with details about our node architecture, software configuration and network topology.

Most existing studies in the literature that deal with the channel assignment problem in the context of multi-radio multi-channel WMNs consider only non-overlapping channels. However, as surveyed in Section 2, works of Mishra et al. (2006) and others have demonstrated via simulations that using overlapping channels in addition to the orthogonal (non-overlapping) channels can

actually improve end-to-end application throughput. With its novel, flexible multi-radio node architecture that provides elasticity in antenna placement and that can effectively deal with the Wi-Fi NIC related crosstalk issues, BilMesh is an attempt to further investigate these problems and explore the limitations arising in realistic settings.

Using our testbed, we first investigate and report on the performance improvements achievable by using orthogonal channels for consecutive wireless hops. Then, we quantify by a set of extensive experiments, the goodput gains of using partially overlapping channels (Mishra et al., 2005a, 2006) instead of using only orthogonal channels which is the method commonly followed in the literature. In our study we also investigate how carrier-sense based multiple access mechanism performs if the carrier sensing radio is operating on a different channel than the actively transmitting radio.

The main contributions of our paper are as follows:

- Through BilMesh, we describe in detail how a single and multi-radio wireless mesh network can be built, established and configured with dynamic routing using off-the-shelf 802.11 wireless routers. We report our own experiences with BilMesh, which can be useful for other researchers who want to establish mesh networks.
- We propose a novel, cost-effective multi-radio node architecture for wireless mesh networking testbeds that is flexible in terms of number of radios, antenna placement and RF shielding. Our multi-radio node architecture also does not have the Wi-Fi NIC related crosstalk issues and scales well with the increasing number of Wi-Fi radios since the amount of available computing resources increases with the number of radios.
- Unlike previous multi-radio mesh networking testbeds, BilMesh uses OLSR (RFC3626: Optimized Link State Routing Protocol (OLSR), 2003) as its routing protocol. The OLSR protocol implementation we use in BilMesh is olsrd (olsrd Project Home Page, 2012). In this paper, we discuss the details of the configuration of olsrd in a multi-radio setting.
- We investigate the effects of physical channel separation on the performance of a wireless mesh network with single and multiple radios. Effects on the network layer parameters such as average delay and delay jitter as well as on the transport layer performance (throughput and goodput) are studied.
- We observe that, although UDP is believed to perform better than TCP in terms of achievable goodput, and is thus generally chosen as the transport protocol for multimedia applications which require high bandwidth, this is not always the case in multi-hop wireless networks. As the number of hops a traffic flow traverses increases, TCP begins to achieve higher goodput than UDP. Hence, we propose that if no flow control is implemented at the application level, the transport layer protocol for multimedia applications should be chosen as a function of the number of hops multimedia packets have to traverse.
- We observe and report, by the results of our detailed experiments, that in a multi-hop UDP flow, round trip times for packets increase almost linearly with increasing hop count, whereas jitter increases almost exponentially.
- We observe that, due to CSMA, separating neighboring 802.11b/g radios with one, two or three channels is a worse option than assigning the same channel to them. However, separating neighbor 802.11b/g radios with at least four channels is a better option than assigning the same channel to them. This observation is very valuable for channel assignment algorithms that utilize overlapping channels.

The remainder of the paper is organized as follows: Section 2 discusses some of the available wireless mesh networking platforms and the testbeds. Section 3 introduces BilMesh, an indoor

multi-radio wireless mesh networking testbed we have established. Section 4 covers the descriptions and performance measurement results for multi-hop topologies with single-radio and multi-radio relay nodes (mesh routers). Section 5 concludes the paper.

2. Related work

In this section, we provide a brief summary of some of the available mesh networking platforms and some related work done in multi-channel multi-radio mesh networks. Most software choices in the platforms mentioned here are available in source code from their developers and operate on a variety of hardware. Most common choices run on Linux and Microsoft Windows operating systems.

MIT CSAIL *Roofnet* (Bicket et al., 2005) is an experimental mesh network developed at the MIT CSAIL and deployed over a 4 km² region providing broadband Internet access to its nodes. The average internode throughput is reported to be 627 Kbps for 37 nodes. Roofnet runs in a pseudo-IBSS mode which omits 802.11 beacons and BSSID mechanism. The main functionality provided by Roofnet is broadband Internet access and not peer-to-peer connectivity. Roofnet software is distributed in multiple choices: as a firmware for Netgear WGT634U access points, as a live CD distribution which contains a 45 MB Linux image compiled for the i386 architecture and as an OpenWRT 2.0 package. Roofnet uses Srcr (Bicket et al., 2005) as its routing protocol, and SampleRate (Bicket, 2005) as its rate selection algorithm.

Microsoft Research's Mesh Connectivity Layer (MCL) (Microsoft Mesh Connectivity Layer (MCL) Software, 2012) is part of *Microsoft's Mesh Networking Academic Resource Toolkit* and is also available as a stand-alone download both in binary and source code forms. The toolkit includes MCL source code for Windows XP and Windows CE together with performance measurement tools, configuration tools and related documentation and papers. MCL is a loadable Windows driver which implements a virtual network adapter. MCL sits between the data link layer and the network layer and implements ad hoc routing with link quality measurements. The routing algorithm is Multi-Radio Link Quality Source Routing (MR-LQSR) (Draves et al., 2004) which is a modified version of DSR. MCL can utilize multiple wireless adapters operating at different channels and hence can be used to drive multi-radio architectures. One limitation of the software is that, in the case of multi-radio systems, the radios should be driven using different device drivers. MCL is a good alternative for those wishing to operate their wireless mesh network on the Windows platform.

JHU DSN *SMesh* (Amir et al., 2006) is an 802.11 mesh network deployed at the Distributed System and Networks Lab at Johns Hopkins University. It provides peer-to-peer connectivity, Internet connectivity and fast handoff to mobile VoIP clients. SMesh operates in standard IBSS mode. Mobile clients send and receive data through the mesh infrastructure provided by SMesh and do not rely on each other for forwarding packets. The multi-hop communication infrastructure used by SMesh is provided by Spines (The Spines Overlay Network, 2012; Amir and Danilov, 2003), which is developed by the same group. Spines provide a generic multi-hop messaging infrastructure that allows unicast, multicast and anycast communication with an API similar to the Unix sockets. SMesh binaries are provided upon e-mail request (Amir et al., 2010). It is reported on the SMesh Internet site that it has been tested on x86 architectures and on Linksys WRT54G routers.

Most of the existing work in the wireless mesh networking literature focuses on using orthogonal channels. However, due to the limited number of orthogonal channels in the IEEE 802.11b/g

standards, researchers have also investigated the possibility of using overlapping channels. One of the early works on this subject belongs to Mishra et al. (2005a). In this study, the authors propose the concept of *I-factor* which models the amount of transmit power radiated by a transmitter on channel *j* and received by a receiver on channel *i*. They propose both an analytical model which allows theoretical values to be calculated for the I-factor between two given 802.11b DSSS channels and an empirical model based on throughput measurements. Mishra et al. (2005b) discuss how partially overlapping channels can be leveraged to improve spatial channel reuse in Wireless LANs. Through experiments, they quantify as a function of the physical data rate, the interference range of an Access Point (AP)–Station (STA) pair with respect to another AP–STA pair operating on an overlapping channel. In the context of *single-radio mesh networks*, the authors also investigate the possibility of receiving data from a transmitter operating on an overlapping channel with respect to the receiver's channel. Mishra et al. (2006) demonstrate via simulations that the use of partially overlapping channels in the contexts of Wireless LANs and multi-hop Wireless Mesh Networks can improve end-to-end application throughput.

Robinson et al. (2005) investigate the limitations of the multi-radio testbed platforms and quantify the impacts of specific platform choices only on the application layer throughput. Their wireless mesh testbed is a 2-hop network consisting of a workstation equipped with multiple PCI 802.11b cards. They identify three main causes of performance degradation: Board crosstalk, RF power leakage and inadequate separation between Wi-Fi antennas. They also try to mitigate PCI board crosstalk by shielding the Wi-Fi cards with aluminium foil. Similar observations about board crosstalk have been made in Adya et al. (2004) and in Draves et al. (2004). Zhang et al. (2009) set up a cabled wireless testbed with two PCs. Each of the PCs are equipped with up to 4 802.11a NICs and all NICs are interconnected by couplers and attenuators through a splitter in order to eliminate all wireless medium related factors. Their aim is to study CPU utilization and the effects of board crosstalk between PCI NICs. They report that, for an 802.11a network in a saturated network condition, computing resources is the key limiting factor on the performance rather than the crosstalk between the PCI Wi-Fi cards.

