

High Mobility in a Realistic Wireless Environment: a Mobile IP Handoff Model for NS-2

Claudio E. Palazzi^(1,2), Brian Chin⁽¹⁾, Paul Ray⁽¹⁾, Giovanni Pau⁽¹⁾, Mario Gerla⁽¹⁾, Marco Rocchetti⁽²⁾

¹ Computer Science Department, University of California, Los Angeles
Boelter Hall, Los Angeles CA, 90095, USA
{naerbnc | paulray | gpau | gerla}@cs.ucla.edu

² Dipartimento di Scienze dell'Informazione, Università di Bologna
Mura Anteo Zamboni 7, 40127 Bologna, Italia
{cpalazzi | roccetti}@cs.unibo.it

Abstract—Wireless technology has the great merit to allow users' mobility. However, this may affect the functioning of all the applications and traditional protocols, worsening the already known problems with wireless connections and requiring the design and evaluation of novel solutions. Scenarios involving high mobility can be very complex; think, for instance, to a realistic case involving several nodes, representing people and cars moving under the coverage of several base stations. Since the geographical width and the number of involved nodes in such a scenario, simulations represent a fundamental tool to analyze the state of the art and newly proposed solutions. Unfortunately, modeling mobility with the most popular network simulator in the scientific community (i.e., NS-2) is not an easy task as its handoff mechanism appears faulty when coupled with a realistic model for wireless transmissions. In this paper, we discuss this problem and propose a new simulation module for Mobile IP able to guarantee smooth handoffs even with realistic simulation set-ups.

Keywords- NS-2; Simulation Techniques, Wireless; Handoff; Mobile IP

I. INTRODUCTION

People perceive the necessity to be always connected, even when traveling, as more and more urgent. To this aim, airways companies are now offering to costumers the possibility to be connected even when flying and vehicular networks are emerging as a “hot” research topic in the field of wireless communications.

However, before highly mobile connectivity can become a reliable and efficient everyday reality, several challenges must be overcome. First and foremost, this scenario exacerbates well known problems in traditional wireless scenarios. Packet errors can increase so much to lead to very low performance of both TCP-based applications and applications that can generally tolerate some packet loss with minimal perceived impact (i.e. UDP-based real time applications such as VoIP and online games [10], [2]).

The connectivity can also become intermittent or completely disappear if a node moves to an area insufficiently

covered by base stations (BSs). Considering the example of a vehicular network, Fig. 1 shows how vehicles may be forced to resort to alternative communication methods such as cellular coverage, ad hoc connectivity (cars V1 and V2 in Fig. 1), and multihop (cars V3, V4, and V5 in Fig. 1) links [3]. Node mobility can lead to frequent changes in network topology and network membership, thus requiring mechanisms able to guarantee an efficient and seamless maintenance of the ongoing session.

As it is evident, high mobility scenario requires extensive study to analyze problems and possible solutions. Any new scheme has to pass through an evaluation of its efficiency and feasibility which may involve even hundreds of nodes moving within the same (wide) area in a small interval of time. Since the complexity of this scenario, real experiments are often unfeasible, especially at the first testing stage. Simulation embodies a fundamental tool for this complex topic.

NS-2 represents the most popular network simulator utilized by researchers [5]. Over the years, this event driven simulator has been adopted as the evaluation tool in a multitude of scientific papers regarding different network layers and a wide scope of issues. Its prominent virtues are its open source nature, the reliability of its outcomes, and the possibility to be used both in serial or in parallel/distributed way to speed up simulations [7].

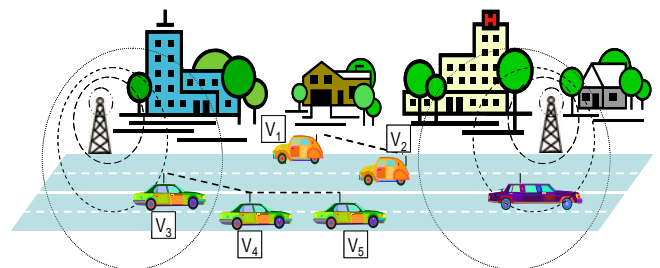


Figure 1. Example of high mobility scenario: vehicular networks.

