Electronic Communications of the EASST
Volume 17 (2009)

Workshops der Wissenschaftlichen Konferenz Kommunikation in Verteilten Systemen 2009 (WowKiVS 2009)

Virtual WLAN: Going beyond Virtual Access Points

Hakan Coskun, Ina Schieferdecker and Yahya Al-Hazmi

12 pages

Guest Editors: M. Wagner, D. Hogrefe, K. Geihs, K. David
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Virtual WLAN: Going beyond Virtual Access Points

Hakan Coskun[1], **Ina Schieferdecker**[1] **and Yahya Al-Hazmi**[1]

[1] [coskun, ina, alhazmi]@cs.tu-berlin.de
Institut für Telekommunikationssysteme
Technische Universität Berlin, Germany

Abstract: Wireless nodes equipped with multiple radio interfaces open up new fields of application. Ranging from multi-channel usage in a cell in order to increase the bandwidth to the creation of meshed multi-hop topologies. Using multiple wireless cards demands a large physical space, more energy consumption and as a consequence decreasing in the battery lifetime. Virtualization of the wireless network interface, which means to use a single wireless network interface to connect to more than one network simultaneously, seems to be a promising approach, since it allows us to realize the mentioned scenarios only with one radio interface. In this paper, we want to shed light on the state of the art and want to introduce new approaches to push this field beyond the current status.

Keywords: 802.11, Virtualization, Power Saving Mode, WLAN, Simulations

1 Introduction

The 802.11 family is the most successful wireless technology for local area networks, today. Everything started in 1997, when the original version was released by the IEEE. In the following year, a set of additional 802.11 standards joined that family and contributed significantly to the success of this technology. Right from the start 802.11 supported different services, each of them enabling different scenarios [iee07]. The basic service set (BSS) and the extended service set (ESS) are usually called infrastructure mode and provided by an access point (AP) which has the control of the channel usage. On the contrary, the independent basic service set (IBSS) does not require any configured infrastructure for operation, no access points. Operating systems call this feature ad-hoc mode, this refers to the ability to setup networks spontaneously. In this mode wireless stations (STA) communicate directly with each other in a peer-to-peer manner. Beside these different services, 802.11 networks support multiple channels for data transmission. Wireless network interfaces are tuned into one channel and configured to operate as STA or as AP. One possible approach to overcome this limitation is to equip the mobile node with multiple radio interfaces which are tuned into multiple channels at the same time [RC05] and operate in different modes. Besides this costly solution, using virtualization techniques on 802.11 MAC layer can create the illusion of having multiple physical wifi NICs inside the node, allowing the same functionality as in multi-radio scenarios. Usually, 802.11 wireless LAN cards and drivers are designed to operate exclusively with one configuration. This configuration is composed of a set of different parameters, like network name, channel, mode, etc. If the signal of the available connection is getting too weak, the management of the card disrupts the connection and tries to find a different configuration to setup a new network connection. Against this background, virtu-

alizing the wireless interface means to break this limitation and to allow different configurations on top of the same physical NIC. The challenge is to switch between the available configurations and to track the states of each network connection and keep them consistent between the virtual interface and the corresponding attachment point.

1.1 Usage Scenarios

The virtualization of 802.11 interfaces enables several capabilities, some of these are listed:

- **Multi-channel Awareness:** A host can use a virtual interface to scan periodically for the characteristics of all available channels (background scanning), without losing its association [1] [SAS08].
- **Simultaneous Connectivity:** A wireless device can be connected to several networks simultaneously; a host can connect to an access point, while staying in an ad-hoc network [RC05].
- **Experiment Multiplexing:** Experiments can co-exist on the same testbed by time-division multiplexing of the wireless facility [SCMB07].
- **Network Coverage Extension/ Relaying:** Mobile stations which are part of a wireless cell, might extend its range by creating a second virtual interface in AP mode. This allows nodes that were originally outside the range of the Access Point to connect to the network.
- **Hybrid mode WLAN:** A station that is connected to an ad hoc network might become a relay to the Internet, when an AP with internet connectivity is in its range [MLV03].
- **Increased Capacity:** By using virtual interfaces on orthogonal channels can reduce interference and thereby increase the capacity of a wireless. Furthermore, the ability to become a relay on-demand can increase the capacity as well [Cet06].
- **Soft Handover** (make-before-break): 802.11 WLAN supports only hard handover; if wireless stations want to connect to a new network, it's only possible after losing the connection to the current AP. This causes long delays (e.g., VoIP services are delay-sensitive). Using a combination of capability 1 and 2 can support soft handover.