In existing multi-radio mesh networking testbeds, multi-radio nodes are built using multiple PCI or mini-PCI Wi-Fi NICs installed in a single computer system. As the previous studies mentioned above have shown, due to board crosstalk on a single multi-radio system built using commodity hardware, multi-hop network performance is severely degraded. In order to be able to completely eliminate the adverse effects of board crosstalk, we take a different and novel approach in the design of our multi-radio nodes. Two physically separate single-radio APs connected with a high speed wired link constitute our multi-radio node. This approach also scales well with the increasing number of Wi-Fi radios of a multi-radio node because each additional Wi-Fi radio of a BilMesh node comes with its own CPU and main memory. With this multi-radio node architecture, we also have the flexibility to spatially separate the Wi-Fi antennas as needed. Unlike previous testbeds, we can also more effectively address the issues caused by RF power leakage by separating the antennas of the multi-radio node spatially and RF shielding them. In some of the experiments discussed in this paper, we have separated the two antennas of the two-radio nodes and shielded RF radiation, from each other using panels covered with aluminium foils. Another key difference between BilMesh and the previous testbeds mentioned above is that we are using OLSR as the routing protocol in a multi-radio setting.

We have set up networks of up to seven hops using single radio nodes and up to four hops using multi-radio nodes, which reveal previously unobserved facts about the relative performance of TCP

and UDP in a multi-hop multi-radio setting. We also quantify, by a set of extensive experiments, the performance gains obtained when partially overlapping channels are used instead of using only orthogonal (non-overlapping) channels. Additionally, we empirically analyze the effects of different channel combinations and permutations on the performance that a multi-hop network flow experiences in terms of throughput and packet loss rate.

3. BilMesh

In this section, we first describe our testbed; how we have built, established and configured it. We describe in detail our node hardware and software architecture.

In the Engineering Building of Bilkent University, we have built and deployed an 802.11b/g mesh network, called *BilMesh*, consisting of single-radio and two-radio nodes. We use BilMesh as our testbed for wireless mesh networking research. BilMesh is based on Linksys WAP54G and Linksys WRT54GL 802.11b/g access points running the Whiterussian and Kamikaze distributions of the popular OpenWRT firmware. OpenWRT (OpenWRT, Linux for Embedded Devices, 2012) is a Linux distribution for embedded devices like Wi-Fi access points that provides a fully writable file system with package management. Since BilMesh is based on commodity hardware and open source software, we can easily add new nodes (or remove existing ones) when necessary. Furthermore, since our two-radio nodes are built from conventional single-radio nodes, when desired, we can easily turn our multi-radio nodes into single-radio ones.

One Linux PC is configured as the mesh network's Internet gateway and DHCP server. During performance measurements, it also acts as the traffic sink. Another Linux PC on the other end of the network is used as the traffic source. As part of BilMesh, we also have a management and monitoring station also running Linux and a server station running MySQL RDBMS, Apache web server and Apache Geronimo J2EE application server (Apache Geronimo Project Home Page, 2012) on top of Linux.

The Linksys WAP54G (Linksys WAP54G Product Support Page, 2012) is a rather restricted hardware platform for mesh networking

with a 200 MHz Broadcom CPU, 2 MB flash memory and 8 MB RAM. It is based on the BCM4318 SoC (Broadcom BCM4318E Product Home Page, 2012) integrating a CPU and an 802.11b/g interface. The Linksys WRT54GL (Linksys WRT54GL Product Support Page, 2012) is a more powerful platform compared with the WAP54G, offering 4 MB flash memory and 16 MB RAM. WRT54GL is based on the Broadcom BCM5352 SoC router (Broadcom BCM5352EL Product Home Page, 2012) which combines a 200 MHz MIPS32 CPU, an 802.11b/g interface and a configurable five port Fast Ethernet switch.

Figure 1 shows the logical topology of BilMesh together with the architectural roles of its constituent nodes and Fig. 2 shows a logical two-radio node. In Fig. 1, MAP stands for Mesh Access Point, i.e., a wireless router (which we also call as a mesh relay), which can have single or multiple (two) radios. In Fig. 2, the two physically separate APs are connected via Ethernet to constitute a single two-radio node.

Each node in the testbed (including both of the constituent wireless routers of a two-radio node) is connected to an Ethernet
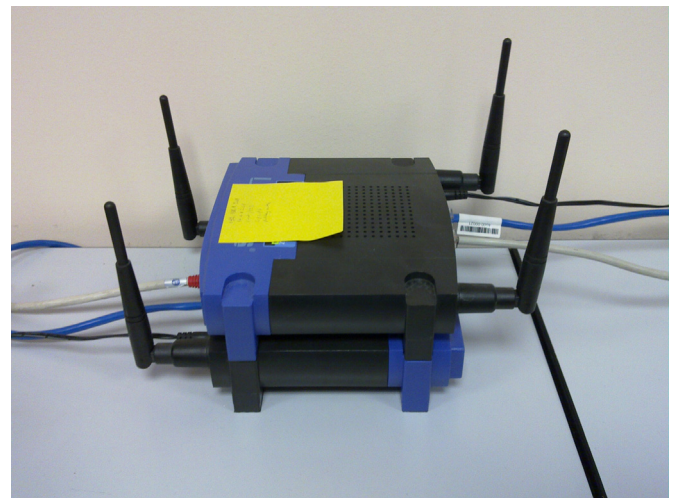

Fig. 2. A BilMesh two-radio node (*Mesh Access Point*) consisting of two distinct APs.
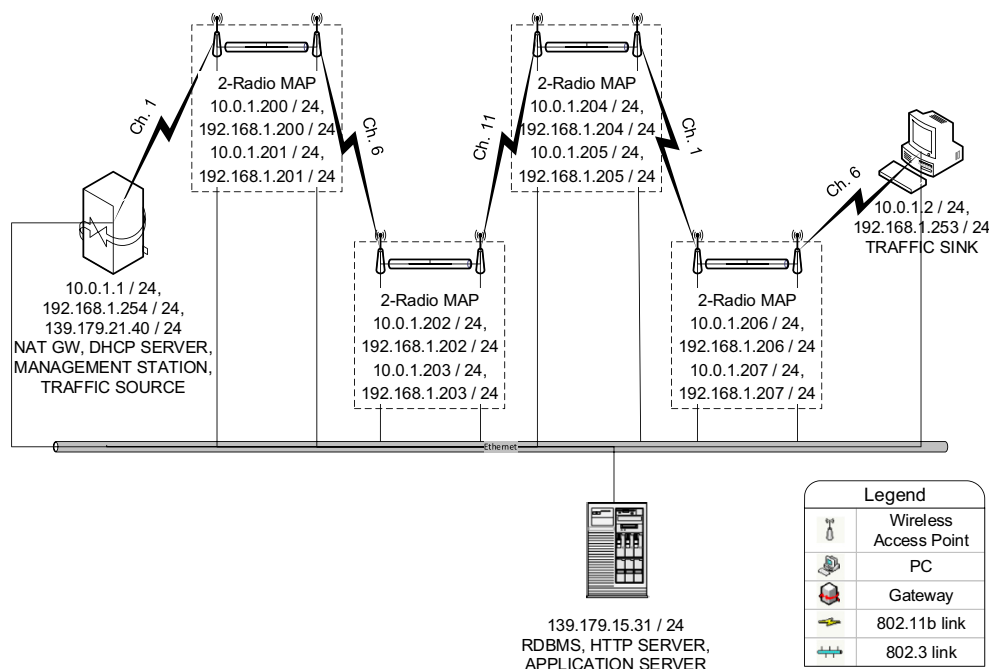

Fig. 1. BilMesh logical topology.

backbone which is used for managing and monitoring the testbed. Using this backbone, channels of the wireless routers can be reliably configured and real-time packet traces can be collected.
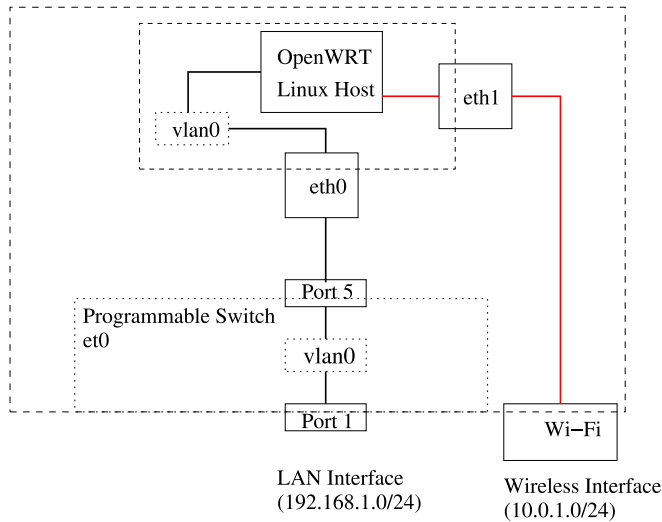

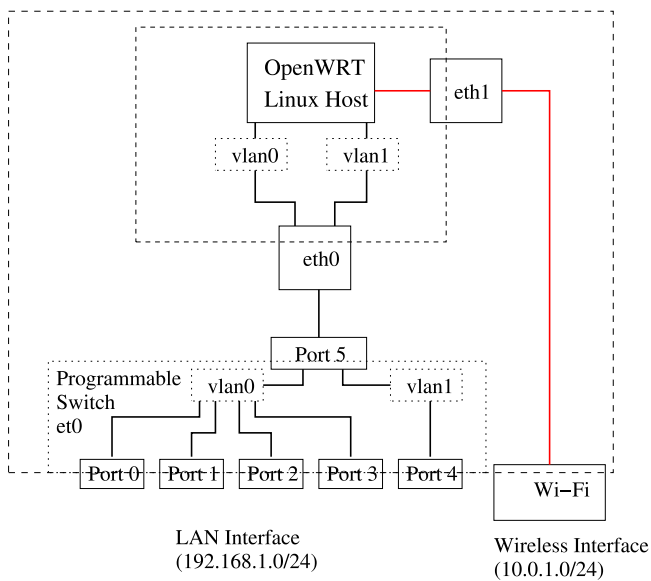**Fig. 3.** OpenWRT based architecture for a WAP54G in BilMesh.

