Study of Best-Effort VoIP Handovers between WLAN and EVDO Networks

Amit P. Jardosh[†] Rajeev Koodli[§] Tat Chan
[†]Dept. of Computer Science, UCSB [§]Nokia Research Center
amitj@cs.ucsb.edu,rajeev.koodli@nokia.com

***Abstract-* The IEEE 802.11 based Wireless LANs (WLANs) have emerged as a viable technology for supporting real-time applications such as Voice over IP (VoIP). Even the Personal Digital Assistants and Smartphones are being equipped with WLAN, leading such devices to operate in *dual-mode* with the WLAN and WAN (Wide Area Network) radios capable of supporting IP communication. This development enables applications such as VoIP to roam or *handover* freely across the two networks. Seamless handover of VoIP between such networks poses two challenges: the call must persist in spite of mobility across networks, and the LAN and WAN must support the delay and packet loss requirement bounds for VoIP calls. In this paper, we present an experimental study of best-effort VoIP handovers across WLAN and the CDMA EVDO networks, using *Skype* as the VoIP application. The objectives of our study were: to understand and evaluate the mobility events and actions taking place in a handover; to evaluate the feasibility of running VoIP over WLAN and CDMA EVDO in the process of understanding mobility; and to provide insight into further research in this area. Our study shows that EVDO link acquisition latency and downlink scheduling, and IP mobility procedures contribute towards VoIP call failures. These results provide crucial input to the design of best-effort real-time applications as well as mobility protocols.**

I. INTRODUCTION

Past few years have witnessed the emergence of VoIP as a viable means of real-time communication. Enterprises are switching to VoIP for various reasons such as cost efficiency, unified data communication, and improved user experience in terms of feature richness [8]. At the same time, enterprises are also deploying the IEEE 802.11 based WLANs. The confluence of VoIP and WLANs is bringing about more mobility and flexibility for enterprises. As this trend continues, handheld devices such as Personal Digital Assistants (PDAs) and Smartphones are being equipped with dual-mode radio interfaces - WLAN as well as high-speed cellular data such as *CDMA EVDO* [5]. This is paving the way for more interesting scenarios in which users enjoy WLAN for higher bandwidth and cost efficiency and the cellular high-speed CDMA EVDO network for greater coverage. Recent research has extensively discussed the performance of VoIP applications such as *Skype*[1] over WLANs [6], [7], [10] and over cellular UMTS networks [9]. These research initiatives and parallel efforts by industry clearly forecasts an increase in the interaction between WLANs, cellular WANs, and VoIP applications.

An important element of WLAN and cellular WAN interaction is *mobility*. Users would likely choose the best-available network for their applications. For instance, an enterprise user within a building may initiate a VoIP call on the enterprise WLAN but may move out of the building into a car and drive away. Such a user will expect the VoIP call to seamlessly *handover* to the cellular WAN network so that a continuous VoIP session can be maintained. There are two crucial problems here. First, the mobility events and actions involved must be within the delay bounds for VoIP. And, second, individual networks themselves (such as WLAN and CDMA EVDO) must be able to support to VoIP; even though this is not a mobility problem per se, it affects the feasibility of VoIP

In this paper, we evaluate the feasibility of VoIP handovers between WLAN and EVDO networks. Our interest lies in VoIP as opposed to traditional circuit-switched mobile voice. We are especially interested in the feasibility of using best-effort VoIP. Our objective is to understand mobility events and actions, and to report the suitability of radio networks for supporting VoIP. Our results provide a basis for building better resilience in to VoIP applications as well as better handover planning and execution of mobility protocols. However, the intention of this paper is not to suggest modifications to radio network design and operations.

