Virtualization of 802.11 Interfaces for Wireless Mesh Networks

Yahya Al-Hazmi and Hermann de Meer
Computer Networks and Computer Communications Lab
University of Passau, 94032 Passau, Germany
yahya.al-hazmi@uni-passau.de
hermann.demeer@uni-passau.de

***Abstract*—Equipping wireless devices with multiple radio interfaces enables multi-channel communication that has attracted recent interest due to its significant benefits in terms of exploiting the available radios to increase the throughput and the creation of WLAN-based Mesh Networks. The usage of multiple network interface cards (NICs) demands higher cost, large physical space and more energy consumption. Instead of that, virtualization of NICs is a promising approach that gives a wireless device equipped with a single card the ability to connect to more than one network simultaneously. In this paper, we present two methods for NIC virtualization. With the increasing need for energy-efficient networking and for supporting a continuous delivery of real-time multimedia services (e.g. VoIP, video streaming) despite mobility in order to avoid packet loss and delay resulting from handovers, enhancements and mechanisms are needed. In this paper, we have conducted some simulations to show how virtualization of 802.11 NICs enables energy-efficient Wireless Mesh Networks and soft handover.**

***Index Terms*—WLAN, WMNs, virtualization, PSM, simultaneous connectivity, energy efficiency, soft handover**

I. Introduction

Wireless Mesh Networks (WMNs) based on IEEE 802.11 WLANs are becoming popular and believed to be a major part of the next generation wireless infrastructure. Not only the low cost, the flexibility and the ease of installation make WLANs widely used in private and public locations, but also a WLAN user has the ability to share a high-speed Internet connection even during its movement. Moreover, the high speed provided by WLANs enables multimedia services. For all these reasons and more, IEEE 802.11 based WMNs is one of the key technologies for providing flexible wireless access to the current and the future Internet. In the literature, WMNs has been defined in different ways. We refer to the definition introduced by IEEE 802.11s [2] which describes a WMN as a wireless network that based on IEEE 802.11 and consists of mesh points (mesh nodes) that perform multi-hop bidirectional forwarding between different wireless devices. The mesh points are equipped with multiple 802.11 NICs.

The usage of multiple NICs demands higher cost, large physical space and more energy consumption. Instead of that, virtualization of NICs is a promising approach that gives a wireless device equipped with a single card the ability to connect to more than one network simultaneously. Virtualization is achieved by introducing an intermediate layer below IP that manages the state information of all network connections as virtual interfaces and has also the full control over them.

However, it's preferable that WMNs provide sufficient quality and availability up to a satisfactory level. This means that permanently available Internet access for users is required. Support for a continuous delivery for real-time services despite mobility is also needed. The packet loss and the traffic delay resulting from handovers when a user roams across multiple access points (APs) in different domains become challenges for mobile users in wireless networks. Furthermore, reducing energy consumption of wireless devices is required, as the energy cost of the wireless Internet access increases with the time due to the growing number of deployed wireless devices.

The motivation behind this paper is to provide support for the emerging WMNs in terms of simultaneous connectivity, energy-efficient networking with permanent access availability and seamless mobility (soft handover). Such support is realized in this work with the help of NIC virtualization. This virtualization enables several new capabilities we have identified in our preliminary work [3]. In this paper, we study in details and examine some of these capabilities that facilitate the needed support for the emerging WMNs, and are referring to:

- *Simultaneous connectivity:* A wireless station can connect to multiple networks simultaneously.
- *Network coverage extension/ Relaying:* Stations, which are part of a wireless cell, might extend its range by creating a second virtual interface in AP mode, allowing stations that were originally outside the range of the main AP to connect to the network.
- *Energy-efficient wireless mesh networking:* One of the possible ways to save energy is to shutdown some of the APs, which serve a specific area of interest, if the number of the served stations are decreased in off-peak hours (e.g. at night, weekend, holidays). This could lead to the fact that some locations in the interested area are not covered, and stations located within these locations will not have connections. These can get connectivity by using *Network coverage extension/ Relaying* capability.
- *Soft handover:* NIC virtualization enables a background scanning approach, in which a station can use the second virtual interface to scan all available channels, while the first virtual interface is used as main interface to the current connection. After choosing the target AP, a station

can authenticate and associate to that AP without losing its initial association. As a result, packet loss and traffic delay are avoided, since the station is connected to the serving AP during the handover process.

At the heart of our contribution in this paper lie the two new ideas of enabling energy-efficient wireless mesh networking and soft handover.

The rest of this paper is organized as follows. Section 2 introduces a brief background and related work. The proposed virtualization approaches are presented in section 3. Two of the above listed usage scenarios are described in sections 4 and 5. Simulation results are discussed in section 6. Finally, the paper is concluded in section 7.

II. Background and Related Work

The concept of virtualization is widely used in different computer programming and networking areas starting from process virtualization (in use for operating systems to enable multiprogramming) up to network virtualization (e.g. VLAN, VPN, Overlays). Virtualization discussed in this paper is within the scope of 802.11 NICs.