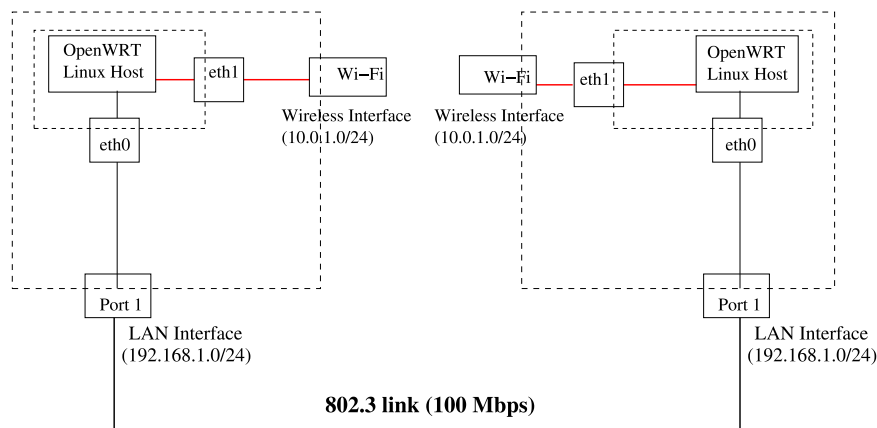**Fig. 4.** OpenWRT based architecture for a WRT54GL in BilMesh.

Also all experiments carried on the testbed can be remotely controlled to prevent any unwanted fluctuations in wireless link conditions caused by moving bodies.

As the routing protocol, OLSR (RFC3626: Optimized Link State Routing Protocol (OLSR), 2003) is run on BilMesh. Each constituent router of a two-radio node runs an instance of the OLSR daemon (olsrd Project Home Page, 2012) which sets up the routing table. Further details on routing software configuration and operation are given in the next section.

3.1. Node configuration

All nodes in BilMesh operate in the 802.11 IBSS mode. In its default configuration on Whiterussian distribution, the 802.3 and the 802.11 interfaces of a WAP54G are bridged together. We break this bridge and configure the wired and wireless interfaces separately, so that packets arriving at the wireless interface can be routed through the wired interface. For maintenance purposes, the nodes can be accessed via their Ethernet interfaces. RTS/CTS is disabled in the network.

Figure 3 shows the architecture based on the OpenWRT firmware for a WAP54G node. The wireless interface eth1 is removed from the bridge br0 (which contains eth1 in the default configuration) and is configured to be in the 10.0.1.0/24 network. The VLAN vlan0 consists of ports 1 and 5 of the programmable switch et0 and is not tagged. For ease of configuration, VLAN vlan0 is configured via the default bridge br0 (from which the wireless interface has been removed) in the 192.168.1.0/24 network. Interface configuration is performed via init scripts. Also in these init scripts, the routing information for a specific node can be supplied if static routing is desired, in which case, OLSR daemon should also be disabled. Similarly, Fig. 4 shows the architecture for a WRT54GL node. Again, the wireless interface has been removed from bridge br0.

The routes followed by packets when ad hoc dynamic routing is used can change quite often even in infrastructure meshes like BilMesh since the conditions of the wireless medium change rapidly. If a set of experiments to be carried on BilMesh requires the network packets to follow the same routes, then we disable olsrd and use static routing, i.e., the routing table is stored at boot time in the network layer of a node's TCP/IP stack. Also for the static routes to be forced, ICMP Redirect message (RFC792: Internet Control Message Protocol (ICMP), 1981) generation and processing are disabled at the Linux TCP/IP stack. This can be achieved by setting the keys net/ipv4/conf/all/send_redirects and net/ipv4/conf/all/accept_redirects to 0 in the sysctl preload/configuration file.

Fig. 5. A dual radio node comprising two WAP54g hardware.

3.2. Building two-radio nodes

We have built the two-radio nodes out of two WAP54G or two WRT54GL or one WAP54G and one WRT54GL devices. In order to achieve a two-radio node, we connected two WAP54G/WRT54GL

The single instance of the daemon is instructed to work on both of the wired `eth0.0` and the wireless `wl0` interfaces.

Listing 1. Part of OLSR daemon configuration on a multi-radio node.

```
Interface "wl0"
{
        Ip4Broadcast 255.255.255.255
        HelloInterval 2.0
        HelloValidityTime 40.0
        TcInterval 5.0
        TcValidityTime 100.0
        MidInterval 18.0
        MidValidityTime 324.0
        HnaInterval 18.0
        HnaValidityTime 108.0
}

Interface "eth0.0"
{
        Ip4Broadcast 255.255.255.255
        HelloInterval 2.0
        HelloValidityTime 40.0
        TcInterval 5.0
        TcValidityTime 100.0
        MidInterval 18.0
        MidValidityTime 324.0
        HnaInterval 18.0
        HnaValidityTime 108.0
}
```

371

devices via their Ethernet interfaces and supplied the necessary routing information to route packets received via the radio interface of a box to the radio interface of the other box in the init scripts. Since packets are routed from one radio to the second radio of this two-radio node via the interconnected 802.3 interfaces, these two radios may be operated on different channels. This effectively gives us a two-radio, two-channel node in which the two radio interfaces can be configured independent of each other. Since the 802.3 link in our setup has a dedicated bandwidth of 100 Mbps, the bottleneck links are the wireless links. Figure 5 shows the architecture of a dual radio node which consists of two WAP54G boxes.

Since our two-radio nodes are built using two separate physical boxes, a single instance of the OLSR daemon cannot access both radios of the logical two-radio node. However, in order to be able to route packets between these radios *which may be operating on different frequencies*, we need to have these radios discover each other with OLSR HELLO messages (RFC3626: Optimized Link State Routing Protocol (OLSR), 2003). Furthermore, the Multipoint Relay (MPR) Selection Sets of each radio must be disseminated to every other radio in the network regardless of the operating channels. To solve these problems we adopted the following approach: on each constituent router of each logical two-radio node, a single instance of the OLSR software is run. OLSR daemon is configured to operate on both the wired (802.3) and the wireless (802.11) interface of the constituent router. In this way, we were able to disseminate network-wide routing information among radios operating on different frequencies, which would not be possible if the daemon operated only on the radio interfaces of the routers. Listing 1 contains the related section of the `olsrd` configuration file.

Using our WAP54G/WRT54GL single and two-radio nodes together with our desktop PCs (and laptops) as endpoints, we performed extensive experiments on our testbed with different network configurations and scenarios. In the following section, we describe in detail our experimental setups and report the results of our experiments.

4. Experimental evaluation

We have conducted experiments on single-hop and multi-hop (up to 5 hops) topologies carrying both UDP and TCP traffic. The TCP and the UDP traffic is generated using the Iperf tool (Iperf: The TCP/UDP Bandwidth Measurement Tool, 2012) on Linux. For two-hop and three-hop topologies, we have built two-radio, two-channel relay nodes and repeated our experiments to compare the results with their single-radio counterparts. To obtain stable routes for controlled experiments, we used static routing as explained in Section 3.1. For each topology, we have also measured RTTs for packets of sizes of 64, 350, 700 and 1470 bytes and we report the jitter values of UDP traffic for each setup. In the discussions that follow, the definition of jitter follows the definition of *Interarrival Jitter* in RFC 3550 (RFC3550: A Transport Protocol for Real-Time Applications (RTP), 2003). Also for the two-hop multi-radio relay node setup, we investigate the effects of channel separation between the interfaces of the two-radio relay node on network performance.

For multi-hop topologies, each intermediate router forwards a packet it receives to the next router in the chain towards the

packet's destination. As an example, the routing tables of the nodes of the five-hop topology are given in Table 1.

The following two subsections discuss the single-radio relay node and two-radio relay node setups separately.

4.1. Experiments with single-radio relay nodes

In order to find the TCP and UDP goodputs achievable on an 802.11b link in our setup, we performed a set of experiments on a one-hop topology. Since the sender and the receiver are only one-hop away from each other, there is no interference from consecutive hops of the stream and signals belonging to other colocated wireless networks constitute the primary source of interference. Figure 6 shows the setup for the goodput measurement experiments on a single-hop topology. PC 1 and PC 2 are connected together via an 802.11b link on channel 1 at 11 Mbps in IBSS mode. PC 1 generates UDP traffic with a demand of 11 Mbps targeted at PC 2. 15 goodput measurements were made with this setup and the average goodput was found to be 6896 Kbps. Another set of 15 goodput measurements were performed where PC 1 generates TCP traffic targeted at PC 2, and the average goodput was found to be 5438 Kbps. The average jitter for UDP packets was 0.45 ms for this setup.