According to the way of operation, virtualization deals with different aspects. In this paper we want to shed light on the technical requirements and methods used to achieve virtualization of 802.11 interfaces in infrastructure based networks.

2 Background and Related Work

Modern operating systems provide the ability to create virtual network devices that are not bound to any physical device per se, thus enabling to cope with packets sent over them entirely in

[1] Atheros-based chipsets already provide background scanning [mad]

software, which is totally different from real network devices. Instead of receiving packets from a physical media, virtual devices receive them from userland application attached to them. This enables the programmer to process the packets sent to this device as desired. From the point of view of the operating system this pseudo-devices provide a usual network-interface giving the OS the possibility to cope with them as such. It is common to use so-called TUN interfaces, which are able to handle layer 3 packets, in order to send IP over any connection, like TCP, UDP [tin], HTTP [htu] or even DNS, what is called as tunneling. The TUN interface is managing the internal connection initialization, setup, management of state etc. Virtual network devices that operate with layer 2 frames such as Ethernet are called TAP interfaces. Applications, like virtual machines, that generate Ethernet frames use the tap interface to write/read Ethernet frames to/from the pseudo-interface.

The creation of virtual devices on top of wired connections, is mainly dealing with the transmission of IP packets or Ethernet frames coming from the OS to the ”wire”, without taking the physical medium and the card into consideration. Physically the link is established between two points of attachment and any impairment of the communication is normally not the case. On the contrary, in wireless networks the physical link is subject to change, not only characteristics of the used channel may vary over time; the possibility to have a changing set of points of attachment is most challenging task here. The virtualization of a wifi interface therefore requires a very tight integration with the MAC sub-layers.

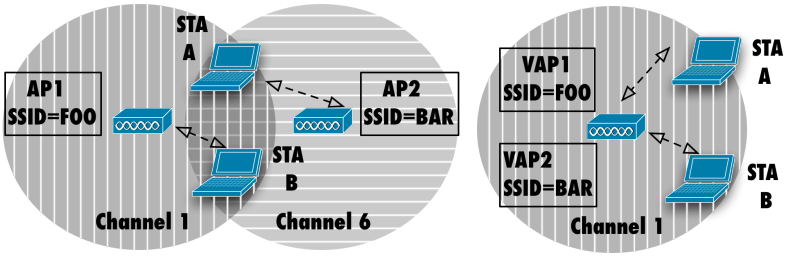
2.1 Soft-MAC

In the beginning, the 802.11 medium access control was entirely implemented by the firmware on the wireless device, which is executed on its embedded CPU. This approach is called ’hard-MAC’ and is essential for embedded applications because host processors do not have the processing power to handle MAC layer functionality. New wifi cards, like Atheros-based chipsets, do not provide a full MAC functionality on the card. Big portions of medium access control is moved into the card driver that allows extensive control over the MAC layer while still allowing the flexible underlying physical layer to define the waveform [mad]. Linux and Windows provide a generic WLAN API that makes use of such a soft-MAC based approach [lin].

2.2 Virtual Access Points

The IEEE 802.11 specification [iee07] defines a management frame called beacon, which is used by an access point to announce its availability and capabilities (Figure 0(a), Figure 1). In the standard there is no restriction how often these beacons are advertised. A single physical AP is allowed to send out beacons as often as it likes. The SSID, specifying the network name, is part of this beacon and is allowed to differ between two beacons.

An AP can provide STAs with the illusion of multiple physical APs within the same enclosure by using different SSIDs inside a beacon or in subsequent beacons (Figure 0(b)). Each of these APs is a virtual access point (VAP). A virtual access point is a logical entity that exists within a physical Access Point and is bound to a virtual network interface. When a single physical AP supports multiple VAPs, each Virtual AP appears to stations (STAs) to be an independent physical AP, even though only a single physical AP is present. It is necessary for Virtual APs to



(a) Multiple physical access points (b) Multiple virtual access points

Figure 1: Virtualizing Access Point

emulate the operation of physical APs at the MAC layer. Emulating the operation of a physical AP at the radio frequency layer is typically not possible for current implementations within a Virtual AP, unless multiple radios are available.

