Opnet Simulations of Handover Mechanisms for IEEE 802.16

Itamar Cohen,
Faculty of Electrical Engineering,
The Technion, Israel Institute of Technology
ofanan@tx.technion.ac.il
Revision 1, Mar-06

Abstract

Keeping high performance of TCP connections and real-time applications during handover is one of the hardest challenges of wireless data networks. The current known schemes suffer from losses and retransmissions data during the Handover.
LPM (Least Packet Marking), is a recently-proposed scheme, which tries to minimize the handover delay by integrating MAC and network layer handovers efficiently.
This paper presents Opnet simulations of the LPM handover scheme. The main result is that LPM is a very promising scheme, which indeed may enable keeping high throughput even under very hard scenarios. However, deploying LPM for every handover in the network may be very costly in terms of network resources. Also, one should keep in mind that due to the current lack of an accurate modeling of 802.16 Technology on the Opnet Modeler, the simulations still suffer from a bit of a lack of accuracy.

List of contents

1. Introduction

The IEEE committee initiated the project 802.16[1] which standardizes Wireless Broadband Access technologies, which achieve higher data rate and wider cell range than the present, existing Wireless Local Area Networks (WLANs).

A major obstacle for the wide deployment of the 802.16 standard is the problem of the handover (HO), which causes packet losses and service disruptions. For the TCP, lost packet cause sever degradation in the performances due to the congestion control mechanism, which dictates retransmission of the lost packets and a decrease of the transmission rate. Multiple lost packets may cause returning of the sender into the slow start phase.

LPM [3] is a sophisticated mechanism, which is aimed to enable a fast, almost seamless handover, by integrating the handovers of the MAC and the network layer efficiently.

In [3], the authors used the NS-2 network simulator for examining LPM. Theirs results demonstrated how deploying LPM may enable a totally seamless HO. The aim of this paper is to extend their work, by using a different simulator – the Opnet Modeler [7]; and by testing LPM under a much wider collection of scenarios.

The remainder of this paper is organized as follows: Section 2 summarizes 802.16e and LPM HO schemes. Section 3 describes the Opnet model. Section 4 details the simulations results. Section 5 summarizes and concludes this paper.

2. Background

2.1 Overview of 802.16e and Least Packet Marking (LPM)

In this section I briefly describe the 802.16e standard and the LPM scheme. The reader is referred to [2, 3, 4] for further reading. Please note that the description of 802.16e below refers to the draft versions, which preceded the formal standard [2], which hadn't been available yet during the work on this report.

802.16e is an IEEE standard which specifies the handover procedure. According to this model, a *Base Station* (BS) periodically broadcasts a neighbor advertisement management messages. A *Mobile Subscriber Station* (MSS) decodes this message, and extracts information regarding the characteristics of the neighboring BSs. If MSS identifies that the signal strength received from the current *serving BS* is too low, it may select a *target BS*, and prepare to the future HO by performing ranging and association procedures with this BS.

The MAC layer (L2) HO is performed in two phases. During the first phase, the HO pre-registration phase, the target BS is selected, but the connection to the currently serving BS is maintained. During the second phase, the real HO phase, MSS releases the serving BS, and re-associates with the target BS.

From now the term "*the MSS"* will be used for denoting a specific MSS, which performs an HO procedure.

Before looking into the HO procedure in greater detail, let us first describe our network model (figure 1). The network is consists of an MSS, which communicates over TCP with a *Correspondent Node (CN)*. The local domain is formed in a tree structure, in which the leaves are BSs, and the rest of the nodes are *Packet Access Routers (PARs)*. For global mobility, the MSS is registered on the *Home Agent (HA)* with the address of the *gateway PAR*, which is the tree's root. Each PAR uses the periodic routing updates for maintaining a routing cache, and upwards data packets sent by MSS.

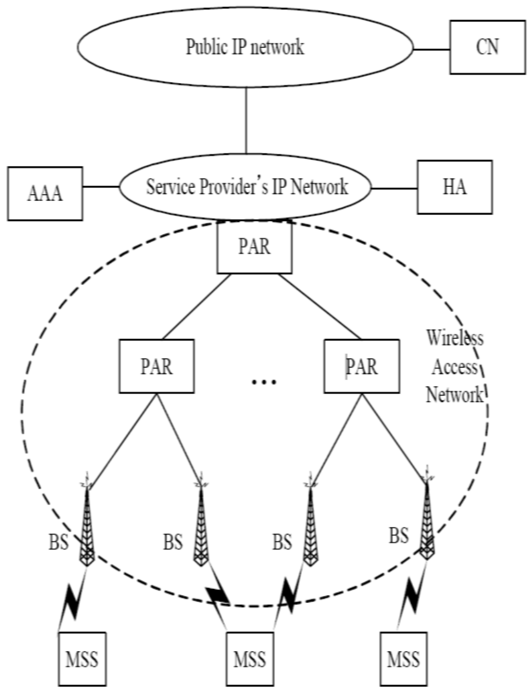


Fig. 1. Wireless Access Network Model

Figure 2 shows 802.16e HO procedure as well as the additions of LPM, which are shown in ***bold and italic.***

Either the serving BS or the MSS may initiate an HO. Then, the serving BS sends to the candidate target BS(s) an HO-pre-notification (1); the IP address of the MSS is added to the original message. Each candidate target BS decides whether or not to accept this MSS, and informs the serving BS of its decision via an HO-pre-notification-response which includes either ACK or NACK (2). LPM requires adding to this message also a *pre-routing update*, which includes the MSS's IP address. By pre-routing update message, a routing entry in the routing cache of PARs is added in the path through the target BS to the cross-over PAR, which is the branching ancestor of the serving BS and the target BS; these added entries will enable the bi-casting of packets later. The target BS prepares a buffer, which would be dedicated for the TCP packets towards the MSS (3).