However, NS-2 currently lacks a very important functionality in order to be a proficient tool in the difficult problem domain of high mobility. This is due to the fact that the Mobile IP handoff mechanism fails when coupled with a particular wireless transmission model: the *shadowing model*. Unfortunately, this model is the only realistic one among those available in the officially released version of NS-2.

More in general, the problem is that the current NS-2's Mobile IP model would fail with every reasonable probabilistic model for wireless transmissions. Consequently, mobile scenarios in infrastructure mode involving handoffs cannot be currently simulated with NS-2 as the results would be biased either by the adoption of a unrealistic wireless model or by the incompatibility of the current Mobile IP model with probabilistic wireless transmission ones.

In this paper we analyze the handoff model currently available for NS-2 to demonstrate its inadequacy to serve the scientific community in simulating high mobility scenarios. We then propose a revised Mobile IP module that is able to guarantee a smooth and correct handoff even when employing a probabilistic model for wireless transmission. We also show a practical example of its utilization demonstrating how our contributed module can be exploited, for instance, to perform analysis of an infrastructured vehicular network.

The rest of the paper is organized as follows. In Section II, we discuss the Mobile IP protocol. Section III specifies current issues in modeling Mobile IP in NS-2. We describe our solution in Section IV. Section V provides details on the simulative environment. Experimental results are displayed in Sections VI. Finally, Section VII concludes the paper.

II. HANDOFF WITH THE MOBILE IP PROTOCOL

Mobile IP is a standard communications protocol designed by the Internet Engineering Task Force (IETF) [4]. It allows a *mobile host* (MH) to maintain the communication with another wired or wireless host, called *remote host* (RH), even when moving and passing by different BSs belonging to diverse WLANs.

As shown by Fig. 2, the MH can move from the coverage area of its *home agent* (HA), which is the BS where the MH is initially registered, to the coverage areas of other BSs which represent *foreign agents* (FAs). By *tunneling* through these FAs, alternative routes between the MH and the RH could be established.

When a MH decides to switch association between two agents, data traveling between that MH and its RH will pass through the FA. In particular, packets traveling from the RH to the MH will continue to route through the wired domain to the HA and, once there, they will be forwarded to the FA in order to reach the MH. To preserve packet information, data must first be encapsulated by the HA and then decapsulated by the FA; then, they can be forwarded to the MH over the wireless link. Packets from the MH to the RH, instead, will not need to route through the HA: they are simply sent toward the FA where they will be directly forwarded to the RH through the wired domain.

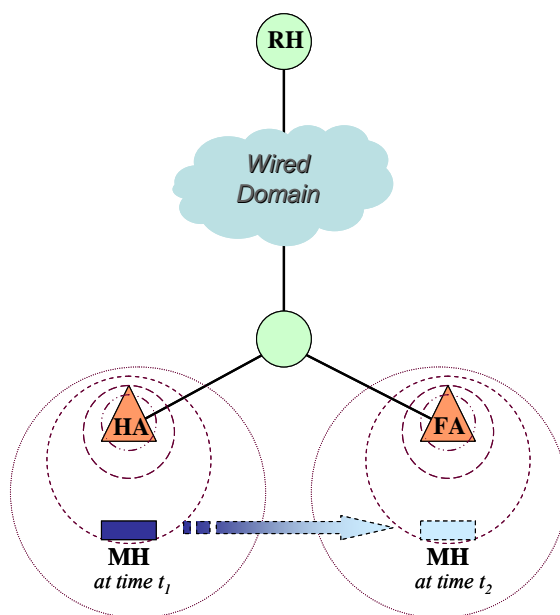


Figure 2. Mobile IP generalization.

As a MH travels from one location to another, as an automobile would in a vehicular network, it may have to frequently switch associations with various BSs due to communication range restrictions. In order for a MH to detect the presence of BSs that it can communicate with, all BSs must periodically broadcast a beacon containing their *care-of-address*. If a MH hears a beacon and decides to associate with the corresponding BS rather than keep its current association, it sends a registration request to that BS. If that BS is not the HA of the MH (it hence is a FA), then it sets up the decapsulator and forwards the registration request to the HA. Acknowledging that the care-of-address of the request is not its own, the HA sets up the encapsulator and sends a registration reply.