Figure 7 shows the setup for the goodput measurement experiments involving single radio nodes in a two-hop topology. The box labeled as Wireless Router (called WR from now on) is a WRT54GL. PC 1, PC 2 and WR form an 802.11 IBSS (Independent Basic Service Set) on channel 1. All links are 802.11b links at 11 Mbps. Nodes are placed purposefully close to one another (each separated by 1 m) to increase the *intra-flow interference*, which refers to the interference on a link of a flow caused by the subsequent links used by the same flow. PC 1 generates UDP traffic with a demand of 11 Mbps targeted at PC 2 but instead of
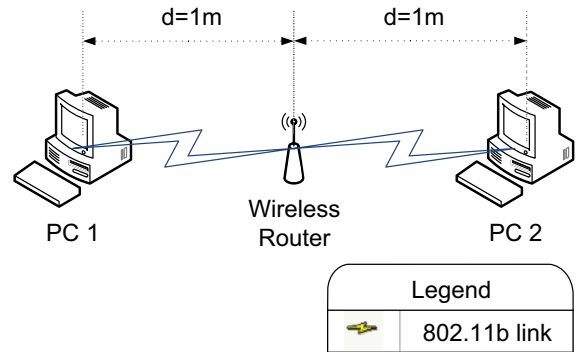
Table 1
Routing table configurations of the nodes of the five-hop topology (entries for the 802.3 interfaces not shown).

Node Id	Local address	Next hop to PC 2	Next hop to PC 1
PC 1	10.0.1.1	10.0.1.200	–
WR 1	10.0.1.200	192.168.1.201	10.0.1.1 (PC 1)
WR 2	10.0.1.201	10.0.1.202	192.168.1.200
WR 3	10.0.1.202	192.168.1.203	10.0.1.201
WR 4	10.0.1.203	10.0.1.204	192.168.1.202
WR 5	10.0.1.204	192.168.1.205	10.0.1.203
WR 6	10.0.1.205	10.0.1.206	192.168.1.204
WR 7	10.0.1.206	192.168.1.207	10.0.1.205
WR 8	10.0.1.207	10.0.1.2 (PC 2)	192.168.1.206
PC 2	10.0.1.2	–	10.0.1.207

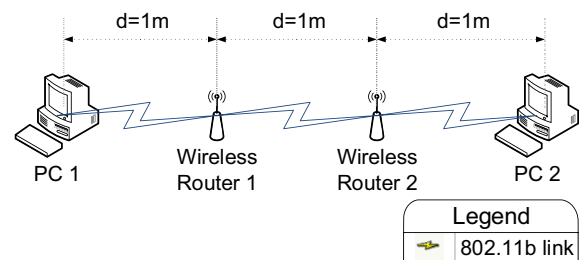**Fig. 7.** Experimental setup for a two-hop network with single radio nodes.

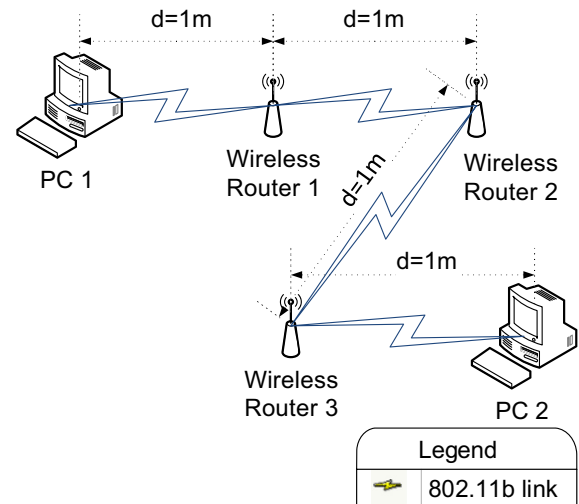**Fig. 8.** Experimental setup for a three-hop network with single radio nodes.

Fig. 9. Experimental setup for a four-hop network with single radio nodes.

directly sending this traffic to PC 2, PC 1 asks the WR to relay this traffic to its destination. 15 measurements were made with this setup and the average goodput was found to be 3377 Kbps. The average goodput for TCP traffic from PC 1 to PC 2 was 2722 Kbps out of 15 measurements and the jitter was found to be 1.67 ms.

For the three-hop topology shown in Fig. 8, where PC 1 is the traffic source and PC 2 is the destination and packets are relayed over WR 1 and WR 2, the average goodput for UDP traffic was found to be 2275 Kbps out of 15 measurements and the average TCP goodput was found to be 1831 Kbps out of 15 measurements. The average jitter for this topology turned out to be 3.55 ms.

For the four-hop topology shown in Fig. 9, where PC 1 is the traffic source and PC 2 is the destination and packets are relayed over WR 1, WR 2 and WR 3, the average goodput for UDP traffic was found to be 1570 Kbps out of 15 measurements and the average TCP goodput was found to be 1258 Kbps out of

Fig. 6. Experimental setup for a single-hop network.

15 measurements. The average jitter for this topology turned out to be 6.83 ms.

For the five-hop topology shown in Fig. 10, where PC 1 is the traffic source and PC 2 is the destination and packets are relayed over WR 1, WR 2, WR 3 and WR 4, the average goodput for UDP traffic was found to be 893 Kbps out of 15 measurements and the average TCP goodput was found to be 900 Kbps out of 15 measurements. The average jitter for this topology turned out to be 13.01 ms.

Table 2 summarizes the averages of the results of the measurements obtained on these 1–5 hop topologies with single radio relay nodes. As it can be seen from the experiment results, as the hop count increases both the achievable TCP and UDP goodputs decrease. For smaller hop counts, due to TCP's acknowledgements and congestion and flow control mechanisms, one can achieve larger goodput by using UDP at the transport layer. The interesting fact observed here is that as the hop count reaches 5-hops, TCP can achieve larger goodput than UDP. The UDP source, lacking any transport layer feedback from the subsequent hops, sends as much traffic as CSMA/CA MAC allows. The amount of traffic a UDP source can offer is a function of solely the capacity of the first link in a multi-hop flow (assuming the application always has packets to send as soon as a packet is successfully delivered to the transport layer). However, a TCP source receives transport layer feedback from the traffic destination and throttles itself by means of flow and congestion control mechanisms. As more and more hops are added to a flow, since the links (of the hops) are spatially separated, the capacity of the first link does not change and the UDP source generates packets in a greedy way, that have no chance to reach their destination. However, a TCP sender expects acknowledgement from its receiver and this prevents it from generating unnecessarily large number of packets that would be dropped in intermediate links with high probability. After 4 hops, as TCP's self-throttling mechanisms mitigate congestion among the hops of the flow (*intra-flow congestion*), TCP begins to perform better than UDP in terms of goodput.
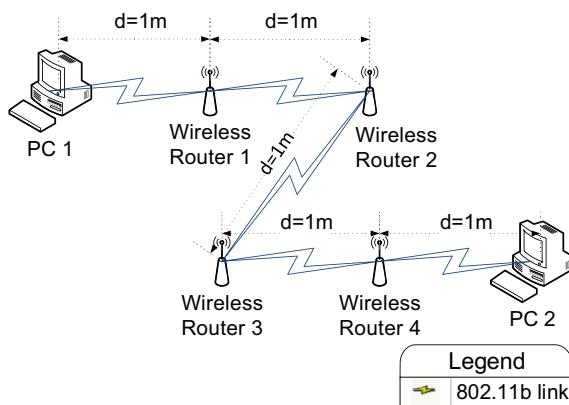
For each of these topologies, RTTs were measured with ping packets of 56 (default payload size in iputils ping), 342, 692, 1462 byte payloads. 1462 bytes of ICMP payload corresponds to 1470 bytes of ICMP message together with the 8 byte ICMP header, which in turn is the datagram size used in UDP goodput measurements. Figure 11 summarizes the RTT measurements. For all packet sizes, RTT increases almost linearly with respect to increasing hop count and the rate of increase of RTT with respect to hop count increases as the packets grow in size.

Figure 12 plots the UDP jitter values on these topologies for datagrams of 1470 bytes. As seen in Fig. 12, the jitter for UDP packets increases almost exponentially with respect to increasing hop count.