When the crossover PAR receives the pre-routing update message, it starts to bi-cast all the data packets toward the MSS in the direction of both the serving and target BSs (4). The PAR that receives the pre-routing update can know whether it is a crossover or not by looking up the routing cache. If another different routing entry for the MSS is in the cache, then it is a crossover PAR.

After the serving BS receives the HO-pre-notification-response (5), it exchanges HO-RSP (6) / HO-IND (7) with the MSS, including the target BS information.

The real HO starts when MSS performs ranging and association with the target BS (9). Soon after that, the MSS sends a routing update message (10) towards the gateway PAR. When the target BS receives the routing update, it forwards to the MSS the buffered packets. The target BS also forwards the routing update upwards in the tree, towards the gateway PAR. When this message arrives to the crossover PAR, it stops the bi-casting of packets to the MSS – from now and on data packets towards MSS will be routed only via the target BS.
The mechanism described by now guarantees that no data packets would be lost (and thus, re-transmitted by CN) because of the HO. However, it does not guarantee that the same packet doesn't arrive to the MSS twice – first via the serving BS, and then via the target BS; such a scenario may cause acute degradation in the TCP performances.
Thus, the target BS should filter out the duplicated packets. However, the IP layer doesn't know the TCP sequence number. So, when the MSS sends the routing update just after the real HO, the information on the last packet received from the serving BS before the real HO can be sent to the target BS. This information is the resulting value of the hash function of (IP Header + Fixed Size IP Payload). When the target BS receives this hash value, it finds the matched packet in the buffer, filters out the duplicated packets and only forwards the following packets to the MSS.

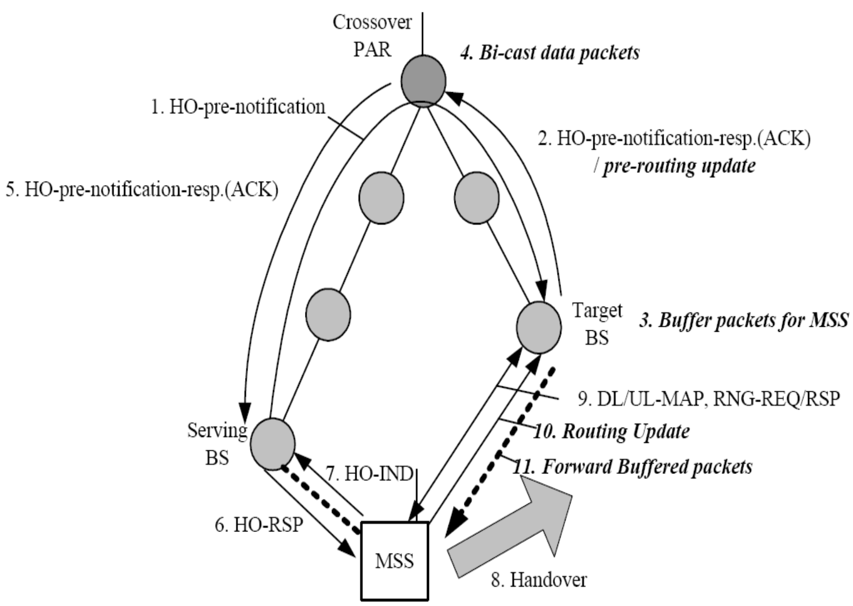


Fig. 2. LPM Handover procedure

2.2 Critique of LPM scheme and possible improvements

2.2.1 LPM's overhead

Deploying LPM requires the addition of a few control messages to those of 802.16e procedure, and the addition of a few fields to exiting messages.
It also adds some computational work to the PARs and the BSs, due to the addition of entries in the routing cache and the duplication and the buffering of data packets.

The most significant overhead, however, stems for the network resources consumed by the duplicated packets. In cases where several target BSs send the HO-pre-notification-response with ACK, many crossover PARs bi-cast the data packets. In the worst case, each level in the tree contains one exclusive leaf, which is an optional target BS, which sends the HO-pre-notification-response with ACK. In that case, each level in the tree contains exactly one crossover PAR. As each crossover PAR duplicates packets, the number of duplications is identical to the number of levels in the tree.

The length of the time interval, during which data packets are bi-casted, is also a very important factor. This length is bounded by both the routing update timeout and the time interval between the acceptance of the **pre-routing update** and the **routing update** by the crossover PAR.

A possible solution for the high overhead of LPM may be deploying it on a selective, QoS-like base: LPM may be fully implemented only in case of high-priority sessions, while the lower-priority sessions would use simpler, cost-efficient, HO scheme, or a limited version of LPM, where the target BS allocates only a small buffer space for the bi-casted packets.

Also, one may think of additional control messages (or addition of fields to the existing messages), so that the number of cross-over PARs, which duplicate data packet, would be significantly reduced.

