HOPOVER: A New Handoff Protocol for Overlay Networks

Fan Du, Lionel M. Ni and Abdol-Hossein Esfahanian
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824

Abstract-**This paper presents a new handoff protocol, Handoff Protocol for Overlay Networks (HOPOVER). This protocol is compatible with Mobile IP and is designed specifically for overlay networks where handoffs happen both horizontally and vertically. Handoff performance is enhanced by a number of measurements including pre-reserving resources, packet buffering in the new network and packet forwarding from the old network to the new network. Our simulation proved the effectiveness of these measurements.**

1 INTRODUCTION

Handoff is the process in which a wireless device moves from current cell to a new one. The performance of handoff scheme directly affects the overall performance of mobile applications. The importance of handoff performance keeps going up as today's wireless networks have increasingly large user population and decreasing cell size. Traditionally, the term handoff is used to refer to the case in which the two cells involved belong to the same wireless network, as shown in the left side of Figure 1. Such handoffs are referred as *horizontal.* As number of wireless networks exponentially increases, they are overlaid with each other to form hierarchies. In such structures, mobile devices can perform handoffs among different layers of networks. Such handoffs are referred as *vertical*, as shown in the right side of Figure 1.

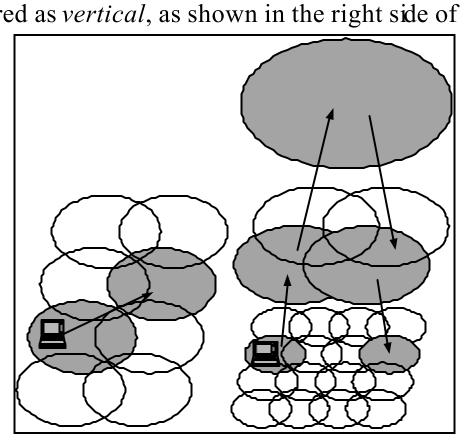

Figure 1. Horizontal and Vertical Handoff

A lot of research work has been done to improve the performance of horizontal handoffs. Examples include [4][6][7][8][9][10][11][12]. However, much less effort has been spent on vertical handoff research [2][5][11][15]. Up to now, most networks still support vertical handoff through Mobil IP [11], which was designed to provide macro-level and slow moving mobility. The scheme suffers from high overhead when high frequency handoffs are needed. More importantly, handoffs cause big gaps in packet flows. Besides Mobile IP, there are only few other approaches and none of them got extensive use due to their limitations and shortcomings.

To address this problem, we designed a new overlay handoff protocol named HOPOVER (HandOff Protocol for OVERlay networks), which supports both horizontal and vertical handoffs and is compatible with Mobile IP. The main approaches used in this protocol include: facilitating a special wireless resource reservation protocol to pre-reserve resources in the new cell and along the path from the new cell to the flow transmission parties; buffering in the new network for the MH and forwarding packets from the old network to the new network. With these measurements, HOPOVER significantly enhances handoff performance.

The remaining part of this paper is organized as follows. First, related work is reviewed in Section 2. Then we present the design of HOPOVER in Section 3. Section 4 presents the simulation results. Section 5 discusses issues and attributes of HOPOVER. Finally, conclusion is given in Section 6.

2 RELATED WORK

In this section, we go through some of the most representative research works in the area of handoff.

2.1 Mobile IP

Traditionally, handoff schemes across networks are constructed based on Mobile IP [11]. Mobile IP provides a simple and efficient solution to maintain IP connectivity for mobile hosts. It is a proposed standard protocol by IETF, and it is widely used and supported by current systems. Mobile IP handles mobility as follows:

- Every Mobile Host (MH) has a Home Agent (HA), which knows the MH's permanent IP address. Every site that wants to allow visitors creates a Foreign Agent (FA).
- When a MN moves to a foreign network, it registers with the FA and gets assigned a Care-Of-Address (COA).
- The FA notifies the HA of the MH's COA. The FA also starts forwarding packets for the MH.
- When packets are sent to the MH's home network, the HA forwards them to the current COA. At the same time, the HA notifies the sender about the new location.
- The sender then sends following packets directly to the new location.

Mobile IP was designed to support macro level mobility and slow moving hosts. It requires that the HA be notified each time the MH receives a new COA, which happens when the MH moves into a foreign network or performs a handoff between cells. Such requirement is OK when MHs move at low speed and handoffs only happen at low frequency.