The decision of when a MH should switch its association is quite complex. The RFC standard suggests two methods. The first is to register with a new agent once the lifetime of the current connection expires. An issue with this algorithm is that since the time spent at a given FA can vary, no general lifetime is appropriate. Think, for instance, to the vehicular network scenario; in this case, in fact, cars travel fast thus staying in the coverage area of a BS for a very limited time and requiring handoffs with a frequency that is much higher than the lifetime expiration rate.

The second method is to switch to any agent outside of the subnet when its beacon is heard, because this would suggest that the mobile host has moved. In the realistic case of two subnets overlapping, however, this could result in repeated switching. Moreover, there is no guarantee that a new FA, which has just been heard, will be a better choice than the current one. In fact, it might be the case when the MH stops moving immediately after having heard the new BS. In this case, the MH could have a better connectivity with the old agent, rather than with the new one since the latter is still too far from it.

III. PROBLEM STATEMENT: THE INADEQUACY OF THE CURRENT NS-2'S MODULE FOR MOBILE IP

NS-2 combines the two aforementioned strategies, as can be noticed by delving into the module named *mip-reg* [5]. Here, each MH keeps a list of possible FAs from which it has heard a beacon. Each element of the list expires after one beacon period (one second by default). Therefore, each time the MH does not receive a scheduled beacon from a FA, the reference to that FA is eliminated from the list. Instead, every time the MH hears an advertisement from an agent that is not in the list, the MH assumes that it has moved; thereby it inserts the new agent in its list and associates with it (the new FA becomes the care-of-address for the MH). In addition, if the current care-of-address expires, the MH associates with the FA that advertised most recently.

This algorithm would work well in simulations utilizing wireless radio models where all packets are successfully transmitted within the ideal circle representing the fixed communication range of a BS (i.e., the *free space model* and the *two-ray ground reflection model*). On the other hand, these models are very distant from the physical laws in the real world and are thus not appropriate to perform meaningful evaluations [6].

Unfortunately, the aforementioned NS-2's algorithm for Mobile IP fails when coupled with more true to life models that utilize probabilistic schemes to determine the success of a transmission (i.e., the *shadowing model*). With these probabilistic models the received power fades with the distance and can be modeled more by a random variable that takes into account both the distance and the degree of obstruction present between the BS and the MH. As a result, simulation will portray irregular connectivity while the MH moves out of the reliable communication range of its engaged BS.

With the help of a practical example depicted in Fig. 3 we can illustrate why the current NS-2's interpretation of Mobile IP does not work when coupled with a probabilistic transmission radio model. To this aim, consider two possible BSs for a mobile host, namely A and B. Assume now that A is very close to the MH and without any serious obstacles in-between, while B is characterized by poor connectivity with the MH. In this scenario, the advertisements from A are always received by the MH while the advertisements from B are randomly dropped with a significant probability.

Without loss of generality, we can assume that A sends its beacons on the second (i.e., 0.0, 1.0, 2.0, etc.), while B sends its beacons on the half-second (i.e., 0.5, 1.5, 2.5, etc.). The node gets an ad from A at 0.0, and thus sets its care-of-address to point to A.

Since the MH receives advertisements from A every beacon period, A will persistently stay in the MH's list without expiring and, thereby, without re-entering. As previously said, the NS-2 module allows only new BSs (i.e. BSs not already present in the MH's list of FA) to become the new care-of-address when heard by the MH; therefore, the MH will never set its care-of-address to A again. The next time the MH gets an advertisement from B, while it is not in the list of FAs, B will become the new FA for the MH and it will stay like this.

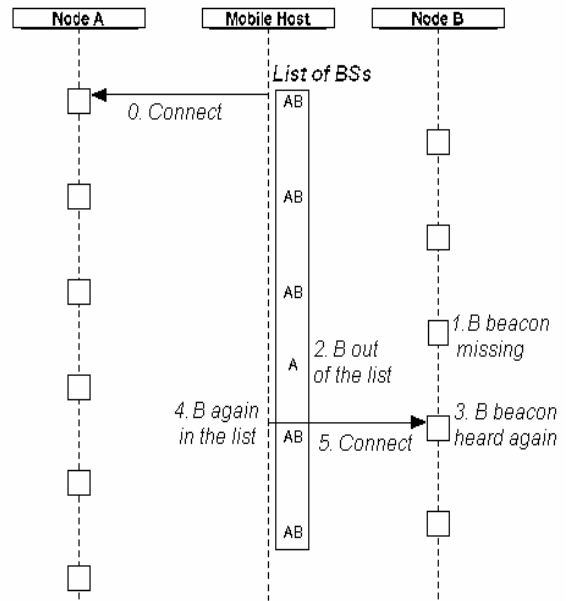


Figure 3. Current handoff mechanism in NS2.