2.2.2 Transmission rate of the buffered packets

Simulations show that in practice, usually the number of buffered packets is quite low. However, if the link-layer HO delay is longer, BS sends to MSS many buffered packets, and thus one should define the exact transmission rate of these packets.

2.2.3 Different network topologies

When the routing algorithm is either a Load-Balancing algorithm; an algorithm which deploys frequent dynamic routing changes; or any other routing algorithm which doesn't obey the tree network topology assumption, using LPM requires many adjustments.

3. The Opnet model

3.1 The Project model

The project model is an implementation of a basic HO scenario, where there exists only one MSS, which moves via a pre-defined trajectory, starting next to the serving BS and ending next to the target BS.

The serving BS and the target BS are hard-coded, i.e. a BS may be either a serving BS or a target BS (but not both) according to its node name. This is a simplification of the general case, where each BS may be a serving BS of a few MSSs, and a target BS of a few others MSSs, simultaneously (see *future work* under section 5).

The trajectory of MSS is defined in the file MSS.trj.

The file *HO.h* is a common header for all the processes' C++ source file (see also 3.3). Among other parameters, this file contains the times for the initiating of various stages in LPM HO procedure (e.g. the time of the HO_PRE_NOTIFICATION). Thus, each change in *MSS.trj* requires changing of the timing parameters in *HO.h* and re-compilation of the process files, which include it.

The *wired* communication is done over Ethernet MAC L2. The *wireless* communication is modeled by 802.11, as no standard 802.16 model is available in Opnet yet.

The project is consists of two scenarios: *Wireless* is an implementation, which uses the exact parameters' values used in the NS-2 simulations in [3]. *Wireless_11MB_100MB* is a scenario, which models a much higher data rate communication, using 100BaseT wire links and 11Mbps wireless 802.11 communication (see 4.6). For saving disk space and complexity, no other custom scenarios were saved, and one may build his own custom scenario by modifying node models' attributes.

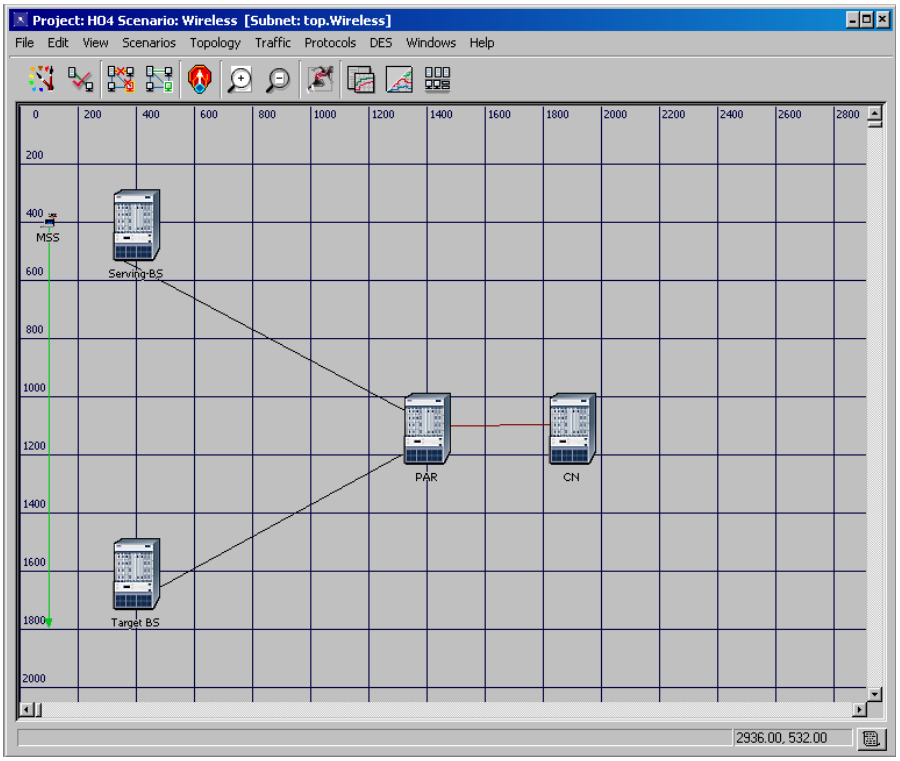


Fig. 3. The project model

3.2 The node models

Let us first explore several characteristics, which are common to all the node models. All node models where created using *device creator*. At each node model, Opnet's standard models which implement IP routing, CPU functionality, the TCP layer and the layers / modules below IP (ARP, MAC and PHY) weren't modified. All the node models, beside CN, which is a mere TCP server, contain an HO module, which controls the LPM HO procedure. The HO module is connected by packets stream to both Opnet's standard IP modules – ip_encap and ip_dispatch.

MSS and CN node models contain also a custom application layer module– which is called simply "*MSS*" and "*CN"*, respectively. These modules open a client–server TCP connection between CN and MSS, where CN creates data packets, and sends them (over TCP/IP) towards MSS.

In MSS, the MAC layer model contains only WLAN modules. In PAR and in CN, the MAC layer contains only Ethernet modules. In BS, which performs both wired and wireless communication, the MAC layer contains both WLAN and Ethernet MAC layer modules.

Figures 4-7 show the various node models. Specific documentation of every node model is available from within the node model editor by choosing *Node interfaces* from the *Interfaces* menu. [4] contains animations of the functionality of some of the node models.