A. Virtual WLAN Interfaces

Using one wireless NIC to simultaneously connect to multiple wireless networks are already presented in [4] and [5]. Net-X Testbed [4] assumes that two interfaces at least are available at each node; one of them is switched to a fixed channel for receiving traffic, while the second interface is switched to any of the remaining channels for sending. Packets are always transmitted to the fixed channel of the receiver. Thus, a wireless node can virtually connect to multiple networks simultaneously. VirtualWiFi, previously called MultiNet [5], is a virtualization architecture for WLAN cards from Microsoft. It abstracts a single card to appear as multiple virtual cards to the user. For achieving this, they use their own driver called *MultiNet Protocol Driver*. Our virtualization method exploits power saving feature of IEEE 802.11 instead.

B. Power Saving in IEEE 802.11

The MAC layer specification in IEEE 802.11 [1] defines two diverse power management modes a station can operate in one of them: active mode and Power Saving Mode (PSM). In active mode, a station is fully powered and is able to exchange frames at any time. While in Power Saving mode, a station is permitted to be in one of two different power states, either in *awake* state or *doze* state. An AP in a wireless network observes the mode of each station. A station must first inform its AP about changing its power management mode using Power Management bits within the Frame Control field of the frame used as a power saving request. A station shall not enter PSM before it receives an acknowledgement from the AP. During the association procedure, a station informs the AP about its listen interval, which is used to indicate a period of time for which a station in PSM may choose to sleep. In case of a successful association, an AP assigns an Association Identifier (AID) to a PSM station (a station operates in PSM) in the Association Response frame. Data frames destined to a PSM station shall be buffered in the AP and only transmitted at designated times. The Listen Interval field is used by an AP in determining the lifetime of frames buffered for a PSM station. The length of a listen interval is measured in beacon intervals. Therefore, a PSM station can sleep and miss a specific number of Beacons without losing any data frames or disconnecting from the network. For instance, considering that a station sets its listen interval to three beacon intervals. It would wake up to listen to every third beacon and check whether there is traffic buffered for it in the AP. According to the IEEE 802.11 standard, an AID is used by the AP to indicate the existence of unicast traffic buffered for the PSM station, and represents a bit in the Traffic Indication Map (TIM), which is sent within each Beacon frame. Every Beacon carries a TIM information element indicating the buffer status of all PSM stations. A PSM station remains in the *doze* state for most of the time and only wakes up to listen for selected Beacons. By reading the TIM, a PSM station can determine if there are data frames buffered for it in the AP. If a PSM station finds traffic indicated in the received TIM, it stays awake and issues Power Saving Poll (PS-Poll) frames to retrieve buffered frames, one at a time, until all packets are received. Otherwise, the station goes back to sleep. A PSM station may wake up at any time to transmit pending data. Timing information is important in power management mechanism. Beacons contain the value of the AP's clock at the moment of transmission, which is used by stations to keep in synchronization with the AP's clock.

C. Mobility and Handover in IEEE 802.11

The mobility supported by IEEE 802.11 standards is limited. Therefore, mobility management is needed when a mobile station roams between APs. To keep an ongoing connectivity session in progress, when a mobile station changes its connectivity from an AP to another, it has to execute handover process. Several attempts have been made to improve handovers, for instance, to reduce the handover delay. Those that improve handovers with the help of make-before-break mechanism are discussed in this section.

In [6] and [7], a station is equipped with two real NICs, one of them is connected to the serving AP and the second one is only used during a handover in order to avoid packet loss and delay. In [8], DualMAC method is represented in which a wireless card is configured with two MAC addresses. This configuration is done either in a WLAN driver or firmware in order to produce MAC frames with different MAC addresses. Thereby a station is able to connect to a new AP without disconnecting to the current one.

In this paper, we present a novel method for a soft handover, which is based on virtualization technology. This can be made without violating the IEEE 802.11 standard. Soft/fast/smooth handover has been already represented in the above mentioned works by means of reducing (***or*** avoiding) handover delay ***or*** packet loss. While the main goal of our proposed method is to avoid packet loss and traffic delay during a handover, and thus enabling real-time applications.

III. Virtualization of 802.11 Interfaces

In this section we discuss how a single NIC can be virtualized to appear as multiple virtual NICs for the user. Theoretically a station with a single NIC can connect to multiple networks by switching the card between those networks. But every switch leads to a disassociation, since the station is not reachable for a time frame. When the station switches back to the previous network, it has to repeat the entire association procedure (scan, authentication and association) which means long delay up to 450 milliseconds [11] until the station can deliver packets to the network. To overcome this problem an intermediate layer below IP is introduced that manages the state information of all network connections as virtual interfaces and has also the full control over them. We have called this layer virtualization layer.

A. Virtualization Methods

Two main features of IEEE 802.11 standard can be used by the virtualization layer to support real network connections on top of a single 802.11 interface without re-association after each disconnection: PSM and Point Coordination Function (PCF) as we introduced in [3]. The concept of virtualization based on these features is explained in the following.