0-7803-7400-2/02/$17.00 © 2002 IEEE

However, as wireless communication evolves, the number of mobile devices increases rapidly, and the use of high-speed moving mobile devices becomes more and more popular. At the same time, to accommodate the increasingly large number of devices, cells have to be designed smaller, thus causing even more frequent handoffs for mobile devices. All of these made the design principle behind Mobile IP no longer suitable and the overhead associated with handoff too expensive.

2.2 *Cellular IP*

To provide a low-overhead handoff scheme, Cellular IP is proposed. [5] Cellular IP adopts a hierarchical approach to manage mobility. At the higher level, Mobile IP will still be used, and Cellular IP is used to handle lower level mobility. When a MH arrives a Cellular IP network network the first time, traditional Mobile IP operations are performed to inform the HA. Now all the packets for the MH will be routed to the gateway router then to the MH. Later, when the MH moves within the wireless network and performs handoffs, it only causes the gateway router to change the base station to forward to. No operation with HA is necessary.

Cellular IP reduced the overhead associated with handoffs and made handoff process smoother. However, it is only applicable for horizontals handoffs where the MH moves within the same network. For handoffs between two networks, MHs still go back to Mobile IP.

2.3 *BARWAN*

The Bay Area Research Wireless Access Network (BARWAN) [3] also proposed a vertical handoff scheme. This scheme is based on Mobile IP, but with modifications to make it fast. Basically, BARWAN improves handoff performance by converting packet delivery mode from 1-1 to 1-n. For each MH, packets are multicast to multiple BSs. BARWAN requires one (local) multicast address for each MH. This requirement makes it unrealistic in real life. This approach is possible only if all the related networks are under control of the same administration unit. Local multicast groups must be created and maintained, and base stations from multiple networks need to cooperate in the process.

3 HOPOVER: HANDOFF PROTOCOL FOR OVERLAY NETWORKS

HandOff Protocol for OVERlay networks (HOPOVER) is our solution to this problem. It is low overhead, scalable and close to seamless. The solution incurs low overhead, even with large number of MHs/BSs and high frequency handoffs. It is scalable: the overlay network is easily expandable without affecting the performance of handoffs. Seamless means that a handoff causes no interruption on MHs' current sessions, no user interference is necessary and applications will not be affected. Perhaps no solution can really guarantee seamless performance. To be modestly safe, we call our solution "close to seamless".

HOPOVER is designed to address both macro and micro level mobility. Vertical handoffs represent macro level mobility and involve many complex issues. Horizontal handoffs represent micro level mobility and are often easier to support. HOPOVE dynamically decides the appropriate action based on what kind of handoff it is handling. Parts of the following discussion are applicable to vertical handoff only, and they are easy to tell based on the context.

3.1 *System Model and Overall Approaches*

In the design of HOPOVER, we applied the following approaches.

- Mobile IP compatible: Mobile IP is the suggested standard by IETF, and HOPOVER is designed to be compatible with it.
- Pre-resource reservation: the handoff process utilizes a special wireless resource reservation protocol to guarantee that MHs have required resources once they enter the new cell/network.
- Buffering: To avoid loss of packets and interruption of flows, the target BS is instructed to buffer packets for the incoming MH. To be sure the sessions will not be interrupted, several nearby BSs can be instructed to buffer packets for the same MH simultaneously.
- Forwarding across different layers: When a MH vertically handoffs to another layer, its packets can be forwarded to the new layer by the BS of the old layer.
- Contact HA only when necessary. It is frequently the case that a MH goes back and forth between two networks or cells for a number of times. HOPOVER does not contact the HA each time a handoff is performed. Instead, only when it is sure that the MH has entered into a stable condition, it contacts the HA to have the care-of address information updated. This feature is based on our use of forwarding packets across cells and networks.

3.2 *Resource Reservation in Wireless Environment*

To guarantee QoS of real-time flows, flows must be allowed to reserve network resources. A big reason behind packet losses and interrupted sessions is that resources are unavailable in the new cell or along the path from the flow sender to the new cell. Comparing to horizontal handoffs, resource reservation is more complicated with vertical ones. In horizontal cases, both BSs involved are frequently in the same IP subnet and share most part of resource reservation path leading to the transmission parties. But in vertical cases, there is usually big difference in resource reservation paths.