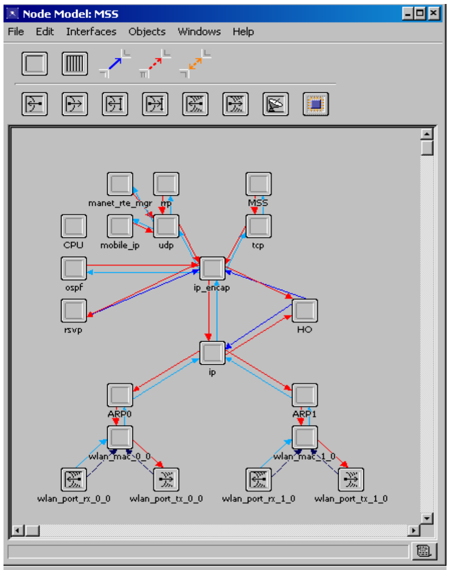


Fig. 4: MSS node model

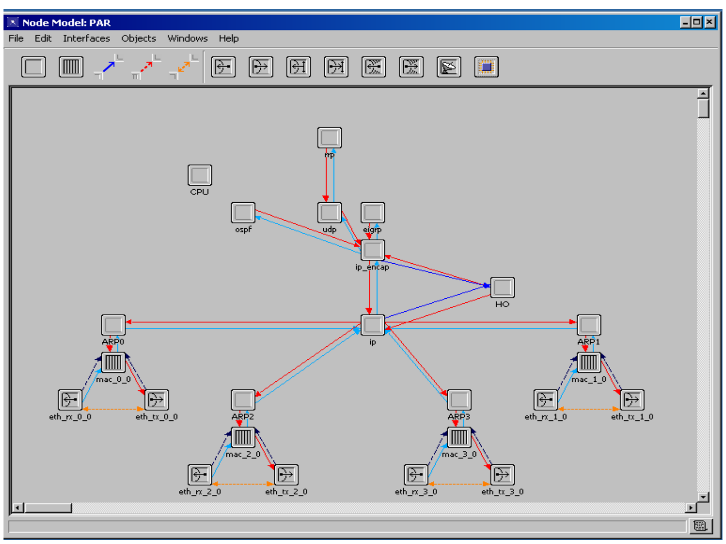


Fig. 5: PAR node model

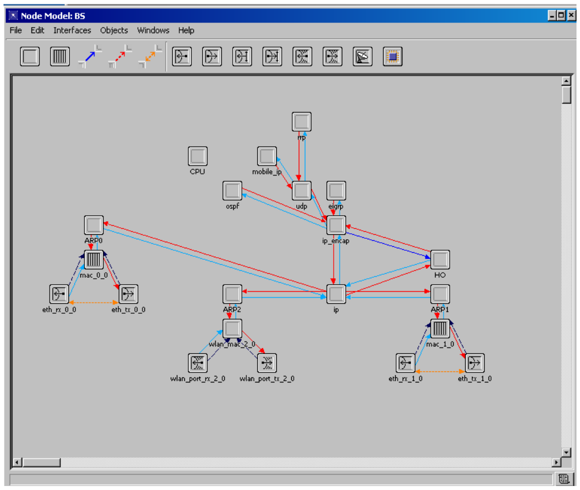


Fig. 6: BS node model

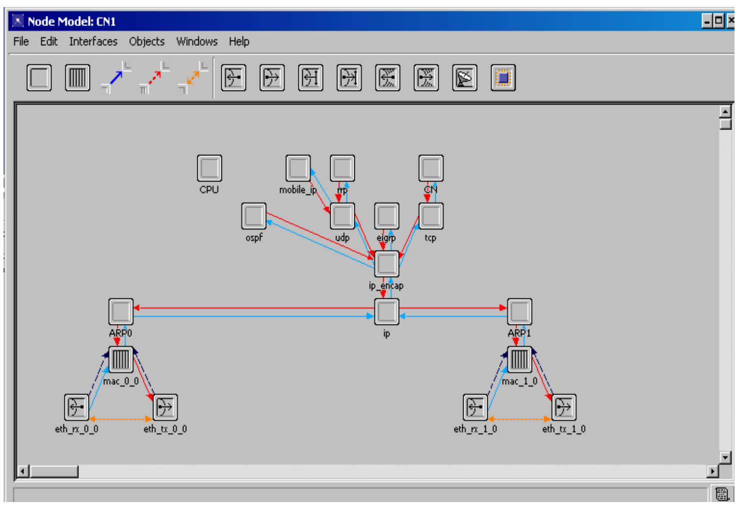


Fig. 7: BS node model

3.3 The process models

The process models within each node model are either custom or standard Opnet process models, sometimes with slight code modifications. The layers which contain modified / custom models are the IP layer and the application layers. In the remainder layers Opnet standard models were used with no change.

The IP layer

The HO models are named X_HO, where X stands for MSS, BS or PAR (no HO model in CN). There control the LPM HO procedure, including the various control messages and interrupts for initiating the various stages in the procedure.
The ip_dispatch module, a standard Opnet model which performs the lower-level IP operations, was slightly modified, so that instead of creating and invoking the child process model ip_rte_central_cpu (as in the original model), it actually invokes the process ip_rte_central_cpu_X, where X is stands for BS, PAR or MSS. These processes are responsible for the interception of some of LPM HO control messages in the midway node, and for the duplication / shifting of data packets.