2.3 Virtual STA

The research community already presented some approaches how one WLAN network interface can be virtualized to simultaneously connect to multiple wireless networks even on different channels (Figure 2).

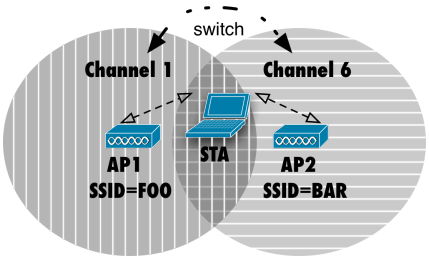


Figure 2: Multiple virtual STAs (with channel switu

VirtualWiFi [Res05], previously known as MultiNet [CB04], is a virtualization architecture for WLAN cards from Microsoft. It abstracts a single WLAN card to appear as multiple virtual WLAN cards to the user. A software based approach has been used by VirtualWiFi for virtualization of wireless interfaces. An intermediate driver, they refer to as the MultiNet Protocol Driver, is placed as between IP and the MAC. This driver exposes to the IP stack that all the virtual interfaces are always active even though at the driver level only one is active. It is also responsible for switching the wireless cards across the different networks, and buffering packets for networks that are currently inactive.

Net-X [PV06] is a testbed being developed by the Wireless Networking Group at UIUC to provide support for exploiting interface capabilities available in a wireless network in the form of multiple channels, interfaces, data rates and transmission power etc. The Net-X implementation assumes that two interfaces are available at each node; one of them is switched to a fixed channel,

while the second interface is switched to any of the remaining channels. The Net-X approach is to develop operating system support for utilizing the interface capabilities. Therefore, they have designed an architectural support in Linux that provides higher layers fine-grained control over all available channels and interfaces, and other interface capabilities. Net-X implementation architecture includes two important protocols: on-demand multichannel routing protocol and buffering protocol. The required support has been added as a new module that operates between IP and the device driver.

2.4 Power Saving Mode (PSM)

The IEEE 802.11 MAC layer specification [iee07] defines two different power management modes a STA can operate in one of them: Active mode (AM) and Power Saving mode (PSM). In the Active mode, a STA is fully powered and is able to exchange frames at any time. While in Power Saving mode, a wireless STA is allowed to be in one of two different power states, either in an awake state or in a doze state.

The access point (AP) in an infrastructure network monitors the mode of each STA. A STA changing its power management mode must first inform its AP about this fact using the Power Management bits within the Frame Control field of the frame used as a power saving request. This frame will be sent following the basic medium access procedure. A STA shall not enter the PSM before it receives the reply from the AP. During the association procedure, the STA informs the AP about its listen interval, which used to indicate the period of time for which a STA in PSM may choose to sleep. In case of a successful association, the AP assigns an Association Identifier (AID) to the PSM STA in the association response frame. An AP transmits regularly management frames called Beacons, spaced by a fixed Beacon interval (typically 100 ms), which used to advertise itself and define the timing for the entire BSS. All data frames destined to the PSM STA shall be buffered in the AP and only transmitted at designated times. The listen interval is used by the AP in determining the lifetime of the frames buffers for the PSM STA. The length of listen interval is measured in Beacon intervals. Thus, a PSM STA can sleep and miss a specific number of Beacons, without losing any data traffic or disconnecting from the network. For example, consider that a STA sets its listen Interval to 4 Beacon intervals; it would wake to listen to every four Beacon and check whether it has traffic buffered for it in the AP. In according to the IEEE 802.11 standard, the AID is used by AP to indicate the existence of a unicast traffic buffered for the PSM STA, and represents a bit in the Traffic Indication Map (TIM), which is sent within each Beacon frame. Every Beacon frame carries a TIM information element indicating the buffer status of all the PSM STAs. Two different TIM types are distinguished: TIM and DTIM. Immediately after a DTIM, the AP shall transmit the buffered broadcast/multicast frames using normal frame transmission rules, before transmitting any unicast frames. The AP shall transmit a TIM with every Beacon frame. Every DTIMPeriod (a multiple of Beacon intervals), a TIM of type DTIM is transmitted within a Beacon frame, rather than an ordinary TIM. A PSM STA remains in the doze state for most of time and only wakes up to listen for selected Beacon frames. By reading the TIM, a PSM STA can determine if there are data frames buffered for it in the AP. If a PSM STA finds traffic indicated in a received TIM, it stays awake and issues Power Saving Poll (PS-Poll) frames to retrieve the buffered frames, one at a time, until all the packets are received; otherwise, the station goes back to sleep. A PSM STA may wakes up at any time to