1) Using PSM: This method is based on the power save mode feature available in 802.11 networks. A station is able to connect to more than one infrastructure mode network simultaneously, without having to repeat the association procedure with every network switch. To give an example, let's suppose that a station is connected to an access point AP1, which is operating on channel 1. When the station wants to communicate with another access point AP2 operating on channel 2, it uses PSM with AP1. As explained in section II-B, AP1 automatically buffers all packets destined to the station. Although AP1 thinks that the station is in sleep mode, 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, the station sends a power saving request to AP2 and switches back to AP1. Through the usage of PSM before every network switch, the station stays associated with both APs all time.

2) Using PSM and PCF: In this method both PSM and PCF are used. PCF is an optional access method which enables a contention-free transmission in 802.11 networks [1]. It is built on top of the Distributed Coordination Function (DCF), and is used only on infrastructure networks. An AP acts as a master called Point Coordinator (PC) and stations as slaves. PCF provides access priority, thus a PC may gain a higher priority by using a smaller Inter Frame Space (PIFS $<$ DIFS). This access priority is used by the PC to create a Contention Free Period (CFP), in which PCF is executed. The CFP shall alternate with a Contention Period (CP), in which DCF is working. At the beginning of each CFP, the PC senses the medium. If it is idle, the PC waits PIFS then broadcasts Beacon containing information (e.g. CFP-Max-Duration) used by stations to set their NAV timers. Thus, the PC gains control of the medium and DCF is prohibited. During a CFP, the PC determines which station has the right to transmit. It grants a contention free channel access to individual stations by polling them for transmissions one at a time. This centralized medium access control in the CFP can be exploited to support the virtualization of 802.11 interfaces. The following example illustrates this method: A node X works in managed mode (as station) and is connected to an access point (AP1) operating on channel 1 (SSID: network-1). When node X wants to act as an AP, it creates a new virtual interface which is configured in master mode, as a matter of fact a new network (SSID: network-2) is created. Connection to AP1 is put on hold as explained in the PSM method. 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. The PCF is used as an access method in network-2. Node X acting as an AP (PC) has the right to announce the CFP, in which stations are able to send traffic only if the PC has given them access. For the virtualization, the CFP is used by node X to defer nodes Y and Z from transmission and switch back to network-1 and communicate with the AP1 as a normal station instead of granting any station the right to transmit.

B. Virtualization Challenges

Virtualizing a single NIC by adding an intermediate layer above MAC layer presents some challenges that have to be solved to get an appropriate virtualization platform. The biggest challenges and their solutions are:

- A *switching mechanism* is required to support switching the physical NIC across the networks to which the station is connected. This is one of the main tasks of the virtualization layer. However, the delay caused by switching between two networks (*switching delay*) has to be as short as possible. Since the virtualization layer keeps states of the associated networks, the *switching delay* is negligible (especially if Native Wi-Fi cards are used [5]).
- Switching between multiple networks results on *Packet loss*. To avoid this loss, a *buffering mechanism* is required to buffer packets for networks that are currently inactive. These packets are delivered to/from a network once the station is active on the corresponding network. PSM feature of IEEE 802.11 standard is used by the virtualization layer to support buffering function.
- *Synchronization* with the networks to which the station is connected is required to switch between them on the correct times. In infrastructure mode, timestamps of Beacons received from an associated AP are used to keep stations in synchronization with that AP.

The applicable scenarios of NIC virtualization are discussed in details in the following two sections, namely energy-efficient WMNs and soft handover.

IV. Energy-Efficient Wireless Mesh Networking

Common wireless networks depend on a small number of APs or hotspots to connect users. In a WMN, the network connection is spread out among a large number of wireless mesh nodes that communicate with each other to share the

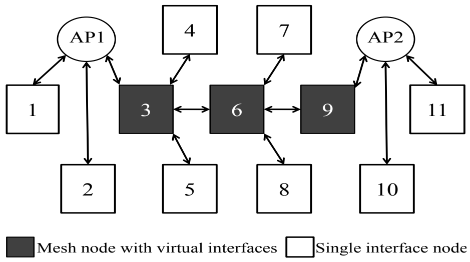

Fig. 1. A sample of mesh networking architecture supported by virtual links.

connection across a large area [2]. This section presents WLAN-based wireless mesh networking with the help of virtualization of 802.11 interfaces. Moreover, a method for reducing energy consumption in WMNs is also introduced.

A. Wireless Mesh Networks and Virtual Links

Using virtualization of 802.11 interfaces, nodes can build a multi-hop WMN. Fig. 1 illustrates a simple example of mesh networking architecture, which is built by using the proposed virtualization methods. There are two types of nodes in this architecture, white and black nodes. Each node is equipped with a single NIC. The NIC in a white node is configured to work in managed mode and connected only to an AP. While a black node represents a mesh node and uses the proposed virtualization methods and thus can create two virtual interfaces to connect to two wireless networks operating in different channels.