Based on our study we make the following observations:

- An evaluation of *Inter-Packet Delay*, a crucial metric for evaluating the sustenance of VoIP calls, shows that traffic on the EVDO downlink is highly bursty. This is attributed to the EVDO downlink packet scheduling algorithms, and causes long unprecedented delays leading to VoIP call failures. This provides useful information for best-effort wireless application developers on embedding adaptation into their design.
- The execution of several sub-operations involved in the establishment of an EVDO link takes about 1.2 seconds. This information provides an estimate on the time available for handover planning and execution.
- The measurement of the handover delay introduced by the IP mobility protocol *Mobile IP* can be large enough

---

[1] http://www.skype.com

to cause call failures. This is useful for mobility protocol developers since any handover protocol design will incur such overheads.

Together, our measurements so far provide crucial insight into the design of real-time mobility of VoIP applications between WLAN and EVDO networks. To the best of our knowledge, this is the first paper that studies and evaluates the feasibility of real-time mobility of VoIP between WLAN and EVDO networks; however, we do recognize the work on evaluation of Skype over individual wireless networks as we noted earlier.

The rest of this paper is organized as follows: Section II provides a description of the WLAN and EVDO networks. Section III describes the several components of our testbed, the network setup, and the data collection process. We evaluate the handovers in Section IV and make corresponding recommendations for performance improvement. Section V discusses our conclusions and future work.

II. BACKGROUND

In this section we provide background information on the operation of the WLAN and EVDO networks we use in our experiments.

A. WLAN

The WLAN used in our experiments utilizes the Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme specified by the IEEE 802.11b MAC specifications. The WLAN uses the 2.4 GHz unlicensed spectrum to transmit packets at any of four different rates, 1, 2, 5.5, and 11 Mbps. The rates are dynamically selected based on packet loss caused by attenuation and/or interference. The transmission and retransmission of packets is controlled by a random exponential back-off mechanism.

B. EVDO

The EVDO network (Revision 0, unless mentioned otherwise) is a broadband CDMA-based network. The network allows peak uplink (known as the reverse link or RL) and downlink (known as the forward link or FL) bandwidth of 153 Kbps and 3.07 Mbps, respectively. A client device is known as an Access Terminal (AT) and the network elements (or cellular towers) that connect ATs to the Internet, collectively form the Access Network (AN). The scheduling on the FL is priority-based and the priorities are determined by the following set of parameters:

- **Radio conditions at the location of an AT**: Each AT in the network regularly sends the AN a brief indicator of the radio conditions. The AN makes a scheduling decision based on the probability at which it can successfully transmit a packet to any of the ATs.
- **Overall traffic conditions between all the ATs and the ANs**: This parameter measures the the number of packets that are being sent and received by an AN element and the ATs associated with that AN element.
- **Transmission history of the AT**: The difference between the last time a specific AT is serviced and the current time. The larger the difference, the higher is the probability of the AT being serviced.
- **User capability**: The specifications of the device that operates as an AT.
- **The membership profile of the AT**: Each different membership level of an AT is offered a different set of QoS guarantees. The higher the membership level, the greater is the probability of the AT being scheduled early. Even though this is mentioned in the specification, we do not believe that it is implemented in the current deployments.

On the other hand, packet transmissions from an AT to the AN on the RL are determined by the following parameters transmitted by the AN to each AT in the network.

- **Rate limit**: Each AN specifies a rate limit; in other words, the maximum data-rate at which the AT is allowed to transmit packets to the AN.
- **Reverse activity bit (RAB)**: RAB is a bit transmitted by an AN at regular intervals to the ATs associated with it. The bit indicates if the AN is busy (1) or not (0); in other words, if the AN senses a large noise floor then it sends a RAB=1 to the ATs, and RAB=0 if not.
- **Rate transition probabilities matrix**: This is a matrix that determines the probabilities of transitioning to a higher or lower rate based on the RAB indicator that is transmitted by the AN to each AT associated with it.

We collectively call the above scheduling scheme as *EVDO Scheduling* in the following part of the paper.

III. EXPERIMENTAL SETUP

In this section we describe the different components of our experiments and their interaction. We use Figure 1 to illustrate our network set up and the experiments.