A similar problem happens if the MH connects first to B (i.e., the BS featured with the worst connectivity to the MH). First, the MH receives an advertisement from B thus setting B as the new FA. Then, MH receives an advertisement from A and sets A as its new FA. Since from now on the MH will receive advertisements from A every beacon period, A will persistently stay in the MH's list of FAs. Therefore, A can only be set once as the MH's FA.

Instead, B is still in the list now, but since the channel connecting it to the MH is lossy, there will be some point where an advertisement from B will be lost. Consequently B will be dropped from the list of possible FAs maintained by the MH. However, as soon as an advertisement from B will get through the lossy channel reaching the MH, B will be re-entered in the list of possible FAs as the new care-of-address. As before, the MH will stay persistently linked to B from that time on.

Summarizing, given a BS next to the MH (with a possible connectivity that would be practically lossless), and a BS at a distance (with a possible connectivity that would be lossy), the MH will always pick the BS at a distance.

IV. A CORRECT MOBILE IP MODULE

The problem explained in Section III is due to the fact that the MH utilizes only a single piece of information (i.e., the hearing of one advertisement from a new BS) to perform a handoff. Obviously, a more stable behavior could be obtained when considering more information. This would be not only desirable, but also more coherent with any realistic implementation of the protocol.

Ideally, in fact, a MH would maintain its association with its current agent until another agent was notably better. To compare agents, the mobile host would need to be aware of its current connection strength, or loss, with known agents. For this, we propose to store a little extra information about each

FA in the MH's list and to have the MH keeping a running average of successfully received advertisements.

To calculate this running average, we must first define a constant α and a binary variable *Beacon_Detected*. The value for α is defined as a number from 0 to 1 and is used to weight the agent's history against its present every beacon period. When the MH checks whether it has detected an advertisement from an agent in its list, it sets *Beacon_Detected* to 1 if it has, 0 otherwise. Using this, it calculates the new running average for that agent is defined as:

$$NewAvg = \alpha \times OldAvg + (1 - \alpha) \times beacon_detected \quad (1)$$

Once running averages have been updated for all agents, an association switch occurs only when another agent's average is better than the current agent's one plus some threshold value T (introduced to provide hysteresis). Finally, a FA is removed from the MH's list if some time period N seconds has elapsed without a detected advertisement.

This is one of the possible interpretations of the switching mechanism between two agents when Mobile IP is utilized. It is indeed a very simple mechanism that, as we demonstrate in Section VI, perfectly serves our purposes. Obviously, nothing impedes other researchers to propose and test better handoff schemes. In fact, our aim is to demonstrate how the current NS-2's module for Mobile IP does not work with a realistic wireless transmission model and to contribute with a properly working module.

Thanks to our modified *mip-reg* module, which is available online at [7], researchers can investigate scenarios involving also Mobile IP. Moreover, by comparing the new module with the old one, it becomes easier to understand how *mip-reg* can be further modified to embody alternative handoff mechanisms.

V. SIMULATION ASSESS

The effects of the problem depicted in Section III and the improvement obtained by utilizing our Mobile IP module can be easily seen via NS-2 simulations. We have employed version ns-2.28 but our outcomes can be extended to all the NS-2 versions. We wrote and run a TCL script to simulate the simple topology depicted in Fig. 4.

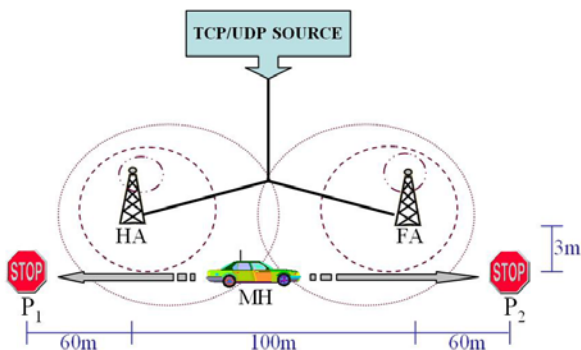


Figure 4. Simulated topology.

