Performance Analysis of Handoff Techniques Based on Mobile IP, TCP-Migrate, and SIP

Shantidev Mohanty, *Member*, *IEEE*, and Ian F. Akyildiz, *Fellow*, *IEEE*

Abstract—Mobility management protocols operating from different layers of the classical protocol stack (e.g., link, network, transport, and application layers) have been proposed in the last several years. These protocols achieve different handoff performance for different types of applications. In this paper, mobile applications are grouped into five different classes, *Class A* through *Class E*, based on their mobility management requirements. Analytical models are developed to investigate the handoff performance of the existing mobility management protocols for these application classes. The analysis shows that applications of a particular class experience different handoff performance when different mobility management protocols are used. Handoff performance comparisons of different mobility management protocols are carried out to decide on the suitable mobility management protocol for a particular application class. The results of mathematical analysis advocate the use of transport layer mobility management for *Class B* and *Class C* applications, Mobile IP for non-real-time *Class D* and *Class E* applications, and Session Initiation Protocol-based mobility management for real-time *Class D* and *Class E* applications. Moreover, through analytical modeling, the parameters that influence the handoff performance of mobility management protocols are identified. These parameters can be used to design new application-adaptive techniques to enhance the handoff performance of the existing mobility management protocols.

Index Terms—Next-generation wireless systems, mobility management, handoff performance, analytical modeling.

1 Introduction

Next-generation Wireless Systems (NGWS) integrate existing wireless networks such as wireless local area networks (WLANs), third generation (3G) cellular networks, and satellite networks to realize a unified wireless communication system that has the best features of the individual networks to provide ubiquitous "always best connection" [10] to mobile users [4]. In NGWN, mobile users are connected to the best available networks that suit their service requirements and switch between different networks based on their service needs. Efficient mobility management protocols are required to support mobility across heterogeneous access networks.

Mobility management contains two components: location management and handoff management [3]. Location management enables the system to track the locations of mobile users between consecutive communications. On the other hand, handoff management is the process by which users keep their connections active when they move from one base station (BS) to another. Handoffs in wireless networks result in performance degradation to applications.

Handoff management protocols operating from different layers of the TCP/IP protocol stack (e.g., link layer, network layer, transport layer, and application layer) are proposed in the literature [4] to minimize the performance degradation during handoff. Mobile IP [17] that operates from the network layer is proposed to support mobility management in IP-based networks. It forwards packets to mobile users that are away from their home networks using IP-in-IP tunnels [17]. Transport layer mobility management protocols eliminate the need for tunneling of the data packets. TCP-Migrate [22] and an architecture called MSOCKS [14] are proposed to support transport layer handoff management. Moreover, work is going on in the IETF to modify the Stream Control Transmission Protocol [23] to allow it to dynamically change endpoint addresses in the midst of a connection [8], [11]. Application layer handoff using Session Initiation Protocol (SIP) is proposed in [25].

There exist several studies to investigate the performance of these handoff protocols [4]. However, to the best of our knowledge, there is no existing work that investigates the interaction between the handoff process and the type of application and, thereby, the effect of handoffs on different application types. In this work, we study the effect of the handoff process on different types of applications. The effect of handoffs on applications can be specified in terms of the following parameters: *handoff latency*, *packet loss*, *throughput degradation time*, *transport-layer transparency*, etc., as described in Section 2.2. To provide efficient handoff support to all application classes, a mobility management protocol must achieve good performance results for all handoff performance parameters. To understand the effect of handoffs on mobile applications, we classified different applications into five categories: *Class A* through *Class E*, based on their mobility management requirements as described in Section 2.1. Then, we carried out the qualitative analysis of the handoff performance of the existing mobility management protocols. Our analysis shows that mobility management protocols operating from different layers of the classical TCP/IP protocol stack achieve different performance results with respect to different handoff parameters. For example, while Session Initiation Protocol-based (SIP) [20] mobility management and TCP-Migrate

- *S. Mohanty is with Intel Corporation. E-mail: shantidev.mohanty@intel.com.*
- *I.F. Akyildiz is with the Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332. E-mail: ian@ece.gatech.edu.*

Manuscript received 13 Nov. 2005; revised 3 Aug. 2006; accepted 28 Aug. 2006; published online 7 Feb. 2007.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-0332-1105.
Digital Object Identifier no. 10.1109/TMC.2007.1040.