A. Components

The components used in each of our experiments are as follows:

- **WLAN**: The WLAN is based in San Diego, California, and consists of the CN desktop and a NetGear WLAN router. The WLAN router is connected to the Internet via a TimeWarner Roadrunner cable modem (Downlink speeds at 4 Mbps and Uplink at 300 Kbps). The coverage of the WLAN router extends to up to 100 feet.
- **EVDO network**: The EVDO network, also present in San Diego consists of AN elements.
- **Mobile Node (MN)**: The MN, as shown in Figure 1, is a laptop with two wireless networking interfaces. One interface, *wlan*, is enabled with a built-in IEEE 802.11b wireless network interface (Intel Centrino) and communicated with a WLAN router. The other interface, *evdo*, is enabled with an EVDO PCMCIA card and communicated with the EVDO network. The MN can therefore acquire two different IP addresses because of the two interfaces.
- **Correspondent Node (CN)**: The CN is a desktop connected via a wired Ethernet to a WLAN router. As shown

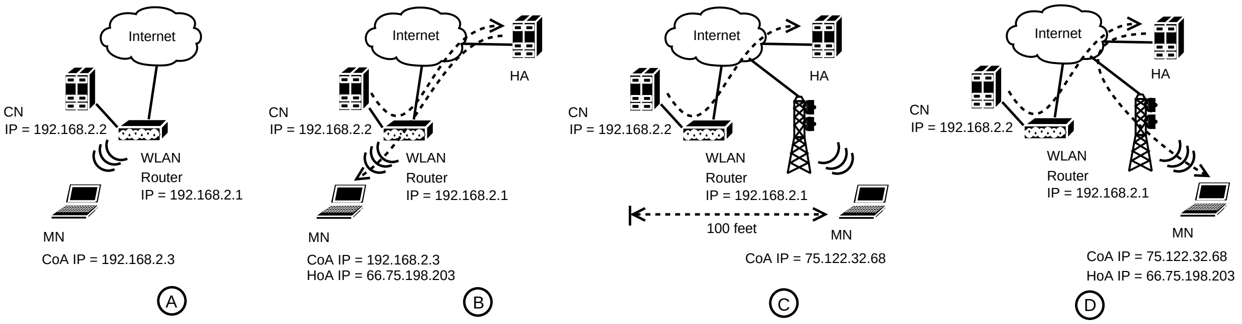

Fig. 1. Network Configuration

in Figure 1, the CN acquires a fixed IP address for all the experiments.

- **Mobile IP (MIP)**: MIP [3] is a protocol that allows a Mobile Node to maintain a single IP address for communication even when it roams across networks acquiring different visited network addresses. The MN relies on a special trust relationship with a *Home Agent* (HA) to forward all packets arriving for its *Home IP Address*(HoA IP) to its mobile IP address, the *Care of Address*(CoA IP) on the Internet. This relationship allows an MN to maintain a persistent IP session using its HoA IP even if the CoA IP changes once the MN moves from one IP network to another. The HA uses IP-in-IP tunneling to route packets arriving for HoA IP to the CoA IP. For our experiments, mobile IP operations are enabled by a commercial HA provided by the Birdstep Corporation[2], located at Seattle, WA. The MN communicated directly with the Birdstep HA using the *co-located mode* [3] of mobile IP. In this mode, the HoA IP for the MN was provided by the Birdstep HA. All IP sessions to the MN were tunneled via the Birdstep HA. The MN's CoA IP is determined by the MN's association to either the WLAN or the EVDO network.
- **Skype VoIP Client**: We used Skype version 1.1.0.79 client software on the MN and the CN to initiate a peer-to-peer (P2P) VoIP call from the MN to the CN.