In order to observe the effects of offered traffic volume on the UDP flow goodput, packet drop rates and the jitter, we performed other sets of experiments with the single-radio nodes for 1, 2, 3, 4, 5, 6 and 7 hop cases. In these sets of experiments, the physical link rates are kept constant at 11 Mbps but the offered UDP traffic volume at the source is varied (whereas in the previous sets of experiments, it was kept constant also at 11 Mbps). Also, aluminium foiled panels were used between the hops to decrease the interference range of the transmitter radios. Figures 13–15 show the averages of the results for these experiments. The offered traffic load is varied from 1 Mbps to 11 Mbps in increments of 2 Mbps and for each offered load in each topology, a total of 15 experiments were performed. As expected, in Fig. 13 as the offered load increases, the application level goodput first increases and
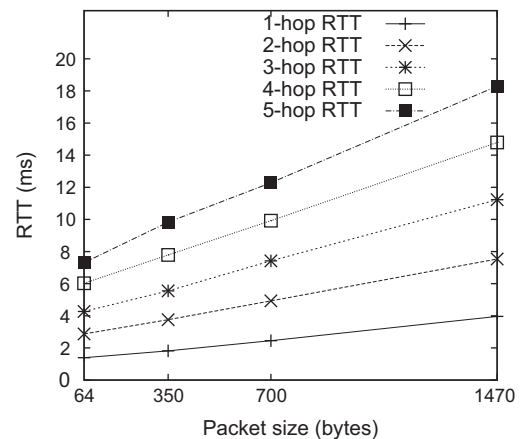
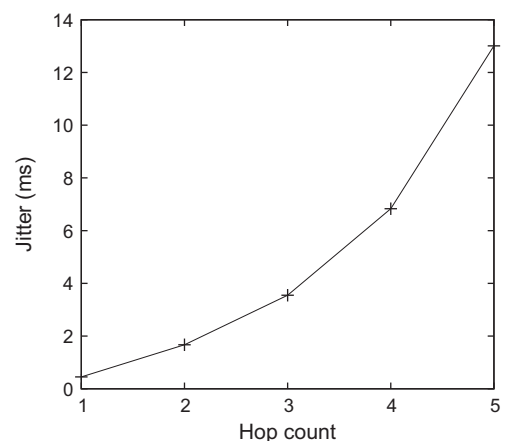**Fig. 11.** RTT measurements for varying sizes of ICMP payloads.

Fig. 10. Experimental setup for a five-hop network with single radio nodes.

Table 2
Averages of the measurements for experiments with single radio relay nodes. RTT averages reported here are for 1470 bytes packets.

Hop count	UDP goodput (Kbps)	TCP goodput (Kbps)	RTT (ms)	UDP jitter (ms)
1	6896	5438	3.95	0.45
2	3377	2722	7.54	1.67
3	2275	1831	11.23	3.55
4	1570	1258	14.79	6.83
5	893	900	18.3	13.01

Fig. 12. Jitter values for UDP packets on 1–5 hop topologies using single-radio relay nodes.
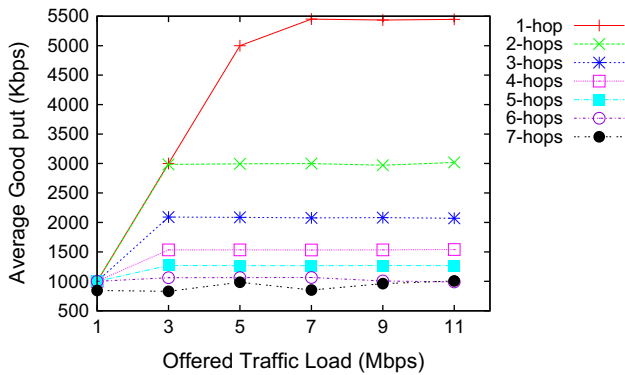
Fig. 13. Average goodput values for various offered traffic volume for single-radio topologies with 1–7 hops.
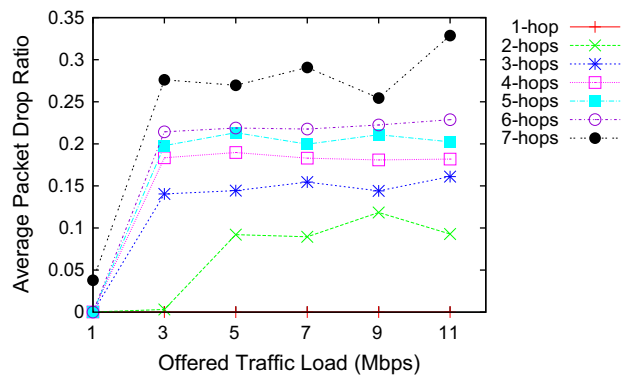

Fig. 14. Average packet drop ratios for various offered traffic volume for single-radio topologies with 1–7 hops.
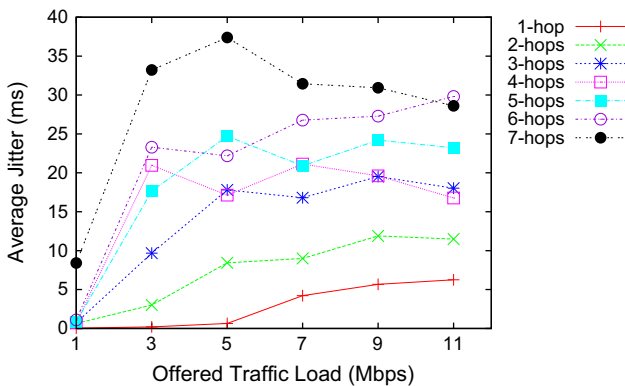

Fig. 15. Average jitter values for various offered traffic volume for single-radio topologies with 1–7 hops.

then saturates at about 1007 Kbps for 7-hops, and at 5447 Kbps for 1-hop. The general trend in Fig. 13 is that at a given offered load, goodput decreases as the hop count increases since the contention among the links increases. The effects of the aluminium foiled panels are also clearly visible for 2, 3 and 4 hops. If these panels were not used to mitigate inter-hop interference, then one would expect an average goodput of 2750, 1833.3 and 1375 Kbps at maximum for 2, 3 and 4 hops respectively, since the single hop average goodput is below 5500 Kbps. To some extent, these panels have been able to mitigate inter-hop interference. Also as expected, Fig. 14 shows that at a given offered load, packet drop ratio increases with hop count which is due to increasing intra-flow interference. The same trend also exists for the jitter measurements, however, with some irregularities as it is displayed

in Fig. 15. Jitter values rise as high as 37 ms for the 7-hop topology. The increase in jitter values when going from an offered load of 1 Mbps to 3 Mbps becomes more significant as the hop count increases.

4.2. Experiments with two-radio relay nodes

In order to test the viability of using overlapping channels in a multi-radio node setting, we conducted a set of goodput measurement experiments, using Iperf: The TCP/UDP Bandwidth Measurement Tool (2012) as our traffic generator. Our aim in performing these experiments is to quantify by measurement, the amount of application level performance degradation due to using overlapping channels on a multi-radio relay node. Figure 16 shows the setup for goodput measurement experiments involving a two-radio relay node in a two (wireless) hop topology. To obtain a two-radio relay node, the two WRT54GL single-radio wireless routers labeled as WR 1 and WR 2 are interconnected via Ethernet and as explained in Section 3.1, each wireless router relays a packet it receives from its radio interface to the other wireless router through the Ethernet connection which then transmits the packet via its radio interface. In this setup, the channel on which the radio of WR 1 operates is changed from 1 through 11, whereas the channel on which WR 2 operates is kept constant at 6. PC 1 connects to WR 1 in 802.11 IBSS mode and hence the channel on which the radio of PC 1 operates is also varied accordingly. PC 2 connects to WR 2 and hence the radio of PC 2 is operated on channel 6. PC 1 generates UDP traffic (with a demand of 11 Mbps) targeted at PC 2 and the system of wireless routers consisting of WR 1 and WR 2 acts as a *two-radio relay node* to carry this traffic. All 802.11 radios in this setup operate in the IBSS mode at 11 Mbps. For each channel configuration, 6 goodput measurements are performed with a total of 66 measurements. Each measurement lasts 10 s. Figure 18 depicts the normalized average goodput values obtained through these measurements. The averages are normalized with respect to the average goodput obtained when WR 1 is at the same channel as WR 2 (channel 6). The maximum average UDP goodput is obtained when
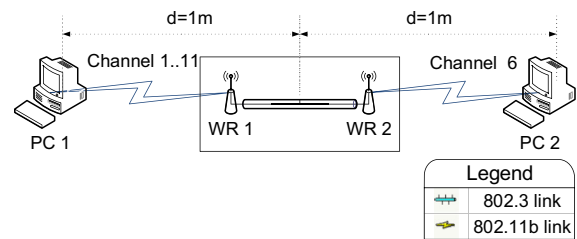

Fig. 16. Experimental setup involving a two-radio relay node in a two-hop topology.
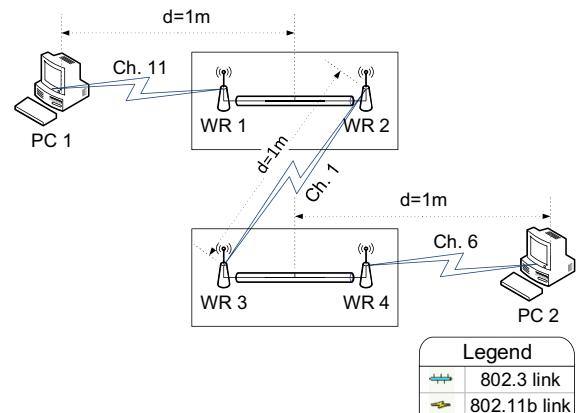

Fig. 17. Three (wireless) hop setup involving two two-radio relay nodes.
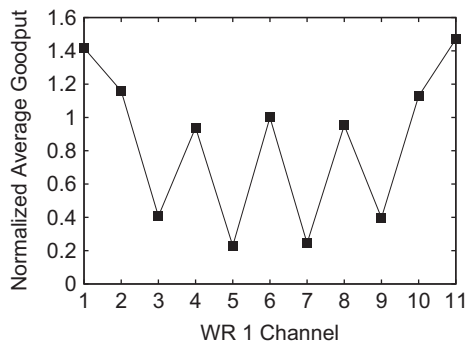
Fig. 18. Normalized average goodput measurements in the setup involving a two-radio relay node.