In the TCP connection's end node – i.e. MSS and CN - ip_encap, which is a standard Opnet model for IP higher-level functionality and interfacing with the transport layer, was slightly modified. The modification is aimed to force the recognition of the duplicated and shifted packets as TCP segments, which should indeed be delivered upwards to the TCP layer upon arrival.

The application layer

The process model "*MSS*" is responsible for the establishment of the TCP connection with CN; upon arrival of data packets, it simply destroys them.

The process model "*CN*" is responsible for the establishment of the TCP connection from the server side. After the connection is open, it initiates data packets at a pre-defined rate (the rate is defined in the common header file "HO.h"), and delivers them to the TCP layer, for sending them towards MSS.

Detailed specific documentation of every process model is available from within the process model editor by choosing *Process interfaces* from the *Interfaces* menu. Further documentation is found in the C++ code files.

3.4 The packet formats

For each of HO LPM control messages, a custom packet format was created. As 802.16 model is not yet available in Opnet, the fields which carry data for the MAC layer communication were omitted.

An additional packet format is *pk2buf.pk*, which implements the encapsulation of a data packet, which is buffered by BS HO module.

3.5 The link models

The link models are based on slight modification of Opnet standard model for 10BaseT and 100BaseT link. The only modification was defining the links' delays as follows:

The wires (BS ←→ PAR): 5ms delay.

The wires (CN ←→ PAR): 50ms delay.

4. Simulation results

This section details the main resulting graphs of simulating LPM under a wide range of scenarios. Additional graphs are found at the accompanied presentation [4].

4.1 The basic scenario

The basic simulation model is shown in figure 3. PAR is connected to the BSs by a 10BaseT link with 5ms delay, and to the CN with 10BaseT link with 50ms delay. The BSs communicate with MSS over a 2Mb/s 802.11 connection. The link layer HO delay is set to 15 ms. An MSS starts the TCP connection with CN at time 3, starts to move at a constant speed at time 10, and finishes its trajectory close to the target BS at time 20. MSS initiates the preparation for the HO at time 14, and the real HO happens at time 14.6.

The TCP connection deploys Tahoe and New Reno algorithms [8]. The TCP window size is set to 20 and the packet size is 1024 Bytes. The application layer of CN initiates a data packet, destined to MSS, every 8ms, i.e. the optimal throughput (TP) at the steady-state is about (1/0.08)*1024*8 [bps] = 1Mbps.

LPM HO was found totally seamless under this scenario. Figure 8 shows the averaging over time of the traffic received by MSS. The HO at t=14.6 is almost transparent to the TCP layer. For a comparison, figure 9 shows the resulting graph when 802.16e HO scheme is deployed without LPM.

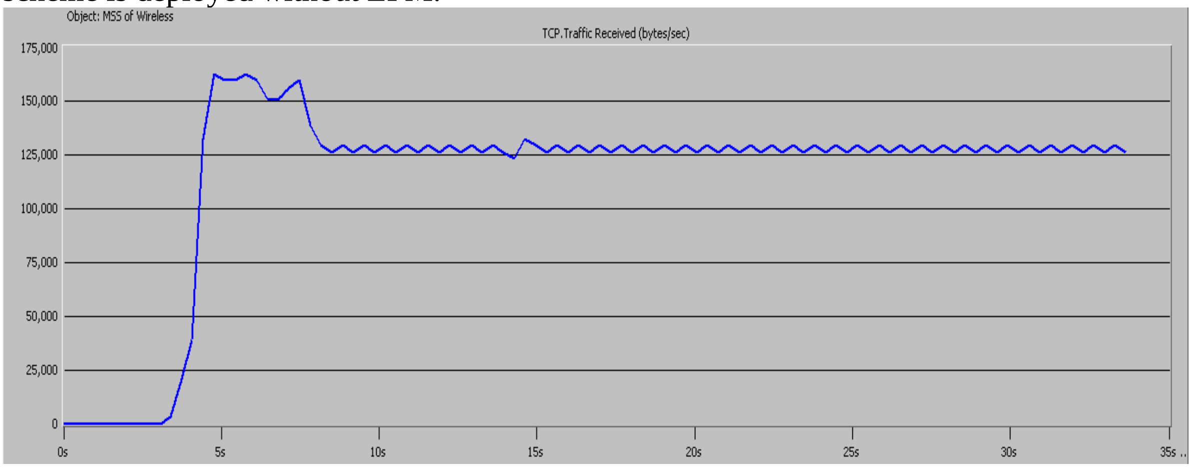


Fig 8: LPM HO: Time averaging of the traffic received by MSS in the basic scenario.