B. Experimental Steps and Data Flow

Each experiment in our study is conducted as a sequence of steps. They are as follows:

1) The MN's *wlan* interface associates with the WLAN network and acquires a CoA IP from the WLAN router, as shown in Figure 1(A).
2) The MN's MIP client registers its CoA address with the Birdstep HA that provides the MN with an HoA IP, as shown at the bottom of Figure 1(B).
3) The MN's Skype client initiates a VoIP call to the Skype client on the CN. The Skype client uses the HoA IP as its IP to communicate with the CN.
4) The CN starts an audio clip and sends VoIP encoded packets to the MN via the Birdstep HA, as shown in Figure 1(B). These packets traverse a path from the CN, the WLAN router, the HA via the Internet, back to the WLAN router, and finally to the MN.
5) As the MN receives packets, it is walked away from the WLAN router such that the signal strength of the packets from the WLAN router to the MN decreases. The pace and distance covered by the walk is the same for all our experiments.
6) As the signal strength of packets received by the MN decreases below an *Excellent* threshold[3], the MN determines that it is moving outside the transmission range of the WLAN router. This indicates to the MN that it may soon loose connectivity with the WLAN router. As a result, MN decides to initiate an EVDO network association, as shown in Figure 1(C). Since the MN uses the HoA IP to establish the Skype VoIP call, it can now ensure a persistent VoIP session by moving the session from the *wlan* interface to the *evdo* interface, even though its CoA IP changes. This mobility of the IP session between the two interfaces is called a *handover*. Our choice of the threshold is *conservative* and therefore the slightest (yet stable) drop in signal strength results in an execution of a handover. As a result, the MN has ample time to execute a handover when the user is moving at a normal walking pace.
7) After the EVDO association succeeds, the MIP client of the MN sends a *Registration Request* to the HA over the EVDO network. Only when the HA sends a *Registration Reply*, the Skype VoIP call is handed over to the EVDO interface, as shown in Figure 1(D).
8) The CN continues to send VoIP-encoded packets to the MN's HoA IP, thos packets now arrive on the *evdo* interface via the HA.
9) The MN then walks back into WLAN coverage area. As the WLAN router's signal strength is detected to increase beyond the *Excellent* threshold, the MN determines that it is again getting closer to the WLAN and initiates an association with the WLAN. The MN's MIP client now registers its CoA with the HA via the *wlan* interface.

---

[2] http://www.birdstep.com/

[3] Intel Centrino WLAN cards provide qualitative thresholds such as *Excellent*, *Very Good*, and *Poor*. The *Excellent* threshold corresponds to signal strength of -90 to -83 dBm.

10) The MN hands over the Skype VoIP call back to the *wlan* interface, as shown in Figure 1(B).
11) The VoIP call is completed some time after this second handover takes place.

The entire exercise takes about 5 minutes. During each run of the experiment, the HA processes a minimum of three MIP registrations (1) when the MN is on the WLAN network to begin with; (2) after associating with the EVDO network but prior to the VoIP call handover; and (3) when the MN moves back to the WLAN network. Each MIP registration with the HA uses a 80-second update time-out for liveness. Therefore, the MN executes re-registrations every 70 seconds, approximately.

IV. EVALUATION

The metrics we study are Inter-Packet Delay (IPD) on the WLAN and the EVDO interfaces of the MN, the EVDO link setup delays, and MIP registration delays. We study these metrics for all our experiments because these metrics provide sufficient information to achieve our objectives of understanding the feasibility of sustaining VoIP calls over the WLAN and EVDO networks and handovers between them.

A. *Experiment Outcomes*

VoIP calls in 60% of all the experiments we ran failed. The main factors leading to call failures were long EVDO FL scheduling delays, MIP registration failures, long link acquisition latency in EVDO, and MIP re-registration failure. Note that not all of these are handover-related failures. We explain the reasons for dropped calls and the characteristics of the VoIP calls in more detail in the following section.

B. *Inter-Packet Delay (IPD)*

In our experiments, the Skype VoIP call is setup by the MN to the CN. We use an audio clip which is played in a loop at the CN. Skype sends VoIP-encoded UDP packets from the CN to the MN, which receives them over the WLAN or the EVDO network. The IPD is the interval or delay between successive UDP packets sent by the CN or received at the MN. IPD is a crucial metric because not only does it serve to understand the jitter and packet loss characteristics of a network path from the CN to the MN, but more importantly it reveals the effects of the packet scheduling techniques on the feasibility of any network to support VoIP calls. Figure 2 shows a CDF of the IPD between packets sent by the CN to the MN (labeled as IPD-CN), and the IPD between packets received at the MN over the WLAN interface (labeled as IPD-WLAN) and the EVDO interface (labeled as IPD-EVDO).