WR 1 is at channel 11 and WR 2 is at channel 6. As it can be seen from Fig. 18, when the separation between the two channels of the relay node is 4, we have a goodput gain of at least 113% compared with a channel separation of less than 4.

Also another interesting observation is that when WR 1 is set to operate on channels 3, 4, 5, 6, 7, 8 or 9, we have a local maximum at channel 6 which is the channel occupied by WR 2. This can be attributed to CSMA. Carrier sensing is more effective when both radios are on the same channel compared with the cases when two radios operate on channels that are 1–3 channels away, reducing packet losses and increasing the goodput. This observation might be valuable for channel assignment tasks involving multi-radio nodes. If a channel assignment algorithm being designed allows overlapping channels to be assigned to neighboring radios, it is better to assign the same frequency to these radios instead of assigning channels that are one, two or three channels away.

Figure 17 shows the setup consisting of two two-radio relay nodes in a three (wireless) hop topology. WR 1 and WR 2 are interconnected via Ethernet and form a two-radio relay node as explained in Section 3.1. Similarly, WR 3 and WR 4 are interconnected via Ethernet to form another two-radio relay node. In this topology, PC 1 communicates with PC 2 via 5 hops two of which are 802.3 links (hence there are 3 wireless hops). In this setup, WR 1 operates on channel 11, WR 2 operates on channel 1, WR 3 operates on channel 1 (so that there exists a 802.11b link between WR 2 and WR 3), WR 4 operates on channel 6. PC 1 and PC 2 operate on channels 11 and 6, respectively. Hence, all of the three wireless links are operated on orthogonal channels. As explained previously, nodes are placed close to one another (separated by 1 m) to increase intra-flow interference. The average UDP goodput is measured to be 5401 Kbps and the average TCP goodput is found to be 3055 Kbps for this setup. Jitter is observed to be 2.05 ms. When these results are compared with their counterparts of the single radio three wireless hops case in Table 2, it can be observed that there is a goodput improvement of about 237% for UDP traffic and about 167% for TCP traffic for 3-hop topologies. The jitter values for the multi-channel case decrease less sharply, by about 42%, compared with the single channel three hops case. We may conclude from these results that the goodput gains for UDP and TCP traffic in the multi-channel case are higher than the jitter gains. This can be attributed to the additional queues introduced with the 802.3 links in the multi-channel setting.

For both multi-radio setups, RTTs were measured for ping packets of 64, 350, 700, 1470 bytes (including ICMP headers). Figure 19 summarizes the averages for these RTT measurements for the two multi-radio relay node topologies discussed. When compared with the performances of their single-radio counterparts depicted in Fig. 11, it can be seen that RTT values are higher in the multi-radio case. The difference comes from the additional 802.3 links and the additional store-and-forward delays
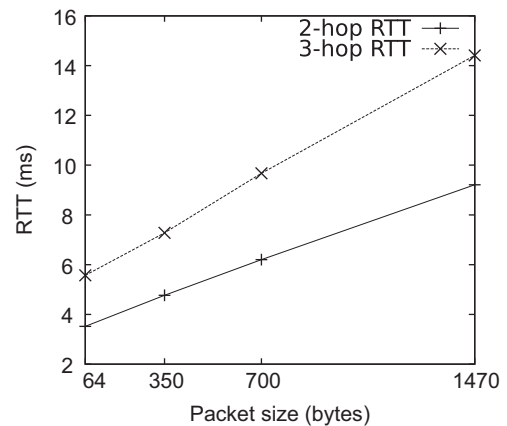

Fig. 19. RTT measurements for multi-radio relay setups with varying sizes of ICMP payloads.
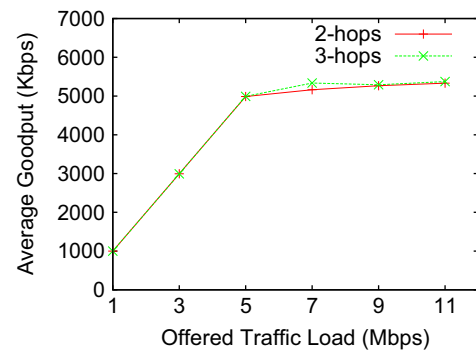

Fig. 20. Average goodput values as offered traffic volume changes for 2-hop and 3-hop two-radio topologies.

introduced in our multi-radio setup. For a two-hop topology, in the single-radio setting, there is only one intermediate (wireless) router relaying the traffic, whereas in the multi-radio setting of our setup, there are two such routers. We should note here, that the ping packets used for measuring the average RTT for a given ping packet size are sent with a separation of 1 s and they are not flooded. Since these ping packets are not flooded (i.e. are not sent back to back), they do not experience intra-flow interference. If these ping packets were flooded, then the multi-radio setup would have an advantage over the single-radio setup because of the mitigation of the intra-flow interference in the multi-radio multi-channel setting.

In order to observe the effects of offered traffic volume on the application goodput in the multi-radio case, we repeated the previously described set of experiments with multi-radio relay nodes for two and three hop topologies. Again, the physical link rates are kept constant at 11 Mbps but the offered UDP traffic volume at the source is varied from 1 Mbps to 11 Mbps in increments of 2 Mbps. For the two (wireless) hop topology, we employed channels 1 and 6 and for the three (wireless) hop topology, we employed channels 1, 6 and 11. The setup for the 2-hop topology is similar to that of Fig. 16 with the only difference that the link between PC 1 and WR 1 operates on channel 1. The setup for the 3-hop topology is identical to Fig. 17. Figures 20 and 21 show the averages of the results for these experiments. As it can be seen from Fig. 20, when non-overlapping channels are used, the maximum achievable goodput does not differ significantly between two and three hops. When the average goodputs reported in Fig. 20 are compared with their single-radio node counterparts of Fig. 13, we can see that, due to more parallel transmissions in the multi-radio case, maximum average goodputs

for 2 and 3 hops have increased up to 170% and 237% respectively. We also observe that, in the multi-radio node setup, the difference in the maximum goodputs of 2 and 3 hop flows have decreased significantly because the flows do not experience intra-flow inter-ference even for 3 hops. Figure 21 shows that the jitter values are below 1 ms for offered loads of 1, 3 and 5 Mbps. But as the offered load rises above 5 Mbps, jitter increases rapidly.

We also experimented with a 4-hop multi-radio topology to assess if goodput can be improved by using distinct overlapping channels instead of repeating non-overlapping channels on differ-ent links. Since the number of non-overlapping channels is 3 in IEEE 802.11 b/g, we need at least 4 wireless hops to investigate this fundamental question. We used the two topologies depicted in Fig. 22 for these sets of experiments. In Fig. 22(a), only non-overlapping channels 1, 6, 11 are used but since we have 4 wireless hops, one of these channels has to be repeated (e.g., channel 1 is used on links 2 and 4). However, in Fig. 22(b), overlapping channels 1, 4, 7, 11 are used and no channel is repeated on subsequent links. Our aim is to investigate whether allowing overlapping channels to be used improves performance over repeating channels on subsequent links.

We performed several goodput measurement experiments with the 4-hop topology depicted in Fig. 22 using various permutations of channels. Due to space constraints, we only report the results of selected scenarios. In each of these scenarios, the scenario name consists of the channel numbers of the links from the traffic sink to the traffic source in order. The channel configuration is also listed in
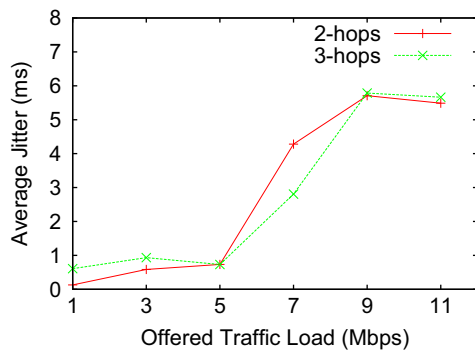

Fig. 21. Average jitter values as offered traffic volume changes for 2-hop and 3-hop two-radio topologies.

the same order. For instance, if the scenario name (or channel configuration) is 4, 7, 1, 11, then in the related experiment from the traffic source to the traffic destination, the 1st link operates on channel 11, the 2nd link operates on channel 1, the 3rd link operates on channel 7 and the last link operates on channel 4. For these sets of experiments, all links are 802.11b links operating at 11 Mbps and the transmit powers for all of the transmitters are fixed at 17 dBm (about 50 mW). Also in order to observe link-level packet losses, using pcap library, we collected packet traces on wireless routers labeled as WR 1, WR 3, WR 5 and on the traffic destination (PC 2) using a pcap snap length of 70 bytes. A snap length of 70 bytes is enough to capture the application level packet header used by Iperf. Iperf assigns a *packet identifier* incremented by 1 to each packet it generates and puts it in the application layer packet header. Also at the end of the session, Iperf traffic source reports to the traffic sink the total number of packets generated during the session. Doing a post analysis on the packet traces after the experiment is completed, we were able to identify the packet loss rate on each of the four wireless links individually. Each experiment is repeated 10 times.