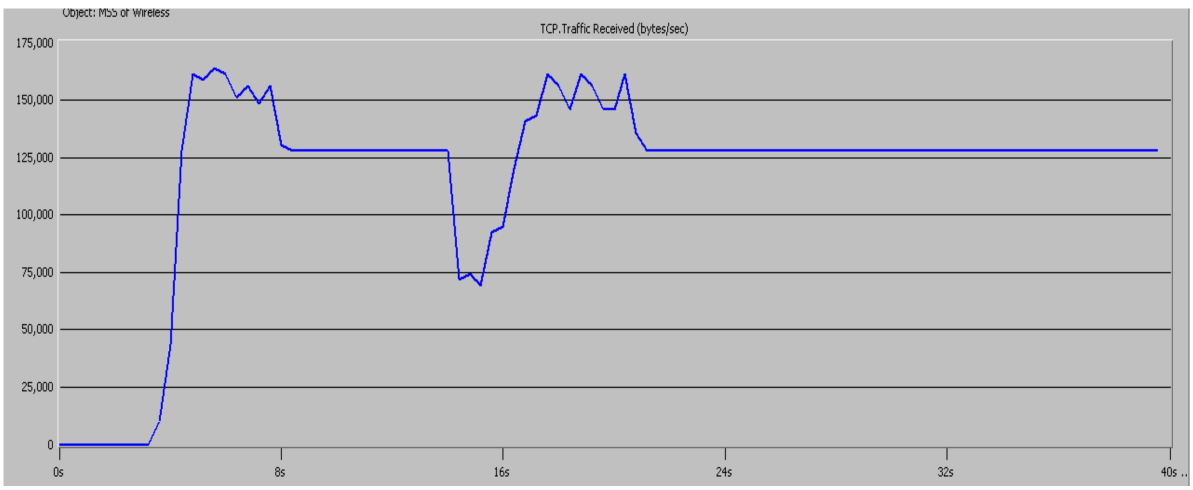


Fig 9: 802.16e HO without LPM: Time averaging of the traffic received by MSS in the basic scenario.

4.2 Scaling TCP parameters

In this section some TCP parameters were scaled, in aim to analyze their impact on the HO delay. Each time one attribute was scaled, while the rest remained as in the default configuration described in 4.1.

The parameters which were scaled are:

- TCP timer granularity, which determines the time resolution of all TCP's timers but the maximum ACK delay timer.
- TCP receiver buffer size.

Scaling of the timer granularity didn't affect the TP. Too small buffer size caused instability and degradation of the TP, but this phenomenon is general in TCP connections, i.e. is not a unique problem of this or another HO scenario or scheme. More graphs and details of these scenarios are given in [4].

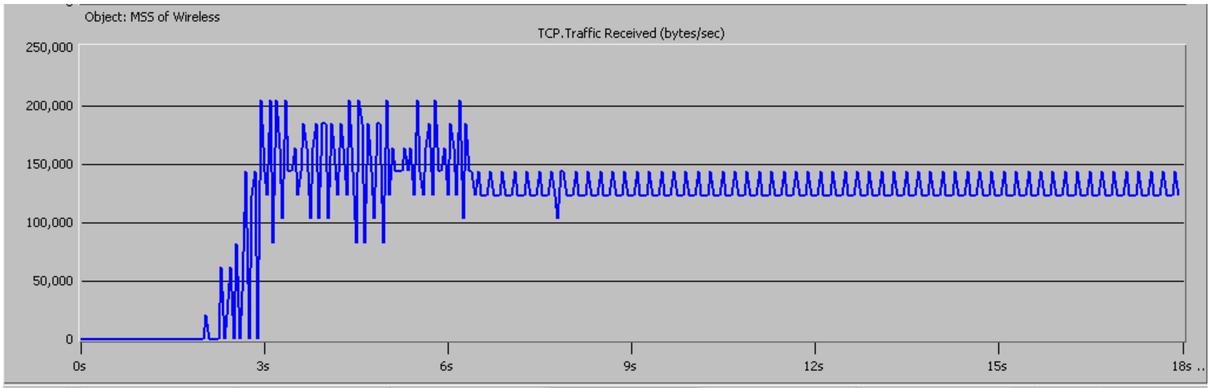
4.3 Slow application

When the application is slow, the TCP layer at the receiver's side may suffer from a lack of buffer space. This is true, of course, especially when the buffer is small.
In Opnet, the application's reading rate from the TCP buffer is represented by the scalable parameter *RCV_SEG_ratio*, which is the ratio (between 0 and 1) of the TCP receiver's buffer which should be full before the next time the application reads data from it. Figure 10 shows the TCP traffic received by MSS for three different values of RCV_SEG_ratio: 0, 0.25 and 0.5. The receiver's buffer size is set to 20KB, as in the basic default scenario. The figure demonstrates that LPM HO scheme does very well even when the application is very slow: after a slight decrease in TCP's TP during the real HO, it rapidly returns to the maximum TP.

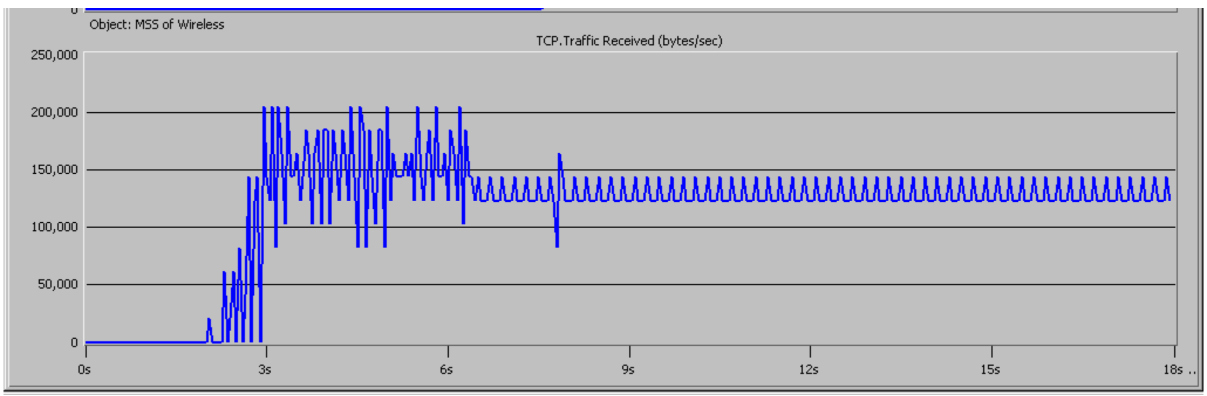
Fig 10: LPM: Time averaging of the traffic received by MSS for different values of RCV_SEG_ratio