We observe that a majority of the packets sent by the CN to the MN were at 20 to 40 ms intervals. This is because Skype uses adaptive VoIP codecs which use packetization at 20 ms to 40 ms based on the performance of the call [9]. IPD-WLAN shows that about 90% of the packets were received at intervals of less than 20 ms by the MN over the WLAN interface. Larger intervals, greater than 60 ms in IPD-WLAN indicate packet loss or jitter caused on the network path between the CN, the HA, and the MN's WLAN interface.

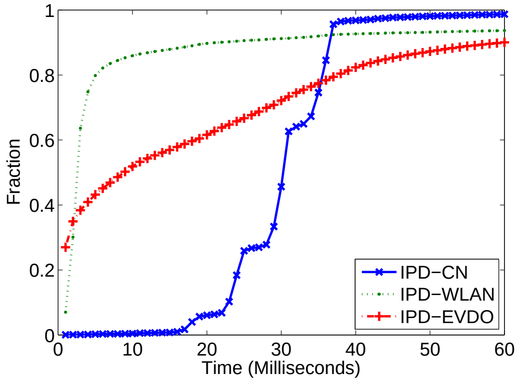

Fig. 2. CDF of the Inter-packet delays (IPD).

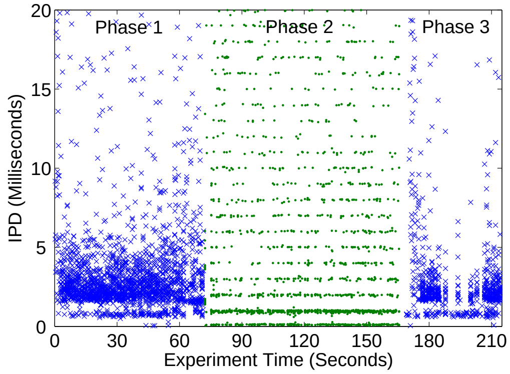

Fig. 3. Time series of the Inter-packet delays (IPD).

In Figure 2, IPD-WLAN and IPD-EVDO are interestingly skewed towards the lower intervals of IPD, suggesting the bursty nature of both the media. In other words, a large portion of the packets are received by the MN in bursts, where the IPD is close to 0 to 2 ms, and the bursts of packets are separated by longer IPDs indicated by the long tail in the distributions. Based on the two different network paths shown in Figure 1(B) and 1(D), we make a simplifying assumption that the differences in the IPDs are primarily due to the differences in radio access network behavior of WLAN and EVDO. This is justifiable because both the networks share the same CN-to-HA path entirely and also the path from the HA to the Internet. As a result, it is the WLAN router to MN and EVDO AN to MN links that determine the difference in the corresponding IPDs. For a clear understanding of the WLAN and the EVDO packet scheduling characteristics, we plot a snapshot time series of IPD values from 20 to 40 ms in Figure 3. In this figure, the three phases of the experiment are annotated.

WLAN Reception: In phases 1 and 3 of the experiment, the Skype VoIP call operated over the WLAN network. We observe that a large number of packets during these phases had very low IPDs, indicating bursty packet reception. These bursts occur primarily due to packet buffering at routers on the network path between the CN and the MN. The MN used for our experiments is the only client associated with the WLAN router; we detected no other WLAN routers or clients in close proximity of our WLAN router and therefore we can assume negligible interference effects on the reception of packets from the WLAN router. As a result, we believe that IPD-WLAN is more sharply skewed towards the lower IPD values; it would have been less bursty if interference effects had played a significant role in our experiments. Also, the probing and association mechanisms in WLANs can also involve long delays, sometimes in the order of several seconds. These mechanisms did not impose significant delays in any of our experiments, however, such delays can be another potential cause of call failures in handover schemes if the MN happens to lose EVDO connectivity before fully associating with a WLAN access point (AP).