TABLE I. MH'S MOVEMENTS

Time	Description
0 s	Simulation starts, MH is in HA's wireless range
25 s	MH starts downloading a file from RH
30 s	MH moves toward FA at 53 km/h (33 mph)
41 s	MH exits from HA's range
45 s	MH reaches its final destination in FA's area
50 s	MH moves back toward HA at 88 km/h (55 mph)
56 s	MH exits from FA's range
59 s	MH reaches its final destination in HA's area
70 s	Simulation ends

We have adopted the shadowing model to determine the radio propagation in our simulations. In fact, this model represents the most realistic one among those available in the official version of NS-2. Moreover, since the aim of this work is to discuss the Mobile IP implementation over a general probabilistic radio model, we do not need "the ideal" radio model. We just need a proof-of-concept probabilistic model to represent communication between the BS and the MH. The shadowing model perfectly serves to this aim.

The shadowing model requires the adjustment of some parameters to be used. Having in mind a vehicular network scenario we have followed the official documentation of NS-2 to represent the outdoor of a general shadow urban area [8]. Specifically, the *loss exponent* and the *shadowing deviation* were set equal to 2.7 and 4, respectively.

The total simulation time is 70 seconds. The MAC layer parameters at the BSs have been modified to realistically embody the IEEE802.11g standard. The simulation outcomes showed us that we were able to reach a maximum achievable bandwidth of around 20 Mbps. This represents a reasonable value over the declared 54 Mbps even in the real world [9].

The topology has been configured in such a way that the wireless link represents the bottleneck of the connection. Buffers on the BSs were set with a size of 50 packets as this is one of the most common configurations for BSs found in commerce.

In this scenario, a MH is initially in the range of its engaged HA and from that position, at 25 s, it starts an application involving also the RH, with a minimum RTT of 80 ms. For the sake of completeness, in our simulations we have considered both a TCP-based and a UDP-based application: a File Transfer Protocol (FTP) and a Constant Bit Rate (CBR), respectively. In both cases, we have compared the behaviors of

the regular NS-2's Mobile IP module and the Mobile IP module we propose. In essence, with a proper functioning of the Mobile IP module, we expect a timely packet re-direction when the MH moves out of the range of its HA and enters in the range of the FA. If this happens and the two areas of coverage overlaps (as in our scenario) very few packets will be dropped. Conversely, having many packet losses would represent the proof of the malfunctioning of the handoff algorithm.

In our simulations, we have first run a FTP application where the MH was downloading a big file from the RH utilizing the regular TCP New Reno as transmission protocol and a maximum packet size of 1500 bytes. In this case, two main reasons suggested us to utilize a chart depicting the congestion window's trend as a metric to evaluate the Mobile IP modules' behavior. First, the congestion window's shape provide us with information about the number of packet losses: every reduction of the congestion window corresponds to at least one packet loss. Second, the congestion window size is directly proportional to the throughput of the connection, thereby, from the height of its values we have a visual measure of the bias introduced by the utilized Mobile IP scheme on TCP's performance.

We have then evaluated a UDP-based CBR flow where the RH was sending every 50 ms a 200 byte packet to the MH. In this case, to compare the two Mobile IP modules, we simply consider the number of packets that reach the destination.

In our scenario, the MH moves back and forth between the two coverage areas with velocities that are in the order of typical car speeds. Details for the movements of the MH are listed in Table I. This is a very simple and general scenario where a car moves passing by the coverage area of different BSs. This configuration perfectly serves the purpose of this work which is to demonstrate how our module allows performing simulations that involve handoffs over a realistic wireless channel, whereas the default NS-2's module does not. However, we have also run simulations where cars were moving in a linear direction (say in a highway) and passing a number of overlapping BSs. Outcomes showed no noticeable difference with the previous case; we hence to omit to report on these results as they would be redundant.

Finally, we want to point out that we have run tens of simulations for the considered scenario, changing the random seed of the simulator, positions of P1, P2, HA, and FA in Fig. 4, and moving times. Changes were made so as to have P1 always under the HA's area of coverage, P2 always under the FA's area of coverage, and the speed always set to a reasonable value for a car. As expected, there was no remarkable among the various cases; we hence present results only for a single but representative case that allows a deeper comprehension of the involved dynamics.