RSVP [13][14] is the most widely used resource reservation protocol, however two big problems prevent RSVP from being used directly in wireless networks. One is the handoff problem as mentioned before, and the other is called "poor link" problem. Solving handoff problem is a complicated process. Pre-reservation is used to address this aspect of the problem. We will discuss the details later. In this section, we focus on the way we solve poor link problem.

Poor link problem stems from the high *Bit Error Rate* (BER) associated with wireless links. Such poor links cause difficulties in reserving and refreshing resources. To overcome this problem, we modified RSVP to adapt to wireless environment. We call the adapted version WRSVP (Wireless RSVP) [1].

If RSVP is used in wireless networks directly, MH receivers will have trouble in both reserving and refreshing resources, due to the very high BERs. In RSVP, flow receivers reserve resources by sending "Reserve" packet s to the sender. The routers along the path decide if the request can be accepted. After resources have been reserved, periodical refresh is needed to keep the reservation alive. In RSVP, only when a reservation can NOT be made will the receiver be notified. If a Reservation or Refresh message is lost, the MH sent the message would not be notified, thus it will falsely assume the reservation or refresh has succeed. If several successive Refresh messages are lost, a flow may be torn down automatically by the network, because the reservation will time out. Such problems are ignored for wired networks, where links are much more reliable.

To solve these problems, there are some obvious methods. One of them is to have all the routers send back acknowledgment (Ack) messages to the MH receiver no matter the reservation/refresh is successful or not. If the MH receives the Ask message(s) over a pre-defined period of time, it knows that the Reservation/Refresh message has been received. The drawback of this method is that it will increase the overhead and consume bandwidth each time the MH refreshes the reservation. Another method is to use larger reservation time out values in routers. It certainly reduces the possibility of unwanted automatic torn-down, but at the same time, resources will be wasted when the MH does stop the reservation, because the MHs are not required to send explicit Release messages. Moreover, even if the MHs do send, the messages can get lost in the wireless environment.

WRSVP solves the poor link problem by separating the reservation process into two parts. One part is between the MH receiver and the BS of the cell, and the other part is between the BS and the wired network. In the wired part, we let the BS do the refresh work for the MH, i.e. the BS periodically sends Refresh messages to the wired network on behalf of the MH receiver. In the wireless part, we add Ack messages to confirm message exchanges. We assume that a MH receiver wants to keep its flows until it explicitly notifies the BS to release them (we discuss the case where MHs are receivers of the flows. If MHs are senders, the case is similar or simpler). More specifically, it works as follows:

The MH receiver sends the first Reservation message to the BS. If the BS can offer the required resources in local network (the cell), it forwards the requirement to the wired network.

After a certain period of time (predefined time out value), if no error message is received from the network, the BS knows the reservation has been accepted. It sends an Ack message to the MH, indicating the success of the reservation. If error messages are received from the wired network, the BS sends an N_Ack message to the MH, indicating the failure of the reservation.

On the MH's side, it waits for the Ack/N_Ack message. If after a predefined period of time, no Ask/N_Ack message is received and the flow it is waiting for does not come, it knows something went wrong with the transmission. Then, it resends the Reservation message. The above process is repeated until an Ack/N_Ack message is received from the BS or the flow comes.

Once the reservation is made, the BS periodically refreshes it in the wired network on behalf of the MH.

When the MH wants to terminate the flow, it sends a Release message to the BS. Again, Ack/N_Ack message from the BS to the MH is used to guarantee that the BS can finally receive this Release message. Upon receiving this message, the BS stops refreshing the flow, or sends an explicit Release message to the wired network.

Now, MHs no longer need to periodically send Refresh messages. Thus, the reservation overhead for the wireless part is greatly reduced, and more valuable wireless bandwidth is saved. Also, unwanted flow torn-downs are avoided.

3.3 HOPOVER Handoff Procedure

With the ability to reserve resources in wireless networks, we are now ready to improve handoff performance using the approaches we discussed earlier: pre-resource reservation, buffering and forwarding. We present the handoff process in the following. Here, we present the more complex vertical case. Horizontal case is simply part of it.