In Figs. 23(a), 24(a) and 25(a), we report the averages of the goodput measurements with respect to increasing offered traffic load. And in Figs. 23(b), 24(b) and 25(b), we report the averages of the percentages of lost packets on individual links. In Figs. 23(b), 24(b) and 25(b), "Packet Drop Ratio" represents the average overall packet loss ratio, which is the ratio of the total number of lost packets (on Link 1, 2, 3 or 4) to the total number of packets sent by the traffic source.

In our experiments, we have observed that if we stick to only non-overlapping 802.11b channels, we obtain almost the same goodput for different permutations of channels *as long as channel repetition on neighboring links* (links incident on a common node) is not allowed. But if we employ overlapping channels, then the permutation of channels chosen has a more profound impact on goodput.

Figures 23 and 24 reveal an interesting fact. Although the channel subset used for the 4 hops of the network is exactly the same in these scenarios, maximum achievable goodput differs by 31% (2325.3 Kbps vs. 3053.8 Kbps). Separating the first two links by three channels performs considerably worse than separating the last two links by three channels. The reason for this phenom-enon is that in our setup, links operating three channels away from each other severely interfere with each other because of being spatially close. If the three channel separation is used
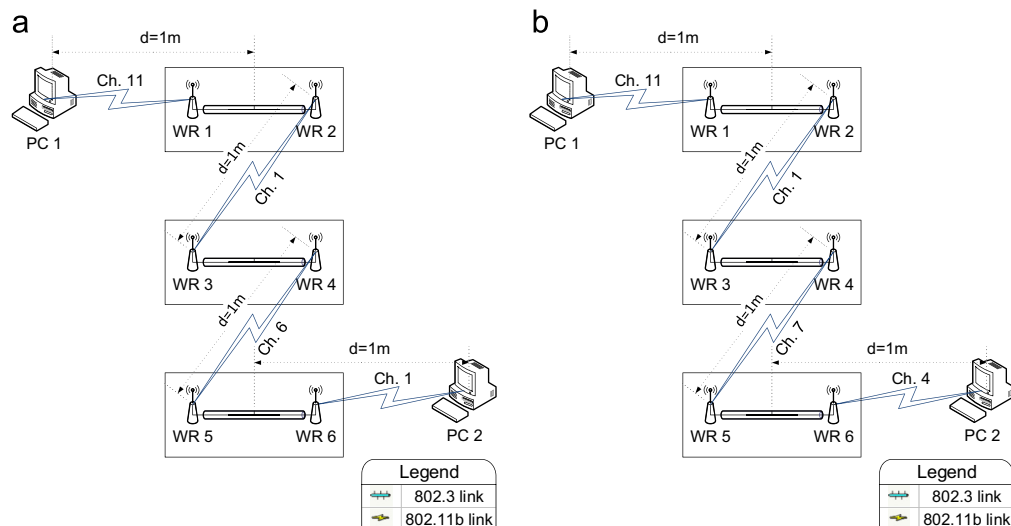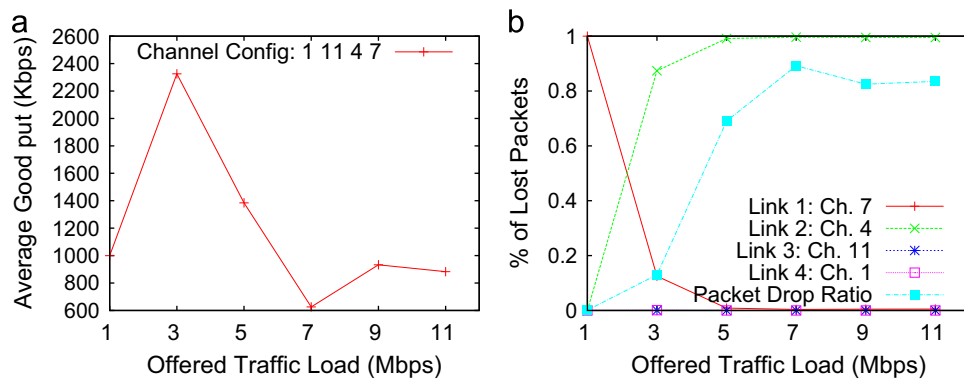

Fig. 22. Motivational Example: Is using channels 1, 6, 11 solely and repeating channels when needed better, or is allowing overlapping channels better? (a) Using only non-overlapping channels. (b) Allowing overlapping channels.

Fig. 23. 4-hop Scenario 1, 11, 4, 7. (a) Average goodput vs. offered load for 4-hop scenario 1, 11, 4, 7. (b) Link loss percentages for 4-hop scenario 1, 11, 4, 7.
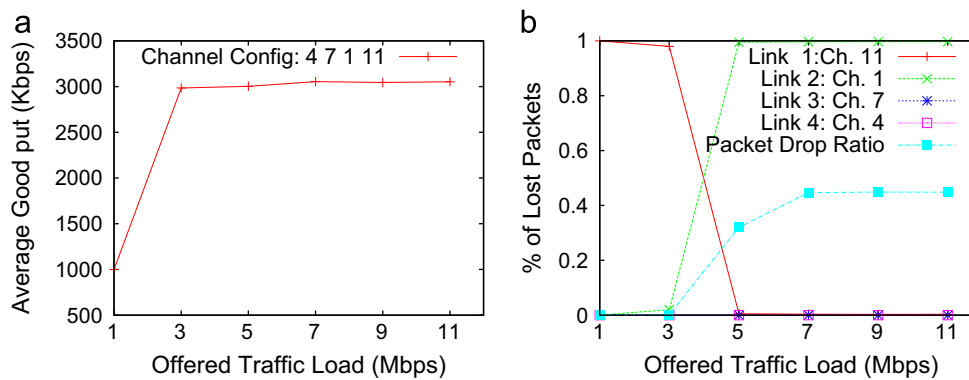

Fig. 24. 4-hop Scenario 4, 7, 1, 11. (a) Average goodput vs. offered load for 4-hop scenario 4, 7, 1, 11. (b) Link loss percentages for 4-hop scenario 4, 7, 1, 11.
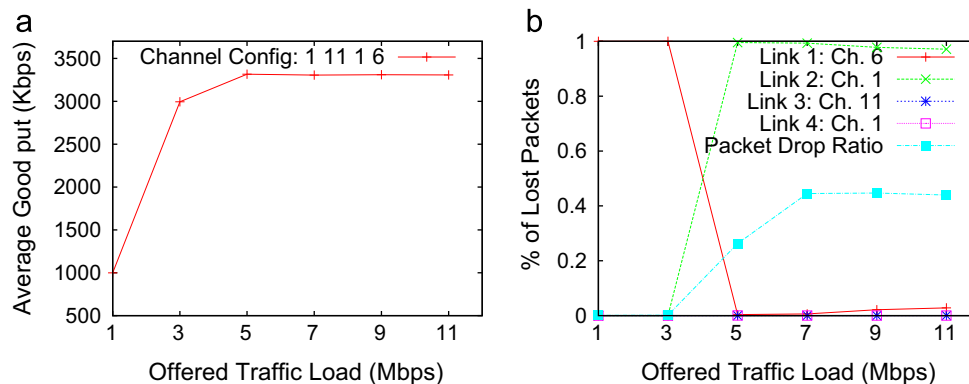

Fig. 25. 4-hop Scenario 1, 11, 1, 6. (a) Average goodput vs. offered load for 4-hop scenario 1, 11, 1, 6. (b) Link loss percentages for 4-hop scenario 1, 11, 1, 6.

between the first and the second links on which more packets are carried compared with the third and the fourth links, more packets are lost due to collisions. However, if the three channel separation is used between the last two links that carry less traffic due to the thinning effect, and interference between the more loaded first two links is kept relatively low (i.e., by employing non-overlapping channels), relatively less number of packets are lost.