EVDO Scheduling: During phase 2 of the experiments, the MN received packets from the CN over the EVDO interface. Figure 3 shows the IPD of the packets received during this phase. We observe that majority of the packets were received at intervals that were multiples of 1 ms each. This occurs due to the scheduling of packets at the AN of the EVDO link. An AN schedules the transmission of a packet to each MN in slots of 1 ms. The scheduling algorithm chooses a slot based on the properties of the FL we discussed in Section III. As a result of this scheduling of packets by the AN, the IPD to the MN varied significantly, even in the order of seconds (not shown in Figure 2). These long and unprecedented IPDs were the prominent cause of call failures during many of our experiments. In most of such failed experiments, the MN received no packets for a few seconds (complete silence), ranging from 4 to 6 seconds. These silence periods were long enough for the Skype's time-out mechanism to deem the call as a dropped call. A large burst of packets immediately followed the silence period indicating that these packets were buffered at the AN and scheduled for transmission only after the large silence period. Furthermore, we found that the long scheduling delays are also responsible for unsuccessful MIP registrations, resulting in a few more failed experiments.

Since the EVDO network belonged to a wireless service provider, we had no control or knowledge of the number of users simultaneously using the network and also the traffic flowing on the network. To account for these unknowns, we ran our experiments at different times of a day. The similarity in traffic characteristics convinces us that the scheduling algorithms used by EVDO unfavorably affects even best-effort real-time applications such as Skype VoIP. The long scheduling delays will not only deteriorate VoIP quality, they even cause call failures.

Recommendations: The Skype we used is not well-equipped to handle the long silences followed by large bursts. To make Skype and other VoIP software feasible for use over an EVDO network, better delay adaptation techniques that can handle such long periods of inactivity and large bursts need to be investigated. Specifically, session progress messages (similar to those used by RTCP [2]) between peers (the MN and CN in our experiments) with the explicit knowledge of long silence followed by large bursts should be investigated. Also, VoIP can be better supported by explicitly providing QoS on EVDO networks

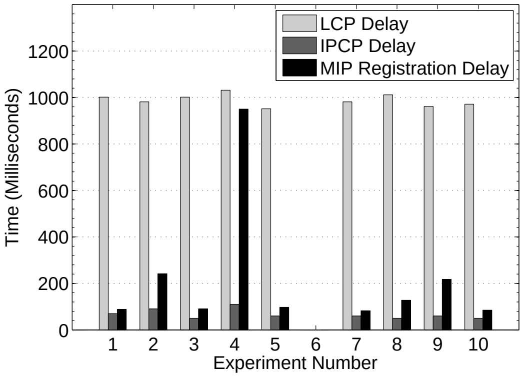

Fig. 4. PPP and MIP registration delays.

C. Handover

The handover operation involves several sub-operations of EVDO link establishment and MIP registration. We evaluate these sub-operations to measure, (1) the typical delays per sub-operation, and (2) the cause of failure of a call due to an incomplete or delayed sub-operation. The sub-operations are as follows:

Point-to-Point Protocol (PPP): An EVDO association between the MN and the EVDO AN is established using the Point-to-Point Protocol (PPP) [1]. PPP consists of two protocols, the Link Control Protocol (LCP) and the IP Control Protocol (IPCP). Figure 4 shows the LCP and IPCP protocol delays for 10 randomly picked experiment runs. The other experiments exhibited similar behavior. We observe that in 9 out of the 10 experiments shown, the LCP and IPCP delays are approximately 1000 ms and 100 ms, respectively. These values indicate that any handover planning algorithm should set aside a delay budget of around 1100 ms for establishing an EVDO link alone. The delay values for the 6th experiment are not shown because they were significantly larger than the y-axis scale; 17 and 12 seconds for LCP and IPCP, respectively. Such long delays will certainly cause call failures. These delays will also certainly increase as network load increases with more subscribers.