transmit pending data. In standard operation, an AP performs mapping of the STA's IP address to its AID and to the associated bit in the TIM. Power management in an infrastructure networks relies strongly on the association between the STA and its current AP. An AID remains valid only as long as the STA keeps its current association. Once the STA associates with a new AP, the AID changes as well. Timing information is important in the power management mechanism. Beacon frames contain the value of the AP's clock at the moment of transmission, which will be used by the STAs to keep in synchronization with the AP's clock.

3 Virtualization of 802.11

Typically, 802.11 stations that want to connect to any AP execute following management operations: scanning, authentication and association. The STA has to repeat this entire association procedure, once it loses its connectivity. Once there is association is established, all packets destined to a node are relayed through the AP. First, the originating node transfers the packet to the AP. Second, the AP transfers the packet to the destination node. If the AP transmits a packet to a STA and it doesn't respond via an acknowledgement it retransmits the packet. This is repeated several times, until the AP deletes the node from its list of associated stations concluding that the STA is out of range.

Theoretically a STA with a single NIC can connect to multiple networks by switching the card between those networks. But every switch could lead to a disassociation, since the node is not reachable for a time frame. When the STA switches back to the previous network, it has to repeat the entire association procedure, which means long delay times until the station can deliver packets to the network. To overcome this problem an intermediate layer below IP can be introduced that manages the state information of all network connections as virtual interfaces and has also the full control over them. Two main features of IEEE 802.11 standard could be used by the intermediate layer to support real network connections on top of a single 802.11 interface without the problems discussed above.

- Power Saving Management (PSM)
- Point Coordination Function (PCF)

The concept of virtualization based on these features, will be discussed below.

3.1 Using Power Save Mode

Using the power save mode feature available in 802.11 networks, a station is able to connect to more than one infrastructure network simultaneously, without having to repeat the association procedure with every network switch.

To give an example, imagine a node **X** connected to **AP1** operating on channel 1. When it wants to communicate with **AP2** which is operating on channel 2, node **X** uses PSM with **AP1**. As explained above, **AP1** automatically buffers all packets destined to node **X**. Although **AP1** thinks that node **X** is asleep, it actually uses the time frame of a *Listen Interval* to switch its interface configuration and actively connect to **AP2**. Shortly before the expiration of the *Listen*

Interval node **X** sends a power saving request to **AP2** and switches back to **AP1**. Through the usage of the power saving mode before every network switch the node **X** stays associated with both APs all time.

3.2 Using PSM & Point Coordination Function (PCF)

Point Coordination Function (PCF) is an optional access method which enables a contention-free transmission. It is built on top of the DCF, and is used only on infrastructure networks. The AP acts as a master called the *Point Coordinator* (PC) and the STAs as salves. PCF utilizes an access priority that the PC may gain by using a smaller *Inter Frame Space* (PIFS $<$ DIFS). This access priority is used by the PC to create a *Contention Free* (CF) period, in which the PCF is executed. The CF period shall alternate with a Contention Period (CP), in which the DCF is working. At the beginning of each CF period, the PC sense the medium if the medium is idle, the AP waits PIFS then broadcasts Beacon frame containing information (like: *CFPMaxDuration*) used by the STAs to set their NAV timers. Thus, the PC gains control of the medium and DCF is prohibited. During the CF period, the PC determines which STA has the right to transmit. It grants a contention free channel access to individual STAs by polling them for transmissions one at a time. This centralized medium access control in the contention free period can be exploited to support the virtualization of 802.11 interfaces. The following example illustrates the trick used in this scenario:

A node **X** works in managed mode and is connected to an access point (**AP1**) operating on channel 1 (SSID network 1). When it wants to act as an access point, it creates a new virtual interface which is configured in master mode and SSID network2. Connection to **AP1** is put on hold as explained as above. For the duration of a *Listen Interval* node X appears as an AP to node **Y** and node **Z** and can communicate with both on channel 2. If the PCF used as an access method in network 2, node X acting as AP (PC) has the right to announce the contention free period (CFP), in which nodes are able to send traffic only if the AP has given them access. For the virtualization, the CFP is used by node **X** to defer node **Y** and node **Z** from transmission and switch back to network 1 and communicate with the AP as a normal station instead of granting any STA the right to transmit.