In this example, node 3 is connected as a normal station to AP1 (SSID: network-1) via one of its virtual interfaces configured to work in managed mode, and acts as an AP by configuring its second virtual interface in master mode; as a matter of fact a new network (SSID: network-2) is created. Node 3 appears as an AP to nodes 4-6. Node 6 operates like node 3, such that it is connected as a station to node 3, and viewed by nodes 7-9 as an AP. Node 9 configures its two virtual interfaces to work in managed modes. One of them is used to connect to node 6 and the other one is used to connect to AP2. These four wireless networks form a multi-hop WMN. In this way, a large WMN can be formed with help of NIC virtualization. If one of the APs has access to the Internet, a large number of nodes have the ability to connect to the Internet too. In the previous example, if only AP1 has an Internet access, nodes 1-9 can connect to the Internet, while nodes 10 and 11 have no access to outside world. Of course network address translation protocol is needed to support the routing through the entire mesh network.

B. Energy-efficient Wireless Networking

It is aimed to reduce energy consumption in wireless Internet access architectures, since the number of wireless devices is growing rapidly. In wireless networks like WLANs, several parameters play a role in energy consumption, such as: number of APs covering a particular area, number of stations in the covered area, locations of APs, their transmission power, traffic load, etc.

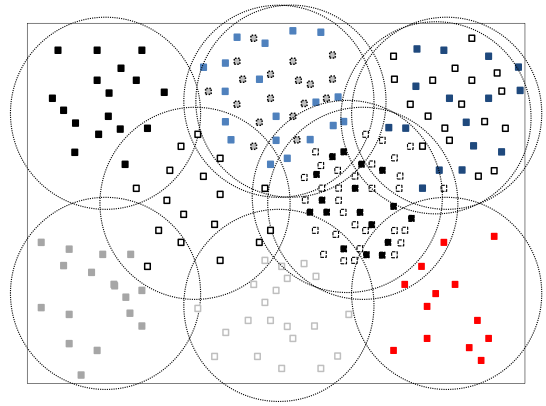

Fig. 2. Stations served with 11 powered-on APs in a specific covered area.

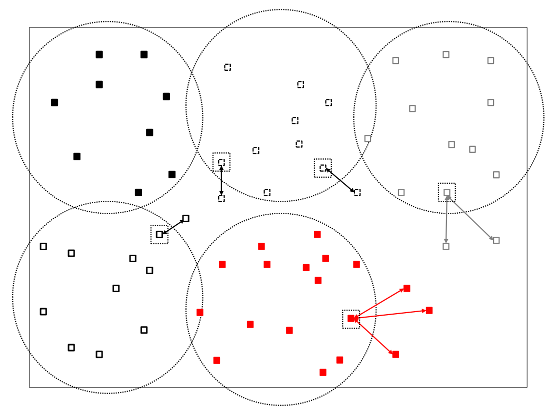

Fig. 3. Stations served with five powered-on APs in a specific covered area, some stations are equipped with virtual NICs to act also as APs and serve the stations existing in uncovered locations.

A scenario is represented in Fig. 2 and Fig. 3, in which a specific area is covered by a number of APs. This number is reduced by shutting down some of them if the number of the served stations is decreased in off-peak hours. In Fig. 2, the whole area is served by eleven powered-on APs. Each circle represents an AP's coverage area (wireless cell). Three of the cells are each served with two APs in order to increase the capacity of the wireless networks. The small symbols inside these cells represent stations. Stations with the same shape and color are served by an AP. While in Fig. 3, only five APs are powered-on to reduce the consumed energy.

The number of APs is only reduced since the number of currently jointed stations is smaller than those in Fig. 2. It's clearly shown in Fig. 3 that the remaining powered-on APs do not cover the whole area. Thus, stations located outside the coverage range of the APs can get access to the Internet via stations (work as relays) connected to any served AP. This is achieved by creating networks between the stations outside and inside the covered area. The inside stations, which work as relay stations, are able to create two simultaneous connections. As depicted in Fig. 3 some stations are surrounded by dotted squares. These are equipped with virtualized NICs so that a station uses one of its virtual interfaces to create a network with other stations while keeping its connection to the serving AP using another virtual interface. Such a station becomes a bridge for other stations on the network (lying in uncovered locations) passing their packets to and from the Internet. However, since relay stations are always active, the energy consumed for extra transmitting and receiving of traffic for others is not high. A station consumes about 0.9 W (idle), 1.3 W (transmitting), and 1.9 W (receiving) [13]. By comparing these values to the consumed energy by an AP, which is up to 20 W (e.g. Cisco Aironet 1250 Series AP), it becomes clear that switching off an AP implies much more energy saving potential than the overhead generated on the relay stations. In addition to the energy saved by shutting down some APs, NIC virtualization plays a role in reducing energy consumption.

However, there is a need for seamless mobility across such WMNs. For this purpose, a soft handover method is discussed in the next section.