VI. RESULTS AND DISCUSSION

In the depicted scenario and with a proper implementation of the Mobile IP protocol, one would ideally expect the MH switching associations from the HA to the FA, and then back to the HA. The TCP's congestion window (*cwnd*) in Fig. 5, however, shows a very poor connection, demonstrating that the

handoff was not as smooth as expected. The slow start threshold (*ssth*) and the bandwidth-RTT product (*RTTxBW*) are shown as well for a better comprehension of the outcome.

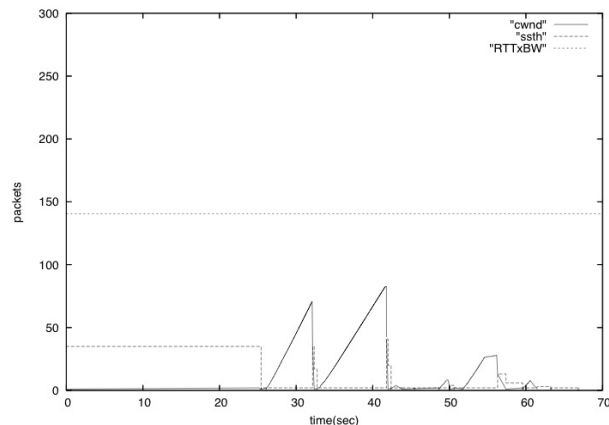


Figure 5. TCP's congestion window; moving MH; regular NS-2 Mobile IP module utilized.

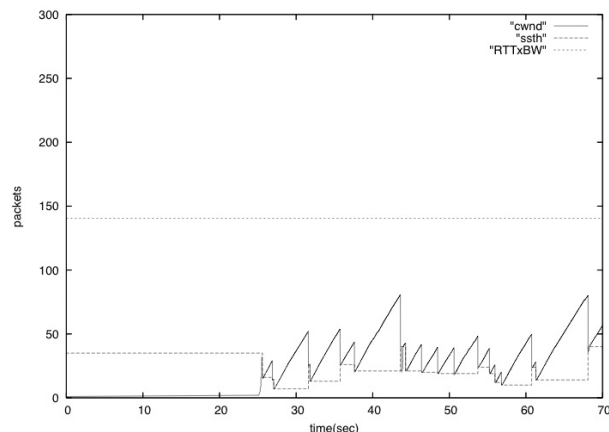


Figure 6. TCP's congestion window; static MH; regular NS-2 Mobile IP module utilized.

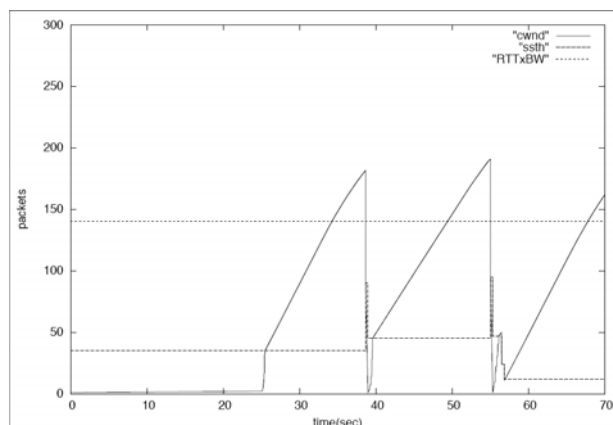


Figure 7. TCP's congestion window; moving MH; Running Average Advertisement for Mobile IP utilized.

Even in the case where we leave the MH static near the HA, we still experience poor transmission performance. Since the two BSs are located in proximity of each other, in fact, even when the MH is close to the HA it can still sporadically receive some advertisement from the FA. Consequently, the MH keeps changing its care-of-address between the two BSs, losing packets and affecting TCP sending rate.

In Fig. 6, we show the congestion window trend for this case. The outcome is definitely better than that of Fig. 5, however, such an irregular saw-tooth shape would be expected from a relatively lossy link. Conversely, for the whole simulation time, the MH had an optimal connectivity option with the HA.