4 Simulation & Results

We have conducted some simulations to understand the impact of virtualization on the traffic properties. We decided to start with the PSM approach and left PCF for future research. MultiNet already has mentioned PSM as an enabler for virtualization, but they provided only results for energy consumption. In our simulations we have focused on different parameters that are related to PSM and wanted to understand how they can change the performance of the virtual 802.11 interface.

4.1 Simulation Setup

We have chosen OMNeT++ as network simulator [omn]. It supports 802.11b and provides a good framework for wireless experiments. OMNeT++ was extended in order to support the

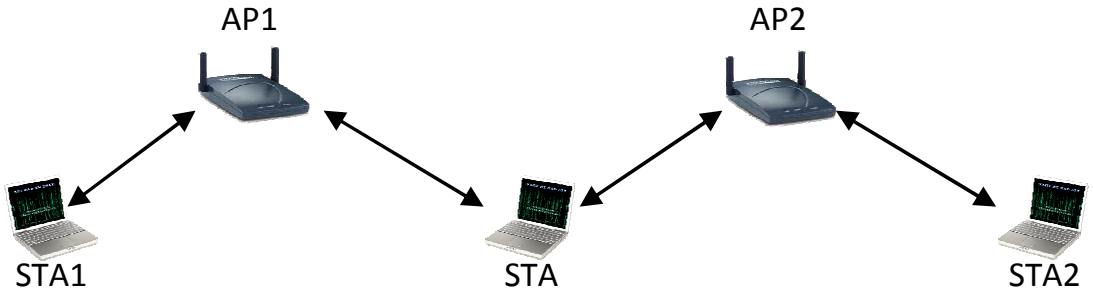


Figure 3: Simulation network

power save mode. Figure 4 depicts the nodes contained in each network in both scenarios. In all simulations, the mobile station is equipped with one single NIC and switches periodically between two different networks.

4.2 With/Without PSM

Two scenarios will be discussed here to show the benefit of virtualization of 802.11 interfaces with power saving support. The first scenario does not make usage of PSM. In the second scenario, the STA switches between its virtual WLAN interfaces to communicate with two networks without interruption (as explained in Subsection 3.1). The *Listen Interval* of the STA was set to three Beacon Intervals (100ms) in both networks. During one *Listen Interval*, the STA has to connect to the both networks. One Beacon Interval was the time in which the STA are connected to one network. Thus in each three Beacon Interval (300ms), the STA is 200ms active and 100ms in doze state.

In the first scenario, the STA loses its connectivity eleven times in 100 seconds. Every time it loses its connectivity it disconnects from the current network and repeats the entire association procedure: scan, authentication and association mechanisms. The more the number of channels the longer is the time needed for scanning. In our simulations a full scan required around 1.15/2.3 seconds. In some cases the scan mechanism is done, and STA tries to start authentication session but it receives an Authentication Error Frame from the AP so it scans again. All this period of time is just spent for scan process, and the STA is still not connected to any network. The STA needs around 2.6 ms for authentication process in this scenario and around 1.5 ms for association process. Thus 4.1 ms is the time spent by the STA to authentication and associate with an AP. But the authentication and association time could be larger with increasing the number of STAs in the network. (From other simulation scenarios, independent of this scenario, a STA needed circa 96.6 ms for the authentication and association processes as the number of STAs were four. While this time was circa 156.9ms as the number of STAs were eleven). In the second scenario the STA does not lose its association after the network switch, which only takes a very short time (negligible in our simulator). Therefore the STA in the second scenario receives more frames than in the first scenario. The throughput is accordingly higher in the second scenario. Table 1 illustrates the throughput for both scenarios in 100 seconds simulation time.