Step 1: Handoff Prepare

MHs can detect that they are going out of current cell or a better network is available in terms of bandwidth and other factors. The decision is usually based on the comparison of base station beacons the MH receives. When a MH decides it may encounter a handoff, it starts with the following handoff-prepare processes.

The MH chooses a small group of neighboring BSs and sends them a Handoff-Prepare (HP) packet. Each BS forwards the packet to its gateway router. The BS selection is based on signal strength, bandwidth, pricing and other factors, if such information is available from BS beacons. When such information is unavailable, the MH simply makes random selection to a pre-set limit.

The authentication server of the new network verifies the validity of the authentication information included in the HP packet. If it is invalid, a HP_NACK (negative Ack of the HP) is sent to the MH, and no further prepare work will be performed. If it is valid, routing state along the path from the gateway router to the chosen BS(s) are set.

Based on the resource reservation information included in the HP packet, resources are reserved in target cells and along the path(s) from the MH's current session sender(s) to the cell(s).

Each of the chosen BS allocates a piece of buffer for the MH and prepares to buffer packets for the MH.

Each of the new BS(s) sends the old BS a HP_ACK packet. Upon receiving such a packet, the old BS adds the corresponding new BS to the forward list for that MH and begins forwarding packets to the new BS.

By the end of these procedures, routing information along the path from the new gateway router(s) to the new BS(s) is set up and packets have been buffered for the MH. Also, with the help of WRSVP, necessary resources have been reserved in the new cells and along the path(s) from the MH's session senders to the new BS(s). All of these help to make the upcoming handoff process smooth.

Step 2: Handoff

When the MH decides it is actually moving to another cell, the actual handoff is performed using the following procedures.

The MH sends a Handoff message to the BS of the cell it is actually moving. The new BS then begins forwarding packets to the MH including the buffered ones.

The new BS sends a Leave message to the old BS and all the "handoff preparing" BSs.

The old BS records the MH's current network. Later packets will be forwarded there. It stops forwarding packets to other BSs which are on the forwarding list.

The old BS removes the MH from its "current MH" list.

Other handoff-preparing BSs remove the related routing information, allocated buffers and buffered packets.

The old BS and other handoff-preparing BSs delete themselves from the resource reservation tree they joined for that MH by sending a Release message. Thus, the network releases those reserved resources.

Step 3: Updating Mobile IP information

After handoff, the previous BS maintains the forwarding address for the MH. To avoid wasted resources and additional delay, sometimes, the MH's Mobile IP FA information should be modified to reflect the MH's current address. We use timers to make sure that the Home Agent is contacted only if the MH stays in the new cell for long time.

After a handoff, both the old and new BSs set up a timer for the MH. If after the timer expires, the MH is still in the new network. The new BS sends Mobile IP update packet to the MH's HA, so the new BS becomes the new FA for the MH. Also, the new BS notifies the old BS to remove the forwarding information for that MH.

If before the timer expires, the MH moves back to the previous network. No contact with the Mobile IP HA is performed. Both timers are removed.

If before the timer expires, the MH moves to a third network. The second BS sends Mobile IP update packet to the MH's HA to make itself the new FA of the MH. At the same time, the second BS begins monitoring the MH's stay in the third network.

4 PERFORMANCE EVALUATION

To verify the effectiveness of HOPOVER, we simulated the impact of handoff on real-time applications. We implemented prototype of HOPVER and compared its performance with Mobile IP.