V. Soft Handover

In this section, we describe a method for performing soft handover in link (and higher) layer(s) via NIC virtualization.

A. Link-layer Handover

In a handover process, a station loses its association because it cannot be associated to more than one AP simultaneously. Unfortunately, real-time traffic is extremely affected by handovers which introduce packet loss and handover delay. Therefore, the proposed virtualization-based method gives the station the ability to keep its current connection and using another virtual interface to perform handover. A station can execute background scanning, authenticate and associate to the chosen AP without disrupting its association to the serving AP. In the proposed method, the NIC is only active on one network for a specific period of time before switching its NIC to another network. Therefore, the station informs its serving AP that it would enter PSM in order to be able to switch its physical card to another channel. The station is in a *doze* state from the prospect of the currently serving AP but it uses another virtual interface to start handover. The serving AP buffers traffic destined to the station. This traffic is polled by the station when it switches back to the serving AP using the main virtual interface. Thus, the station does not lose its traffic during its absence from the serving AP. Moreover, packets received in the station from the application layer are buffered if the serving AP is currently unreachable, but will become reachable in the near future. These packets are delivered to the serving AP once the station is active again.

The virtualization approach is used to keep an ongoing connection and to avoid packet loss. But buffering packets for a long time in an AP during *doze* state is a critical issue for delay-sensitive applications. Therefore, the way in which background scanning, authentication and association processes are performed is the key to avoid delaying packets for a long time. Fig. 4 illustrates a handover in link and network or application layers, while a station is in VoIP session. When a station decides to perform a handover, it informs its serving AP that it goes in *doze* state. Then it can start with a scanning process by sending a probe request in the first channel. If it receives a response, it stores information about the AP that sent the probe response. This process can be done in *few ms*. The station switches back to the serving AP and polls its buffered packets and continue its session using the main virtual interface. After a specific period of time, it switches its card again to scan the next channel. It scans only one channel in each time it switch its card. The station alternates between execution handover-related processes (scan all available channels, authentication and association with the target AP) and communication with its currently serving AP. Even during a handover process, a station communicates with its serving AP. It executes handover-related processes intermittently as shown in Fig. 5 which illustrates the activities of the two virtual interfaces during a handover.

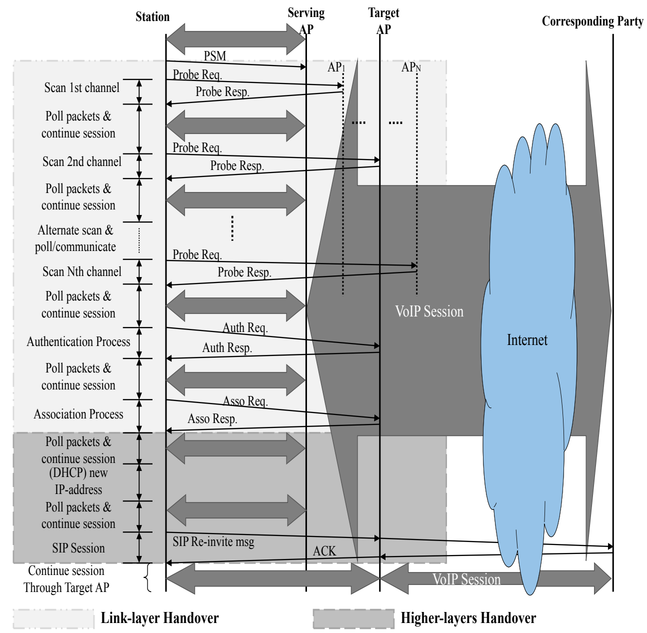

Fig. 4. Soft Handover.

B. Application- or Network-layer Handover

A mobile station executes only a link-layer handover, if a handover takes place between two APs in the same subnet. The traffic is routed unproblematic to the target AP. If the serving and the target APs are in different subnets, a station firstly performs handover in link layer then in network layer. In case of a VoIP session, Session Initiation Protocol (SIP) is utilized to execute application-layer handover without disconnecting the current session. However, after getting a new IP-address in the new subnet, a station sends a SIP RE-INVITE request to the corresponding party in the current session to update its IP-address via the initial AP. With this regard, the network and application layers are notified about such change in order to update network and application connectivity information. Since the station is simultaneously connected to two APs, it can perform application-layer handover process through the

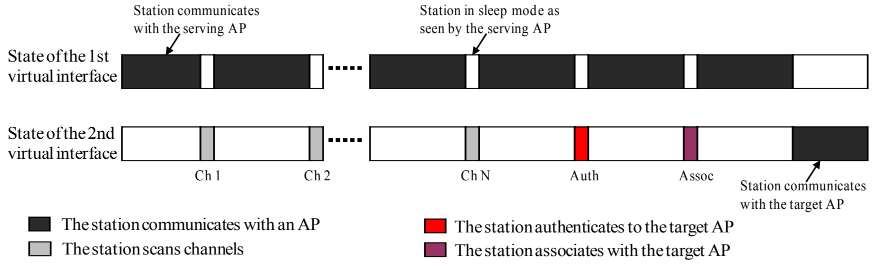

Fig. 5. View of virtual interfaces' activities during a handover process.

target AP using the dedicated virtual interface, while exchanging traffic through the serving AP at the same time. Once the IP-address of the station is updated in the corresponding party, the traffic is routed to the associated AP in the new subnet.