Scenario	Throughput (kbit/s)
single NIC	89.9
two virtual 802.11 interfaces	172

Table 1: Throughput in two different scenarios (with and without virtualization)

4.3 Understanding PSM and its parameters

Using multiple NICs to connect to multiple networks gives much better throughput than the virtualization mechanism, but in the other hand, multiple NICs consumes more power. Two NICs consume around double the power consumed by the virtualized NIC, as MultiNet investigated. We will study the variation in the throughput, when a STA uses virtual interfaces to connect to multiple networks. The throughput is varied by changing the operating conditions, in which a STA using virtual interfaces will be applied. Power saving feature is only useful when the traffic destined to a PSM STA is small [KKPJ05]. When the power saving is utilized to support the virtualization with a high traffic networks, a high number of frames will be dropped from the buffer of the AP when they buffered more than a listen interval and the PSM STA could not manage to poll them.

To verify this issue, we simulated a simple scenario, in which a STA used two virtual interfaces to connect to two networks. Video streaming has been used in the first network as an application to represent a high traffic. While in the second network a ping application has been slowly executed to represent a low traffic.

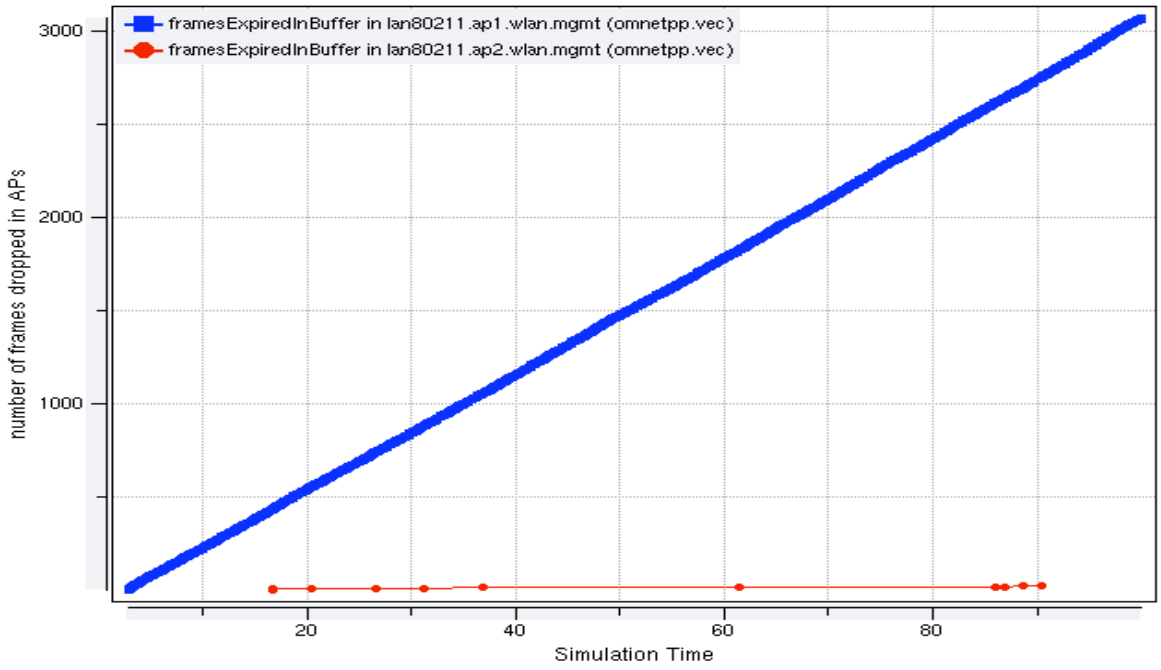


Figure 4: The number of frame dropped in the AP in high and low traffic

Figure 4 shows the number of frames dropped in two APs (each AP belong to a network), when their lifetime is finished before polled by the STA. The blue line represent the number of frames dropped in the AP1 which is received a high traffic pending to the STA. While the red line represent the number of frames dropped in the AP2.

Up now our simulation scenarios will be focused on the low traffic operating conditions. As is known, the larger the listen interval the smaller the time is, in which the STA can receive or send frames. Thus, the throughput will be decreased by increasing the listen interval value. This has been investigated in the following scenario: a STA uses two virtual interfaces to connect to two networks simultaneously. The numbers of nodes in both networks are as shown in Figure 3. Because all the parameters and the number of nodes in the two networks were identical, the throughput was approximately equal. Table 2 represents the throughput in one of both networks in several simulation runs. In each run the listen interval was set to a different value. In all simulation runs, the STA was active for 100 ms with each network. The results illustrate how the number of dropped frames in the AP is increased by increasing the *Listen Interval* value.