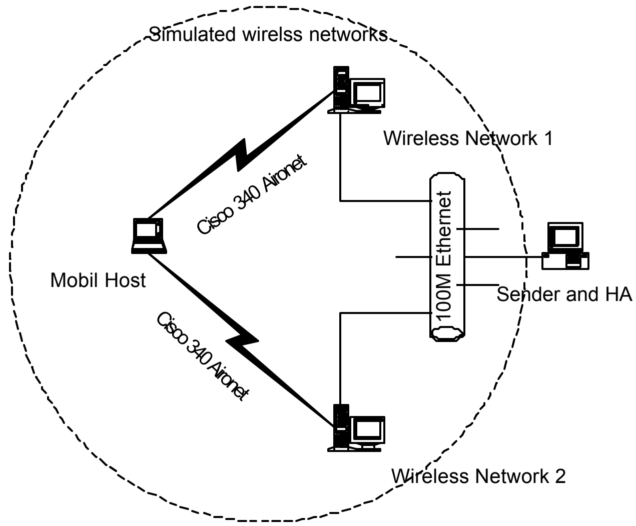

Figure 2. Simulation Topology

4.1 *Simulation Environment*

The wireless testbed used in the experiment consisted of two wireless LANs connecting the BSs and the MH, and a 100M Fast Ethernet network connecting the two BSs. The wireless communication devices were Aironet 802.11b wireless devices that use Direct Sequence Spread Spectrum modulation technique to provide reliable communication and to protect against eavesdropping. All hosts run the Linux OS, kernel 2.2.14, which supports TCP/IP protocol package.

The machines used to simulate the overlay wireless network included two Pentium II PCs and a Pentium III laptop. On the MH machine, only one program was run. The program simulated both the MH handoff daemon and a user sound player application. Gateway program was run on the other two machines, so they played as both the gateway routers and base stations. On another machine, we run a HA program and a user sender program. The topology is shown in Figure 2.

To represent the transmission delay, packets to be transmitted were placed in a queue and held until the delivery time. The BSs used circular buffer to hold packets for the MHs. Since the buffer was circular, only the newest packets were kept and the old ones dropped.

4.2 *Simulating Handoff with a Controllable Error Model*

To simulate handoffs in short distance is not a trivial task. Generally, wireless connection quality is physically determined by characteristics such as frequency, distance, noise and so on. It is difficult to quantify all the elements and associated level of errors. A natural idea would be to move the MH between two BSs, so it hands off to the closest BS in the movement. But this idea fails in reality, because in short distance as in one room, wireless links are always highly reliable. Also, it is difficult to get the movement of MH into an accurately repeating pattern so the results are comparable.

In our experiments, we designed a controllable error model to manually introduce packet losses. As just mentioned, the links of the wireless LAN in short distance can be considered error-free, so only the errors generated by the error model would affect the transmission. In our model, the error rate in each BS changes periodically, to represent different distances with the MH. We simulated the case that the MH moves back and forth between those two BSs, so we set the two BSs to have same error change period, but different phases. That is, when the error rate is at the highest point in one BS, it is at the lowest point in the other BS. In other words, when the MH is closest to one BS, it is farthest away from the other one.

The controllable error model was implemented by modifying the Aironet wireless device driver, which is a loadable Linux kernel module. It can be easily integrated into the kernel or detached from it by using "insmod" and "rmmod" commands respectively. With this error mode, a packet is discarded by the BS if the error generator determines that an error has occurred.

4.3 Sound File and Sound Player

In the experiments, a WAV format sound file was transmitted. The WAV file was a 3-minute long real human talk. It was sent from the sender to the MH sound player simulator using UDP. A log file was used to record information for all the packets received, such as time and packet number. Also, the sound player simulator recorded the number of packets lost and missed time slots. The player simulator used a playback buffer and the packets were played back with a small period delay to reduce gitter. Both methods are common approaches to enhance playback quality.

4.4 Simulation Results and Analysis

The following default parameters were used in the simulations unless specified otherwise. The handoff threshold was the loss of 10 consecutive beacons. The packets were sent at 50/s frequency, and the play back delay was 0.1 second. When HOPOVER was not used, handoff depended on Mobile IP only. The results of handoff are shown in Figure 3 and Figure 4. With appropriate setting, e.g. sufficient buffer, HOPOVER can cut the numbers of lost packets and missed slots by 80% or even more. (Lost packets and missed slots can be different because packets can arrive late and miss the time slot they are to be played). The result is especially apparent when the sender and home agent is far way, the case HOPOVER is designed for.

Figure 3 shows effect of playback buffer size. The buffer size changes from 0.1 (very small) to 1 second (very big). BS buffer is half the size of playback buffer. The playback delay is 0.1 seconds and the remote host distance is 150 milliseconds. As expected, the number of lost packets and missed slots decrease as the size of playback buffer gets bigger. When HOPOVER is not used, the decrease is very steady. When HOPOVER is used, there is an elbow point where buffer size is 0.25 seconds. The effect of HOPOVER is more apparent when the buffer size is larger. HOPOVER cuts the number of lost packets/missed slots by about 70%. Even with very small buffer, the effect is about 60%. Without HOPOVER, the gap caused by each handoff is 0.8 to 0.9 seconds, which is apparent to human ears. With HOPOVER, the gap is reduced to 0.2 to 0.4 seconds.