VI. Simulation-Based Results

In this paper we decided to start with the PSM-based virtualization approach and left the PSM&PCF-based approach for future work. Therefore, we focus on one hop infrastructure based networks. VirtualWiFi [5] has already mentioned PSM as an enabler for virtualization, but they provided only results for energy consumption. In our simulations we have focused on different PSM-related parameters and wanted to understand how they can change the performance of the virtual 802.11 interfaces used for simultaneous connectivity in WMNs. Furthermore, we focused on investigating the performance of our virtualization approach to enable soft handover.

A. Simulation Setup

We used OMNeT++ as network simulator [14] and its associated INET-framework which is intended to support wireless and mobile simulations within OMNeT++. IEEE 802.11b standard is implemented as a WLAN model in the INET-framework. We have extended this model in order to support the power save mode and also implemented the virtualization layer below IP layer discussed in section III.

B. Simultaneous Connectivity in WMNs

The benefit of using virtual 802.11 interfaces in WMNs to enable the mesh nodes to create multiple simultaneous connections is discussed here. These mesh nodes are parts of a WMNs. The use of multiple NICs to connect to multiple networks gives much better throughput than a virtualized NIC, but multiple NICs consume more power. Two NICs consume around double the power consumed by a single virtualized NIC, as investigated by VirtualWiFi [5]. As our virtualization mechanism is based on PSM, we study some important parameters which have a high effect on the throughput, such as traffic load, packet buffering and chosen listen interval.

1) Performance of packet buffering: Power saving feature is only useful when the traffic destined to a PSM station is small [9]. Therefore, when the power saving is utilized to support the virtualization with a high traffic networks, a high number of packets would be dropped from the buffer of the AP when they buffered more than a listen interval and the PSM station could not manage to poll them. To verify this issue, a simple scenario is simulated in our preliminary work [3], in which a station uses two virtual interfaces to connect to two APs. Each AP belongs to a separated network. In the first network, a video streaming is used as an application to represent a high traffic. While a ping application is slowly executed in the second network to represent a low traffic. The simulation results show that the number of packets dropped in the high loaded AP was around 3000 packets in 100s, while the dropped packets in the low loaded AP were circa 20.

2) Effect of Listen Interval on the throughput: In the following we focus on the amount of traffic destined to a PSM station with different listen interval values. As is known, the larger the listen interval the larger is the buffer size needed in the AP to guarantee buffering packets for maximum one listen interval. If a PSM station uses virtual interfaces to connect to multiple networks simultaneously, it has to choose a listen interval that is large enough to be able to switch between all available networks. That means if a station wants to connect to four networks, it shall set the listen interval to five beacon intervals. The station is only one beacon interval active with each network. Consequently, the awaken time in each network is small in comparison to the sleeping time. This can have a negative impact on the throughput. But to what extent the throughput is affected, is being investigated in the following scenario: a station uses virtual interfaces to connect to multiple networks simultaneously. This simulation scenario is executed in different runs, which differ from each other in the chosen listen interval. The number of possible network connections depends on the chosen value. For instance, setting the listen interval to four enables three possible connections. In all simulation runs, the station is only one beacon interval (100ms) active with each network. Video streaming packets were generated every 33ms based on *H.263 codec* [12]. In 100 second simulation time, the number of the received and the lost packets by the station from one network in several simulation runs are represented in Table I. It indicates how such numbers are varied by increasing the listen interval (which means a large sleeping time). This leads to increase the number of dropped packets, because the station is only active for one beacon interval. The results show that using two virtual interfaces to create two networks does not cause packet loss, as the listen interval was set to 3. With the same value, another simulation run is executed to see how this case

TABLE I
VARIATION IN THE NUMBER OF RECEIVED AND LOST PACKETS BY CHANGING THE LISTEN INTERVAL

Listen Interval (Beacon Intervals)	No. of received packets	No. of lost packets
3	2969	0
4	2546	408
5	2031	902
6	1635	1316
7	1435	1519
8	1222	1721

is influenced by changing the packetizing interval from 33ms to 20ms. This results in loss of packets. This is because within a listen interval (300ms), the server generates 15 packets. But due to the exchange frames needed by the CSMA/CD protocol and the limited data rate in 802.11b to 1Mbps, the station was able to poll in average 7.5 packets in a beacon interval. Since the station is only one beacon interval active with each network, it received only 2540 packets with 2326 lost packets in 100 second simulation time. Packet loss has to be reduced. This can be improved using 802.11e or 802.11n. They support data link frame aggregation protocol, in which a station can receive a group of packets together as a response to one PS-Poll frame, thus saving DIFS, contention period and ACK. Another possible way is to adapt the activity periods in the networks to which a station is simultaneously connected.