Listen Interval (Beacon Interval)	**Throughput (kbit/s)**
3	170
4	156.6
5	133.3
6	117.6
7	108.3
8	101.3

Table 2: Variation in the Number of dropped frames in the AP by changing the Listen Interval value

5 Conclusion

Using virtual 802.11 interfaces to connect to multiple networks simultaneously, instead of using multiple network interface cards, enables savings in energy costs, minimizes the physical space, and provides the capability to build large and small wireless mesh networks. The IEEE 802.11 PSM is usually utilized to save energy consumption when a STA has small traffic to send or to receive. We presented how this feature can be used to support our virtualization concept to allow mobile stations connect to more than one network. Through simulations we showed how network switching with virtual interface affects wireless communication.

Our results showed that the higher the traffic in the network, the smaller is the performance of the virtual wireless interface. In this case large number of frames will be dropped in the buffer of the access point because their lifetimes expire. Our simulations also showed that by increasing the Listen Interval value, the number of dropped frames will increase accordingly. The STA is only active to retrieve its buffered frames from the AP.

Therefore virtualization of 802.11 interfaces using PSM is good for low traffic networks. We recommend to utilize virtual interfaces to connect to multiple networks with low traffic, by choosing an appropriate Listen Interval value, and according to it a long period in which the STA is

awake.

For further research, more scenarios are required to figure out the throughput of using the virtualization, by changing the active and sleep times adaptively so that a STA can connect to multiple networks in different awake time periods according to the amount of traffic in each network.

Bibliography

[CB04] R. Chandra, P. Bahl. MultiNet: connecting to multiple IEEE 802.11 networks using a single wireless card. *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies* 2:882–893 vol.2, March 2004.

[Cet06] B. Cetin. Opportunistic Relay Protocol For IEEE 802.11 WLANs. Technical report, Swedish Institute of Computer Science, 2006.

[htu] HTun - HTTP Tunneling Interface. URL: http://htun.runslinux.net.

[iee07] IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pp. C1–1184, 12 2007.
doi:10.1109/IEEESTD.2007.373646

[KKPJ05] D. H. Kwon, S.-S. Kim, C. Y. Park, C. I. Jung. Experiments on the Energy Saving and Performance Effects of IEEE 802.11 Power Saving Mode (PSM). In Kim (ed.), *ICOIN*. Lecture Notes in Computer Science 3391, pp. 41–51. Springer, 2005.

[lin] Linux Wireless. URL: http://www.linuxwireless.org.

[mad] MADWiFi: Multiband Atheros Driver for WiFi. URL: http://madwifi.org.

[MLV03] M. J. Miller, W. D. List, N. H. Vaidya. A hybrid network implementation to extend infrastructure reach. Technical report, 2003.

[omn] OMNeT++ Discrete Event Simulation System. URL: http://www.omnetpp.org.

[PV06] C. C. Pradeep Kyasanur, N. H. Vaidya. Net-X: System eXtensions for Supporting Multiple Channels, Multiple Interfaces, and Other Interface Capabilities. Technical report, University of Illinois at Urbana-Champaign, Wireless Networking Group, 2006.

[RC05] A. Raniwala, T. cker Chiueh. Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* 3:2223–2234 vol. 3, March 2005.
doi:10.1109/INFCOM.2005.1498497

[Res05] M. Research. Virtual Wifi. URL: http://research.microsoft.com/en-us/um/redmond/projects/virtualwifi/, 2005.

[SAS08] G. Singh, A. P. S. Atwal, B. S. Sohi. Effect of background scan on performance of neighbouring channels in 802.11 networks. *Int. J. Commun. Netw. Distrib. Syst.* 1(1):19–32, 2008. doi:http://dx.doi.org/10.1504/IJCNDS.2008.017202

[SCMB07] G. Smith, A. Chaturvedi, A. Mishra, S. Banerjee. Wireless virtualization on commodity 802.11 hardware. In *WinTECH '07: Proceedings of the the second ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*. Pp. 75–82. ACM, New York, NY, USA, 2007. doi:http://doi.acm.org/10.1145/1287767.1287782

[tin] tinc - VPN daemon. URL: http://www.tinc-vpn.org.