4.4 When should the preparation for the HO start?

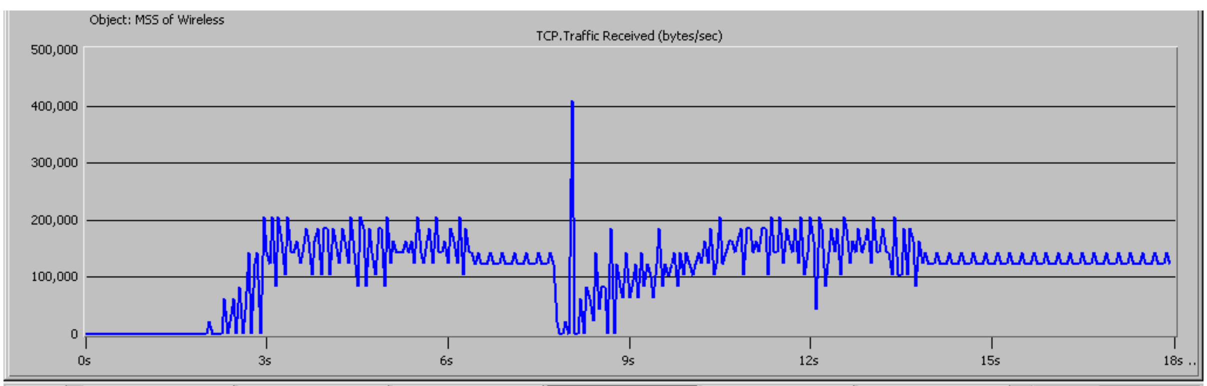
Too early preparation for the HO causes large overhead due to the duplicated packets. Thus, the preparation for the HO should start at the latest time which still guarantees a seamless HO.
Let us term the time interval between the preparation for the HO and the real HO the *preparation interval*. Simulations show that when the preparation interval is at least 0.35sec, the HO is totally seamless. When the preparation interval is between 0.3sec and 0.35sec, LPM HO still may do well at some probability. Below 0.30sec there is a very fast decrease at the TP, as shown in figure 11, which shows the TCP traffic received by MSS when the preparation interval is scaled.



(a)



(b)



(c)

Fig 11: LPM: Time averaging of the traffic received by MSS for different values of the preparation interval *pi*: (a): *pi* = 0.30sec. (b) *pi* = 0.29sec. (c) *pi* = 0.28sec.

Knowing the minimum preparation interval enables estimation of the overhead of the bi-casted packets. As the minimum preparation interval is 0.35sec, and the link layer HO delay is 0.15sec, the time interval during which the cross-over PAR duplicates data packets is about 0.5sec. At a data rate of 1Mbps, this means an additional load of at least 0.5Mb per each MSS which performs HO. A longer preparation interval or having a few crossover PARs (see section 2) may significantly increase this overhead.

4.5 High-Speed Trajectory

When MSS moves very fast, it's much harder to keep high TP during the HO. However, LPM keeps high TP even for a very high-speed trajectory.

Figure 12 shows the TP for the following scenario:

- t=1.50: MSS starts the TCP connection with CN.
- t=7.99: MSS starts the preparation for the HO.
- t=8.00: MSS starts moving towards the target BS.
- t=8.34 until 8.49: The real HO.
- t=9.00: MSS finishes its 1400m trajectory.

Figure 12 demonstrates that LPM HO is totally seamless either under this extreme scenario.

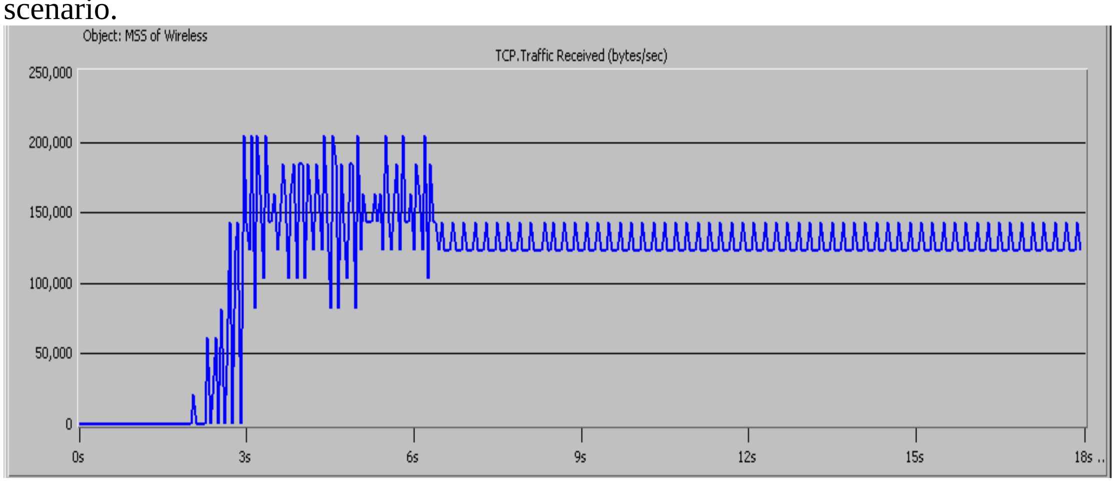


Fig 12: LPM: Time averaging of the traffic received by MSS at a high-speed trajectory. The real HO happens between 8.34 and 8.49.