3) Adaptive activity period: In order to reduce packet loss, the time in which the station is active with any network has to be sufficient to receive all its destined traffic. For instance, a station wants to connect to two networks one of them introduce high traffic while the other not. The station can choose its listen interval to one beacon interval and be active with the first network 70ms and only 30ms with the second one. If the packetizing interval was set to 20ms in the first network, around 15 packets are generated in 300ms. Every 300ms, the station is 210ms active in the first network. Based on our simulation platform, the station are able to receive 15 packets every 200ms. Such adaptive activity period seems to be a good solution. The traffic delay and jitter caused when the station disappears are other challenges for real-time applications. This does not discuss here.

C. Soft Handover

We have conducted some simulations to investigate the performance of our virtualization approach to enable soft handover by performing a handover process while the station is continuing its current session with the serving AP.

1) Simulation scenarios: Two simulation scenarios were executed to compare performing a handover process with and without using the proposed virtualization method. In the first scenario, a traditional handover is performed, while in the second scenario, a soft handover is executed using the proposed solution as explained in section V. Both scenarios were executed using the same topology and simulation parameters (deterministic values). The topology used in these simulation scenarios consists of a station, four APs and streaming server. All APs are connected to the streaming server. VoIP application is not implemented in the used simulator. Hence, the video streaming is used in this simulation as a real-time multimedia streaming application. At the beginning of the simulation in both scenarios, the station connects to the best AP with the strength received signal. The station moves during the simulation run. When the received signal from the currently serving AP becomes weak, the station decides to perform handover. As aforementioned, real-time traffic is extremely affected by handover that introduces packet loss and delay. These are discussed as suitable criteria in this comparison. However, to execute a soft handover (in the second scenario) without disturbing the ongoing session, some conditions are to be taken into consideration.

The packetizing interval (*PI*) is defined as a function of the frequency of generating packets (*fr*). In case of video/audio real-time traffic, *fr* is within the following interval $fr \in [33\ packets/s, 50\ packets/s]$, which means $PI \in [20ms, 30ms]$. In the executed simulations, we have set the value of *fr* to 50 packets per second (the common used value). UDP packets were generated by the server in a Constant Bit Rate every 20ms. According to these simulations, the time needed for the station to scan one channel was around 1.26ms, around 2.6ms for authentication process and 1.5ms for association process. But these values are affected by the number of stations competing for access to the medium. From other simulations we have conducted, a station needed in the worst case circa 156.9ms for the authentication and association processes, as the number of coexisting stations were eleven instead one. Therefore, the longer the time, in which a station disappears from its serving AP and uses its second virtual interface for scanning, authentication or association process, the better you ensure the success of executing any of these processes. We called this time as switching time (t_{switch}). We have chosen the switching time (t_{switch}) according to the following assumptions:

1. For successful execution of any of the above mentioned processes,

$$t_{switch} > 5ms \tag{1}$$

2. To avoid real-time traffic delay, the switching time (t_{switch}) is chosen according to the following assumption:

$$t_{switch} \leq PI \tag{2}$$

From the above definition:

$PI = 20ms$, since $fr = 50\,packets/s$

Because (1) and (2), we are choosing: $t_{switch} = 20ms$

Furthermore, MaxChannelTime was set to 20ms in both scenarios.

2) Results and Analysis: Table II illustrates the results of the simulation scenarios that have been executed. The handover duration, the traffic delay (the time in which the station was unreachable) and also the packet loss are represented. The time needed for scanning is bound to the number of channels. In the first scenario, stations needed around 231ms

TABLE II
HANDOVER DURATION, TRAFFIC DELAY AND PACKET LOSS
WITH/WITHOUT OUR VIRTUALIZATION APPROACH

Handover Type	Handover duration (ms)	Traffic delay (ms)	Packet loss (pckts)
Conventional Link-layer HO	235 [1]	235 [1]	12
Conventional Network-layer HO	1500	1500	75
Link-layer HO using virtual NIC	1104 [1]	0	0
Network-layer HO using virtual NIC	1500	0	0