By looking at the trace file produced by the simulation we had the confirmation of our supposition. The reason for this unexpected outcome, in fact, is that the MH is located in the reliable broadcast range of the HA but, at the same time, also in the lossy broadcast range of the FA. While the MH received all advertisement packets from the HA, maintaining the HA persistently in the list of possible agents for the MH, occasionally advertisements from the FA were lost. These losses caused the FA to expire (if present) from the MH's agent list. When the MH received an advertisement packet from FA again, it associated with it. In this way, NS-2 compels mobile nodes to engage lossy BSs.

This performance degrading behavior does not represent the outcome that a proper implementation of Mobile IP would produce in the real world; it rather represents the consequence of an unfortunate interpretation of that protocol's RFC [4]. Therefore, we need to provide a correct but wiser rule for making MHs switch between BSs.

To this aim, we modified the NS-2 module as explained in Section IV and we have replicated the exact setup of the previous simulation but using this time our Mobile IP module. In particular, the experiment was configured with beacon periods at 0.1 seconds, $\alpha = 0.9$, $T = 0.1$, and $N = 120$.

Obtained results are shown in Fig. 7. As desired, the handoffs become far more obvious and the congestion windows are able to grow large without repeated switching between BSs. A small amount of packet loss can be noticed at the 55 second mark. This is because the channel was still lossy, but not so that progress could not be made. As the MH moves further into range of the BS (i.e., the HA), the connection progresses well.

A clear advantage of using the running average advertisement method is that the MH will tend to continue communicating with an agent until an obviously better alternative appears. Indeed, hysteresis will avoid rapid association switches that would consistently degrade the performances with no connectivity advantages.

These results involving a TCP-based FTP are confirmed by those where we simulated a UDP-based CBR. Indeed, Fig. 8 shows the number of packets received by the MH when utilizing the two Mobile IP algorithms: the regular NS-2 (*reg-NS2* in Fig. 8) one and the one we propose (*proposed* in Fig. 8).

In both cases 900 packets were sent in 45 s by the RH to the MH. However, with the proposed modifications to the Mobile IP module almost all sent packets received the destination, whereas with the regular NS-2's module many packets were sent to the wrong BS while the MH was moving. This caused many packet losses which are clearly visible as plateau phases in the growth of the *reg-NS2* line in Fig. 8.

As a confirmation, Fig. 9 shows the percentage of packet loss experienced by the UDP flow with the two possible Mobile IP modules. As it is evident, when our Mobile IP module is employed, losses represent only a reasonably little percentage of the total amount of UDP packets sent; whereas almost one fifth of packets are lost with the regular module due to its wrong interpretation of the handoff procedure.

To sum up, we have created a Mobile IP version that features MHs with a more stable (and realistic) handoff behavior. Handoffs can now be part of the simulated scenario when evaluating new techniques and protocols without biasing the outcomes.

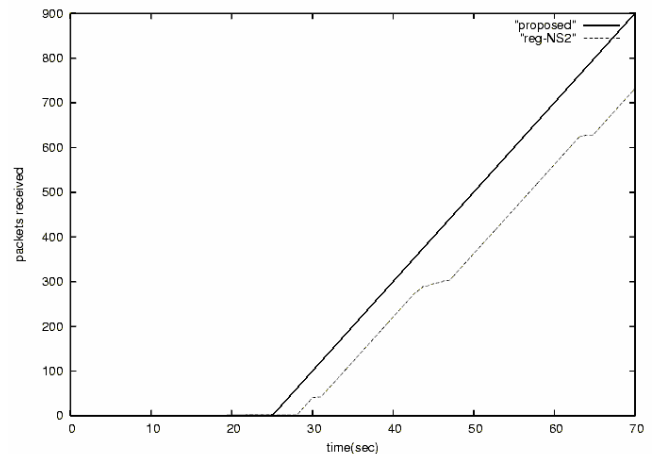


Figure 8. Received packets (UDP); moving MH; comparison between Mobile IP algorithms.