One should note, that there's no use to simulate faster movements, because it doesn't make sense to simulate trajectory shorter than 1sec when the preparation for the HO takes at least 0.5sec.

4.6 HO at higher data rate

802.16 is aimed for a much higher data rates than 802.11. Thus, it makes sense to check out whether the excellent results found in the previously described sections remain so either at a much higher data rate.

At this section, LPM HO was simulated at the highest data rate available for 802.11 - 11Mbps (reminder: 802.11 is our approximating platform, as 802.16e is not available yet on Opnet).

In this scenario, some more parameters were scaled (relatively to the default configuration in 4.1) as follows:

- The 10BaseT wires were replaced by **100BaseT** wires with identical delays.
- MSS’s TCP receiver buffer size was raised to **128KB**
- The rate of the data initiated by CN application was raised to **8Mbps**

Figure 13 demonstrates that LPM HO is totally seamless either under this scenario.

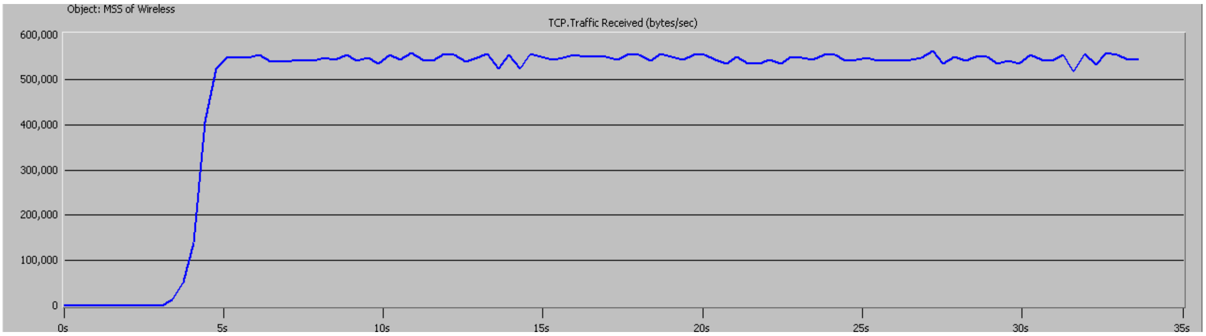


Fig 13: LPM: Time averaging of the traffic received by MSS under a high data rate scenario. The real HO is at time 14.6 until 14.75.

5. Conclusions

5.1 Review of LPM scheme

The results of the simulations presented at this paper are very clear: LPM is indeed a very promising scheme, which may offer a significant decrease in the HO delay, thus enabling an almost-seamless HO even under extreme scenarios.
However, one should recall that A) the simulations are not accurate enough yet (see section 5.2), and B) deploying LPM requires high overhead (see sections 2.2 and 5.3).

5.2 Future work

The current model uses Opnet modeling of 802.11 Technology for approximating the behavior of the 802.16. Once Opnet supplies a standard model for 802.16, one may upgrade the existing PHY and MAC layers modeling to 802.16 technology, thus providing more accurate results, and modeling which better considers the real-world implementation issues. One may also re-write the existing model in a more general way, so that each BS will dynamically determine its status as either a serving BS or a target BS (or an idle BS), and LPM HO control messages will be sent by a real need (i.e. realization of a drop in the received signal strength) rather than in pre-defined times. Such a modification would make the model more elegant and scalable. It would also be important and interesting to use such a modeling for simulating a closer-to-reality scenario, where each BS serves many MSSs, from which a few may leave the BS and a few others join it - everything simultaneously.

References

[1] IEEE Standard 802.16, IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems (2001)

[2] 802.16E-2005 IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands.

[3] A Seamless Handover Mechanism for IEEE 802.16e Broadband Wireless Access, Kyung-ah Kim, Chong-Kwon Kim, Tongsok Kim. Lecture Notes in Computer Science, Volume 3515, Jan 2005, pp. 527 – 534

[4] Opnet Simulations of Handover Mechanisms for IEEE 802.16, Itamar Cohen, presentation slides, revision 3, Mar-06.

[4] The IEEE 802.16 Working Group on Broadband Wireless. [Online]. Available: https://www.ieee802.org/16/

[5] The IEEE 802.16 WirelessMan Standard for Broadband Metropolitan Area Networks, Roger B. Marks, IEEE computer society distinguished visitors program, 9-Apr-03 (presentation slides).

[6] The WiMAX Forum [Online]. Available: http://www.wimaxforum.org/home/

[7] Opnet Modeler [Online]. Available: https://support.riverbed.com/content/support/software/opnet-model/modeler.html