Another observation that follows from these figures is that having *higher* inter-link intra-flow interference at the *beginning* of a flow (e.g., interference between the first two links of the *same* flow) makes the flow *less stable* with respect to increasing load. If we consider Fig. 23, increasing the number of packets on the first link by increasing the offered load, decreases goodput up to 73% (when the offered load is 7 Mbps). However, if the first two links operate on non-overlapping channels and do not interfere with each other as in the scenario given in Fig. 24, the flow is much

more stable with respect to increasing offered load. As it can be deduced from Figs. 24 and 25, on a linear topology, repeating a channel on non-consequent (two or more hops away) links is a better choice than using overlapping channels in consequent links when goodput is concerned. Again in Fig. 25, we observe that the obtained goodput as offered load increases is more stable when compared with that of Fig. 23. In Fig. 25(b), we observe that for offered loads of 1 and 3 Mbps, nearly 100% of the packet losses occur on the first link but the overall packet loss ratio is almost 0%. But as the offered load is increased beyond 3 Mbps, exceeding the capacity of the path, the overall packet loss ratio jumps to over 40% and almost all of the losses occur on the second link. The situation is similar for the scenario 4, 7, 1, 11 as it can be observed in Fig. 24(b). On both of these scenarios, the first and second links are operated on non-overlapping channels. If overlapping channels are used on the first and second links as in Fig. 23(b), at an offered load of

3 Mbps, the overall packet loss ratio rises to 13% and like the previously mentioned scenarios, most of the packet losses (about 87%) are on the second link. However, an offered load of 3 Mbps is above the path capacity for this scenario.

When we consider link-level packet losses, it can be seen from Figs. 23(b), 24(b) and 25(b) that the second link is the most vulnerable link in our experiments under heavy traffic load. This is because packet losses occur at the transmit queues of the nodes rather than at the links themselves. Since the first link's transmit queue is at the PC and is larger than the subsequent router transmit queues and since the number of packets making it to the 3rd and the 4th links' transmit queues is substantially smaller, more packets are dropped at the 2nd link's transmit queue. This is in accordance with our observation stated above, that it is more important to protect the head of a flow from interference (intra-flow or external) than to protect the tail. When assigning channels to radios or when *making routing decisions*, this fact must be taken into account.

5. Conclusions

In this paper, we report on an indoor 802.11 mesh networking testbed at Bilkent University and provide experimental evaluation results on multi-hop topologies for TCP and UDP traffic. The achievable goodput quickly drops as the hop count increases when operating on a single channel but employing multi-radio, multi-channel nodes as the intermediary relaying nodes can provide up to 192% improvement in UDP goodput and up to 176% improvement in TCP goodput in a two-hop topology. The UDP goodput improvement reaches 237% when the flow is three hops long. TCP is more sensitive to the increased packet loss rate and increased RTTs as the hop count increases. With the multi-radio architecture used in our experimental setups, RTTs in multi-hop topologies where packets are relayed by multi-radio nodes are longer and RTTs grow faster as hop count increases compared with the case where packets are relayed by single-radio nodes. This is due to the additional processing performed by the access points constituting the multi-radio relay node, when routing the packets from the receiving wireless interface of one access point to the transmitting wireless interface of the *other* access point via the 802.3 link.

Despite this adverse effect in our multi-radio architecture, the additional channel capacity utilized by making use of multiple physical radio interfaces results in improvements of achievable goodput up to 167%. Another interesting result reported in this study is that when utilizing overlapping 802.11b channels for multi-radio nodes, one has to take special care to separate the channels assigned to the radio interfaces appropriately. This is because separating the 802.11b radio interfaces with 1, 2 or 3 channels (corresponding to central frequency separations of 5, 10 and 15 MHz respectively) may severely degrade the achievable performance compared to the case in which the same channel is assigned to the interfaces. On the other hand, a separation of 4 channels, which implies the assignment of overlapping channels in the context of 802.11b, achieves goodput improvements of up to 189% for UDP traffic in a two-hop topology when compared to the single-radio case. As mentioned above, using non-overlapping channels achieves even higher goodput improvements. According to the results reported, operating the radio interfaces of the multi-radio relay node on the same channel effectively turns it into a single-radio relay node from the perspective of network performance for UDP traffic without providing any advantage for using multi-radio nodes.

Acknowledgments

References

Adya A, Bahl P, Padhye J, Wolman A, Zhou L. A multi-radio unification protocol for IEEE 802.11 wireless networks. In: Proceedings of the first international conference on broadband networks, broadNets'04; 2004.

Akyildiz IF, Wang X, Wang W. Wireless mesh networks: a survey. Computer Networks 2005;47(4):445–87.

Amir Y, Danilov C. Reliable communication in overlay networks. In: Proceedings of the IEEE international conference on dependable systems and networks, DSN'03; 2003. p. 511–20.

Amir Y, Danilov C, Hilsdale M, Musaloiu-Elefteri R, Rivera N. Fast handoff for seamless wireless mesh networks. In: Proceedings of the fourth international conference on mobile systems, applications and services, MobiSys'06. ACM; 2006. p. 83–95.

Amir Y, Danilov C, Musaloiu-Elefteri R, Rivera N. The SMesh wireless mesh network. Science and Technology 2010;28(3):1–4.

Apache Geronimo Project Home Page. Last Access: Dec. 2012; URL ⟨http://geronimo.apache.org/⟩.

Bahl P, Chandra R, Dunagan J. SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 Ad-Hoc wireless networks. In: Proceedings of the 10th annual international conference on mobile computing and networking, MobiCom'04; 2004.

Bicket JC. Bit-rate selection in wireless networks. Wireless Networks 2005:1–50.

Bicket J, Aguayo D, Biswas S, Morris R. Architecture and evaluation of an unplanned 802.11b mesh network. In: Proceedings of the 11th annual international conference on mobile computing and networking, MobiCom'05 2005. p. 31.

Broadcom BCM4318E Product Home Page. Last Access: Dec. 2012; URL ⟨http://www.broadcom.com/products/Wireless-LAN/Wi-Fi-Phone-Solutions/BCM4318E⟩.

Broadcom BCM5352EL Product Home Page. Last Access: Dec. 2012; URL ⟨http://www.broadcom.com/products/Wireless-LAN/802.11-Wireless-LAN-Solutions/BCM5352EL⟩.

Draves R, Padhye J, Zill B. Routing in multi-radio, multi-hop wireless mesh networks. In: Proceedings of the 10th annual international conference on mobile computing and networking, MobiCom'04; vol. 04. ACM; 2004. p. 114.

Iperf: The TCP/UDP Bandwidth Measurement Tool. Last Access: Dec. 2012; URL ⟨http://iperf.sourceforge.net⟩.

Linksys WAP54G Product Support Page. Last Access: Dec. 2012; URL ⟨http://homesupport.cisco.com/en-eu/support/routers/WAP54G/⟩.

Linksys WRT54GL Product Support Page. Last Access: Dec. 2012; URL ⟨http://homesupport.cisco.com/en-eu/support/routers/WRT54GL⟩.

Microsoft Mesh Connectivity Layer (MCL) Software. Last Access: Dec. 2012; URL ⟨http://research.microsoft.com/mesh⟩.

Mishra A, Rozner E, Banerjee S, Arbaugh W. Using partially overlapped channels in wireless meshes. Wimesh, Santa Clara 2005a;26.

Mishra A, Rozner E, Banerjee S, Arbaugh W. Exploiting partially overlapping channels in wireless networks: turning a peril into an advantage. In: Proceedings of the fifth ACM SIGCOMM conference on internet measurement. USENIX Association; 2005b. p. 29.

Mishra A, Shrivastava V, Banerjee S, Arbaugh W. Partially overlapped channels not considered harmful. ACM SIGMETRICS Performance Evaluation Review 2006;34 (1):63.

olsrd Project Home Page. Last Access: Dec. 2012; URL ⟨http://www.olsr.org/⟩.

OpenWRT, Linux for Embedded Devices. Last Access: Dec. 2012; URL ⟨http://openwrt.org⟩.

Rappaport TS. Wireless communications principle and practice. second ed. Prentice Hall PTR; 2002.

RFC3550: A Transport Protocol for Real-Time Applications (RTP). 2003. URL ⟨http://www.ietf.org/rfc/rfc3550.txt⟩.

RFC3626: Optimized Link State Routing Protocol (OLSR). 2003. URL ⟨http://www.ietf.org/rfc/rfc3626.txt⟩.

RFC792: Internet Control Message Protocol (ICMP). 1981. URL ⟨http://www.ietf.org/rfc/rfc792.txt⟩.

Robinson J, Papagiannaki K, Diot C, Guo X, Krishnamurthy L. Experimenting with a multi-radio mesh networking testbed. In: WiNMee workshop; 2005. p. 1–6.

So J, Vaidya N. Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. In: Proceedings of the fifth ACM international symposium on mobile ad hoc networking and computing, MobiHoc'04; 2004. p. 222–33.

The Spines Overlay Network. Last Access: Dec. 2012; URL ⟨http://www.spines.org⟩.

The Wi-Fi Alliance. Last Access: Dec. 2012; URL ⟨http://www.wi-fi.org⟩.

The WiMAX Forum. Last Access: Dec. 2012; URL ⟨http://www.wimaxforum.org⟩.

Zhang C, Kowalik K, Davis M. An experimental study of the impact of using multi-radio in WLAN mesh networks. In: Proceedings of the fifth international conference on wireless communications, networking and mobile computing, WiCom'09. IEEE; 2009. p. 1–4.

ZigBee Specification. ZigBee Document Ver 053474r06 2008; URL ⟨http://www.zigbee.org⟩.