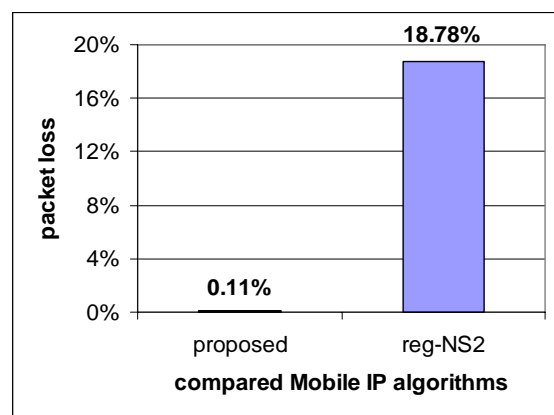


Figure 9. Lost packets (UDP); moving MH; comparison between Mobile IP algorithms.

VII. CONCLUSION

As discussed in this paper, high mobility poses several interesting problems that needs a deep analysis. A realistic scenario can be very complex and involve many entities. Therefore, simulations, both serial and distributed/parallel, represent a fundamental tool for the scientific community.

We focused on typical features of high mobility scenario, fast mobility and continuously changing network membership, and demonstrated through experimentation the inadequacy of NS-2 to support this kind of investigation. In particular, we exposed flaws of NS-2's current method of handling Mobile IP when coupled with a probabilistic model for wireless transmission.

Through our running average advertisement algorithm we have provided an efficient method for deciding when and to which agent a MH should switch. Differently from the default implementation in NS-2, our module works independently from the wireless model. Researchers can hence adopt their preferred realistic wireless models and couple them with our handoff module to generate complex and yet reliable mobility simulations.

Moreover, by taking advantage of our work and of the comments we introduced in the code to simplify its comprehension, researchers can more easily implement their ideas on how to improve handoffs and simulate them with NS-2.

Finally, we want to point out that our Mobile IP algorithm is fully included in the *mip-reg* module. Basically, a user has only to substitute the regular *mip-reg* module with ours and recompile the simulator's source. Therefore, our module is fully compatible with the regular NS-2 release both when employed in the standard serial way and when used in a parallel/distributed fashion [7].

Our module is currently online [11], available to the scientific community.

ACKNOWLEDGMENT

This work is partially supported by the Italian Ministry for Research (M.I.U.R.) via the Interlink Initiative, the ICTP/E Grid Initiative, and the Damasco Initiative, by the National Science Foundation through grants CNS-0435515/ANI-0221528, and by the UC-Micro Grant Micro 04-05 private sponsor STMicroelectronics.

REFERENCES

- [1] Consumer Electronics - IT Facts. <http://www.itfacts.biz/index.php?id=P1861>
- [2] T. Beigbeder, R. Coughlan, C. Lusher, J. Plunkett, E. Agu, and M. Claypool, "The Effects of Loss and Latency on User Performance in Unreal Tournament 2003", in Proc. of ACM Network and System Support for Games Workshop (NetGames), Portland, OG, USA, Sep 2004.
- [3] A. Nandan, S. Das, G. Pau, M.Y. Sanadidi, and M. Gerla "Cooperative Downloading in Vehicular Ad Hoc Wireless Networks", in Proc. of IEEE/IFIP International Conference on Wireless On demand Network Systems and Services, St. Moritz, Switzerland, Jan 2005.
- [4] C. Perkins, "IP Mobility Support for IPv4", IETF RFC 3344, Aug 2002.
- [5] The Network Simulator - NS-2. <http://www.isi.edu/nsnam/ns/>
- [6] T. S. Rappaport, Wireless communications, principles and practice, Prentice Hall, 1996.
- [7] PDNS - Parallel/Distributed NS. <http://www-static.cc.gatech.edu/computing/compass/pdns/>
- [8] The Manual (formerly Notes and Documentation). <http://www.isi.edu/nsnam/ns/doc/index.html>
- [9] A. L. Wijesinha, Y. Song, M. Krishnan, V. Mathur, J. Ahn, and V. Shyamasundar, "Throughput Measurement for UDP Traffic in an IEEE 802.11g WLAN", in Proc. of 6th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN'05), Towson, MD, USA, May 2005.
- [10] L. Sun, G. Wade, B. Lines, and E. C. Ifeachor, "Impact of Packet Loss Location on Perceived Speech Quality," in Proc. of the 2nd IP Telephony Workshop, Columbia University, New York, Apr 2001.
- [11] Mobile IP for NS-2. <http://www.cs.unibo.it/~cpalazzi/mobileip.html>