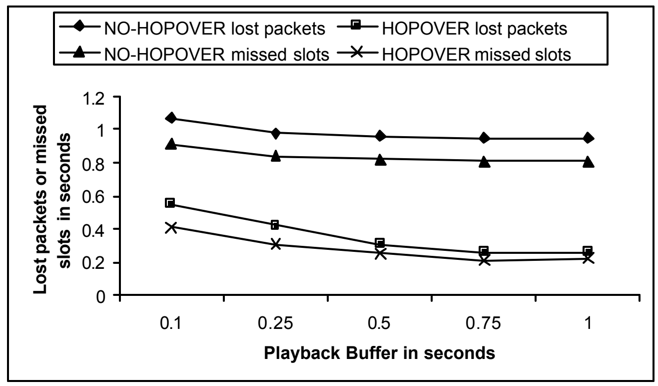

Figure 3. Effect of Playback Buffer

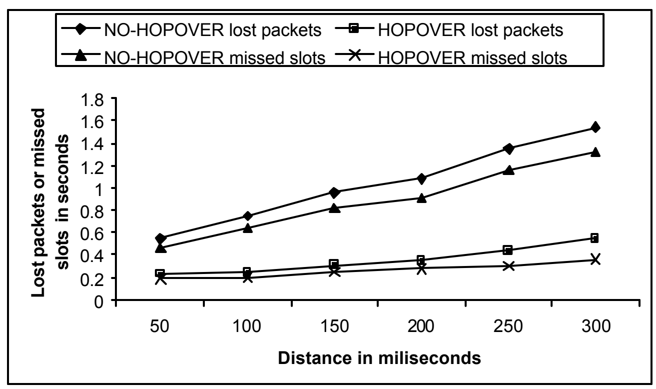

Figure 4. Effect of Remote host distance

Figure 4 shows the effect of remote host distance. The distance changes from 50 ms (very close) to 300 ms (very far away). As expected, the number of lost packets and missed slots increases as the distance of remote hosts gets bigger. But with HOPOVER the increase is much slower. Without HOPOVER, the gap caused by each handoff ranges from 0.53 to 1.41 seconds, nearly a triple. With HOPOVER, the change is only from 0.26 to 0.31 seconds. With very close HA and sender, HOPOVER cut the gap by about 50%; with very far away condition, the cut is nearly 80%. With HOPOVER, since the packets sent to old BS are not lost, QoS becomes less sensitive to the distance of sender and HA.

5 DISCUSSION

In this section, we briefly discuss the overhead associated with HOPOVER and some deployment issues.

5.1 *Overhead Analysis*

The overhead of HOPOVER mainly includes the following parts: 1) handoff packets exchange; 2) processing power and spaced used by buffering and forwarding operations at BSs and 3) resources occupied by pre-reservation in targeting and neighboring cells and reservation with neighboring networks.

The first part of HOPOVER overhead includes the necessary packet exchange between MHs, BSs and gateway routers. Since each of the packets is very short, they are not taking much resource from the system. In analyzing the second part of HOPOVER overhead, we point out that as part of the wired network backbone, BSs usually hold much more processing power and space than those of MHs. In addition, more resources can be added as necessary since there is no space limitation. As the processors become more powerful and memory technology advances rapidly, we don not anticipate this part of HOPOVER overhead to be an issue. The last part of HOPOVER overhead also poses no big concern due to two reasons. First, the number of neighboring cells to use can be limited by the MH configuration. Usually the number is very small. Second, the overhead is mitigated greatly by the use of WRSVP/RSVP structure. Although several BSs make reservation from wired network, the common part of the reservations is merged. So, the overhead is usually limited to the BSs, where we argue resources can be added with little effort.

5.2 *Deployment Issues*

For any new protocol, the expectation of a one-shot deployment everywhere is unrealistic. A very important and highly desirable property is that the new protocol can cooperate with the old network components. That allows a smooth, step-by-step deployment. HOPOVER follows this design philosophy.