MIP Registration: An MIP registration sub-operation involves the MN reporting its CoA IP to the Home Agent but only after the EVDO link has been established. An MIP registration consists of two messages; a *Registration Request* message sent by the MN to register its new CoA to the HA and a *Registration Reply* message sent by the HA to the MN to confirm the registration. A handover of the Skype VoIP call is completed soon after the HA receives the registration request message from the MN. Both the registration messages are subject to loss and therefore, if an MN does not receive a registration reply within a specified time interval it retransmits registration request messages one or more times as specified by the MIP configuration. The time interval between the first registration request sent by the MN and the registration reply received by the MN constitutes the MIP registration delay.

Figure 4 shows the MIP registration delays for the 10 experiments. We observe that in 9 out of the 10 experiments the MIP registration delays were between 100 ms to 200 ms. These delays are also influenced by retransmission of MIP registration messages. These delays and the possibility of retransmissions should be considered for handover planning in addition to the EVDO link establishment delays. The MIP registration delay in experiment 4 was of about 900 ms. Since the MIP registration sub-operation identifies the actual handover of the Skype call between the interfaces, such large MIP registration delays may induce large IPDs - causing call failures. Although only one call from all our experiments failed due to an MIP registration failure, we believe there is a serious need for better handover techniques that can reduce the overall delays and achieve fast and reliable handovers.

Recommendations: Techniques to mitigate the effect of long delays and potential packet loss of registration messages need investigation since these messages currently are on a critical path for call handover. For instance, with prediction and some network conifiguration information at hand, the MN may be able to perform the registration before it relinquishes WLAN and attaches to EVDO [4].

V. CONCLUSIONS

In this paper we discussed the methodology and evaluation of the handover of best-effort VoIP calls between a WLAN and an EVDO network. Our study showed that the current EVDO scheduling policies may induce delays that unprecedented by VoIP applications such Skype, resulting in VoIP call drops. We also observed that the handover delays can also cause call failures. Our study calls for (1) greater application elasticity to handle long silences followed by large packet bursts; (2) intelligent handover planning algorithms that consider link acquisition latency, especially for EVDO, and (3) techniques to enable fast MIP registration operation and avoid call failures.

We believe that the experiments we conducted were fairly conservative in nature; the MN moved at a normal walking pace and the trigger for handovers was a signal strength value corresponding to a conservative *Excellent* threshold. We believe that if users move quickly between WLAN and EVDO they will require more agile handover planning. A majority of our following work will include the use of the observations presented in this paper and more experimental data to design agile and reliable handover algorithms and determine the necessary parameter settings to achieve VoIP handovers across disparate networks.

REFERENCES

[1] RFC 1661, http://www1.ietf.org/rfc/rfc1661.txt.
[2] RFC 1889, http://www.ietf.org/rfc/rfc1889.txt.
[3] RFC 3344, http://www1.ietf.org/rfc/rfc3344.txt.
[4] RFC 4068, http://www1.ietf.org/rfc/rfc4068.txt.
[5] S-856 Rev A - cdma2000 High Rate Packet Data Air Interface Specification (1xEVDO).
[6] S. A. Baset and H. Schulzrinne. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. In *Proceedings of Infocom*, Barcelona, Spain, May 2006.
[7] S. Guha, N. Daswani, and R. Jain. An Experimental Study of the Skype Peer-to-Peer VoIP System. In *Proceedings of IPTPS*, pages 1 – 7, Santa Barbara, CA, February 2006.
[8] T. Henderson, D. Kotz, and I. Abyzov. The Changing Use of a Mature Campus-wide Wireless Network. In *Proceedings of ACM MOBICOM*, Philadelphia, PA, September 2004.
[9] T. Hobfeld, A. Binzenhofer, M. Fiedler, and K. Tutschku. Measurement and Analysis of Skype VoIP Traffic in 3G UMTS Systems. Technical Report 377, Department of Computer Science, University of Wurzburg, Germany, December 2005.
[10] K. Suh, D. R. Figueiredo, J. Kurose, and D. Towsley. Characterizing and Detecting Relayed Traffic: A Case Study using Skype. In *Proceedings of Infocom*, Barcelona, Spain, May 2006.