for scanning 11 channels. The time needed for the scan mechanism in the worst case was around 413ms. This case occurred when a scan mechanism is executed and the station tries to start authentication session, but one of the exchanged authentication frames is lost or collided. In this case the station scans again. The time needed by the station to authenticate and associate to the target AP was 4.1ms. But if the station is not able to authenticate or associate to an AP within the switching time (20ms), it tries again in the next switch. According to our setup, the overall time needed to execute the link-layer handover in the first scenario was around 235ms as shown in Table II[1]. The time that is spent on executing the network-layer handover was set in both simulation scenarios to 1500ms according to the result from [10]. The video streaming traffic was delayed around 1735ms with 87 lost packets in the first scenario. The results from the second scenario showed that performing handover using the proposed method enables continuing the communication session without packet loss. However, the handover duration is a bit longer than that spent in a conventional handover mechanism, because the station was able to receive its destined traffic during the handover duration. Traffic is delayed in the worst case for 10-110ms in the buffer of the serving AP. This occurs very rarely if a station misses receiving Beacon after switching its NIC back to the serving AP and has to wait for the next Beacon (beacon interval was set to 100ms) to find out whether it has buffered packets in the AP. This delay is for some Internet applications acceptable (web browsing, file transfer), and is also acceptable with constraints for some applications (VoIP). But for bidirectional applications like games, there is no delay tolerance. However mobility occurs rarely! Otherwise, cross-layer communication between the WLAN MAC layer and the application layer is recommended.

VII. CONCLUSION

The use of virtual wireless interfaces to connect to multiple networks simultaneously instead of using multiple physical NICs enables saving energy, minimizes the physical space, and provides the capability to build large and small energy-efficient WMNs. In this paper, we presented how the PSM in IEEE 802.11 can be used to support virtualization of 802.11 interfaces. Through simulations we showed how network switching with virtual interfaces affects simultaneous connectivity in WMNs. The simulation results showed that the higher the traffic in the networks, the smaller is the performance of the virtual 802.11 interfaces. Our simulations also showed that the performance is highly depending on the choice of the activity periods in each network. Therefore, using the PSM-based virtualization approach is more suitable for low traffic networks. We showed how NIC virtualization enables soft handover so that a handover process is performed without packet losses and without delaying the traffic which belongs to an ongoing session. For further work, more scenarios are required to figure out the optimal configurations in order to improve the performance. For instance, the activity periods in the networks to which the station is simultaneously connected can be dynamically adapted according to the amount of traffic in each network. From the obtained results, virtualization of 802.11 interfaces is promising for some applicable scenarios; therefore a real implementation is a possible future work.

[1]In the worst case: a station spent 413ms for scanning. Authentication and association take longer time with a high number of coexisting stations

ACKNOWLEDGMENT

Parts of the work in this paper were undertaken in the context of the MOMO and EEWMI projects (EuroNF Network of Excellence IST 216366). We would like to thank Hakan Coskun for his help in the initial steps of this work.

REFERENCES

[1] *Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE Standard 802.11, June 2007.

[2] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Simple Efficient Extensible Mesh (SEE-Mesh) Proposal*, IEEE 802.11s.

[3] H. Coskun, I. Schieferdecker and Y. Al-Hazmi, "Virtual WLAN: Going beyond Virtual Access Points," *ECEASST*, vol. 17, March 2009.

[4] P. Kyasanur, C. Chereddi, and N. H. Vaidya, "Net-X: a multichannel multi-interface wireless mesh implementation," *ACM SIGCOMM Computer Comm. Rev.*, vol. 11, no. 3, pp. 84-95, July 2007.

[5] R. Chandra, "A Virtualization Architecture for Wireless Network Cards," PhD, thesis, Cornell University. January 2006.

[6] E. Ivov and T. Nol, "Soft Handovers over 802.11b with Multiple Interfaces," *in proc. ISWCS2005*, pp. 549-554, September 2005.

[7] V. Brik, A. Mishra. and S. Banerjee, "Eliminating handoff latencies in 802.11 WLANs using Multiple Radios: Applications, Experience, and Evaluation," *in proc. IMC2005*, pp. 299-304, October 2005.

[8] S.-L. Tsao and P. H. Lo, "DualMAC: A soft handoff mechanism for real-time communications in secured WLANs," *Computer Communications*, vol. 30, pp. 1785-1793, February 2007.

[9] D. H. Kwon, S. S. Kim, C. Y. Park, and C. I. Jung, "Experiments on the Energy Saving and Performance Effects of IEEE 802.11 Power Saving Mode (PSM)," *in proc. ICOIN'05*, LNCS 3391, pp. 41-51, February 2005.

[10] Y.-S. Chen, M.-C. Chuang, and C.-K. Chen. "DeuceScan: Deuce-Based Fast Handoff Scheme in IEEE 802.11 Wireless Networks," *IEEE Trans. on Vehicular Technology*, vol. 57, no. 2, pp. 1126-1141, March 2008.

[11] A. Mishra, A Shin, and W. Arbaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process," *ACM SIGCOMM Computer Comm. Rev.*, vol. 33, no. 2, pp. 93-102, April 2003.

[12] J. C. Wu, C.-H. Huang and R.-T. Sheu, "Performance Study of AAL2 Protocol for Low-Bit-Rate Multimedia Services," *ICOIN*, pp. 793-798, January 2001.

[13] R. Mangharam et al., "Optimal fixed and scalable energy management for wireless networks," *in proc. INFOCOM2005*, vol. 1, pp. 114-125, March 2005.

[14] The OMNeT++ Discrete Event Simulation System and INET-framework. [Online]. Available: http://www.omnetpp.org.