All modifications introduced by HOPOVER and WRSVP are local. The interface to other component of the network is unchanged. Only standard Mobile IP and RSVP packets are exchanged with other parts of the network. So for components outside of the HOPOVER network (or network segment), the use of HOPOVER is transparent. For BSs, the upgrade to HOPOVER benefits all MHs, not just those HOPOVER aware ones. For old MHs, they may just find that resource reservations become more stable and handoffs smoother. The reason is that WRSVP BS daemons help to establish and refresh resource reservations; and HOPOVER BSs help in resource pre-reservation, buffering and forwarding packets. For HOPOVER MHs, they can tell if a BS supports HOPOVER from its beacon. For a regular BS, MH should follow the normal Mobile IP and RSVP procedure. But when the MH is encountering a handoff, it can try to send Handoff-Prepare packets to the neighboring cells. If fortunately, the targeting cell supports HOPOVER, the MH can expect a better service after it gets there.

6 CONCLUSION AND FUTURE WORKS

In wireless networks, handoffs often cause severe QoS damage. HOPOVER enables smooth handoffs intra- and inter-network, and it is compatible with Mobile IP. HOPOVER helps mobile devices using the following measurements: pre-reserving resources in the new cell and along the path from the new cell to the flow transmission parties; buffering in the new network for the MH and forwarding packets from the old network to the new network. With these methods, HOPOVER significantly enhances handoff performance. We also addressed poor link problem which is a big obstacle in resource reservation in wireless networks. We separate resource reservation into wired and wireless parts and add Ack messages into wireless message exchanges.

Our simulation results proved the effectiveness of these measurements. HOPOVER significantly reduces the gap caused by each handoff. The effectiveness of HOPOVER is most apparent when the sender and home agent are far away, the case QoS improvement is needed the most.

REFERENCES

[1]. F. Du, X. Dong, H. D. Hughes and L. M. Ni, "Resource Reservation in Wireless Networks", *Symposium on Performance Evaluation of Computer and Telecomm Systems 1999.*

[2]. M. Stemm and R. H. Katz, “Vertical Handoffs in Wireless Overlay Networks”, *ACM Mobile Networking (MONET), Special Issue on Mobile Networking in the Internet 1998*.

[3]. R. H. Katz, E. A. Brewer, E. Amir, H. Balakrishnan, A. Fox, S. Gribble, T. Hodes, D. Jiang, G. T. Nguyen, V. Padmanabhan, and M. Stemm, “The Bay Area Research Wireless Access Network (BARWAN),” *Proceedings Spring COMPCON Conference, 1996*.

[4]. H. Balakrishnan, S. Seshan, and R.H. Katz., “Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks,” *ACM Wireless Networks,* 1(4) 1995.

[5]. G. Valko. "Cellular IP - A New Approach to Internet Host Mobility," *ACM Computer Communication Review* 1999.

[6]. S. Acampora and M. Naghshineh, “An Architecture and Methodology for Mobile-Executed Hand-off in Cellular ATM,” *IEEE Journal on Selected Areas in Communications*, 12(8), 1994.

[7]. V. Garg and J. Wilkes, *Wireless and Personal Communications Systems*. Prentice-Hall, Chapter 8, 1996.

[8]. R. Ghai and S. Singh, “An Architecture and Comm . Protocol for Picocellular Networks,” *IEEE Personal Comm. Magazine*, 1(3) 1994.

[9]. IBM Infrared Wireless LAN Adapter Technical Reference. IBM Microelectronics - Toronto Lab 1995.

[10]. J. Ioannidis and G. Q. Maguire, “The Design and Implementation of a Mobile Internetworking Architecture.” In *Proceedings of Usenix Conference*, 1993.

[11]. C. Perkins, IP Mobility Support. RFC 2002, 1996.

[12]. S. Tekinay and B. Jabbari. 1991. “Handover and Channel Assignment in Mobile Cellular Networks.” *IEEE Comm. Magazine*, 29(11), 1991.

[13]. L. Zhang, S. Deering, D. Estrin, S. Shenker and D. Zappala. “RSVP: A New Resource ReSerVation Protocol”, *IEEE Network*, 7(5) 1993.

[14]. IETF RSVP group web, http://www.ietf.org/html.charters/rsvp-charter.html

[15]. Y.B. Lin, W.R. Lai, and R.J. Chen.. “Performance Analysis for Dual Band PCS Networks,” *IEEE Tran. on Computers*, 49(2) 2000.