[22] achieve minimum *end-to-end delay*, TCP-Migrate and Mobile IP [17] achieve *transport-layer transparency*. On the other hand, Mobile IP introduces additional *end-to-end delay*. Similarly, SIP-based mobility management does not provide *transport-layer transparency* to the applications. Thus, none of the existing mobility management protocols can support efficient handoff management for all application classes.

To answer the question "What is the suitable mobility management protocol for a particular application class?" we developed an analytical model to study the handoff performance of the existing mobility management protocols when they are used for different application classes. The results of our analysis advocate the use of transport layer mobility management for *Class B* and *Class C* applications, Mobile IP for non-real-time *Class D* and *Class E* applications, and Session Initiation Protocol-based mobility management for real-time *Class D* and *Class E* applications. Moreover, through our analytical modeling and performance investigation, the parameters that influence the handoff performance of mobility management protocols are identified. These parameters can be used to design new application-adaptive techniques to enhance the handoff performance of the existing mobility management protocols.

The remainder of this paper is organized as follows: In Section 2, we classify different applications into five categories and carry out qualitative analysis of handoff performance of existing mobility management protocols for these application classes. We derive the basic formulations that we use in our analytical modeling in Section 3. In Section 4, we develop the analytical models to investigate the handoff performance of *Class B* and *Class C* applications when Mobile IP and TCP-Migrate are used. Then, we carry out a similar analysis for *Class D* and *Class E* applications when Mobile IP and SIP are used in Section 5. Finally, we summarize the results of our analysis and conclude the paper in Section 6.

2 Classification of Applications and Qualitative Handoff Performance Analysis of Existing Mobility Management Protocols

In this section, we first classify different applications into five classes based on their mobility management requirements. Then, we carry out a qualitative analysis of handoff performance of existing mobility management protocols for these application classes.

2.1 Classification of Applications

In NGWS, there will be different types of applications, e.g., voice, real and non-real-time data, and multimedia services. Based on their mobility management requirements, we classify these applications into the following categories:

- **Class A Applications:** TCP or UDP applications that are short lived and originated by a mobile node (MN) such as Domain Name Service (DNS) resolution [13], [22]. Here, the Correspondent Node (CN) (usually a server) typically resides in the fixed backbone network and has a permanent IP address. We can assume that the MN knows about CN's IP address in advance. Since every Internet packet includes the IP address of the sender, the CN learns about the IP address of the MN from the first IP packet that it receives from the MN. As these applications are short lived (most are over in seconds from the initial service request by the client [13], i.e., MN in this case), there is no need for handoff support. If the transaction time happens to coincide with the handoff time, it is always possible to restart the transaction after the handoff [13]. As the transactions are initiated by the MN, there is no need for the CN to learn about the current location of the MN. Therefore, these applications do not require location or handoff support.
- **Class B Applications:** TCP applications that are long lived and originated by an MN such as Web browsing and telnet sessions. These applications do not require location support as the MN initiates the connection. However, as they are long lived, they require handoff support as they may stay active over several cell transition instances. Therefore, these applications do not require location support but require handoff support.
- **Class C Applications:** TCP applications that are long lived and terminated at an MN such as telnet sessions. In this case, the originator of the application needs to learn the IP address of the MN before it can start the connection. Therefore, location support is required. Moreover, as these applications are long lived, handoff support is required. Thus, such applications require both location and handoff support.
- **Class D Applications:** UDP applications that are long lived and originated by an MN such as mobile telephony where MN is the calling party. These applications require only handoff support.
- **Class E Applications:** UDP applications that are long lived and terminated at an MN such as mobile telephony where MN is the called party. In this case, the originator of the application needs to learn the IP address of the MN before it can start the connection. Therefore, location support is required. Moreover, as these applications are long lived, handoff support is required. Thus, these applications require both location and handoff support.

As Class A applications do not require location or handoff support, we do not consider these applications in this work. Class B, Class C, Class D, and Class E applications require handoff support. Therefore, it is essential that these applications remain transparent to the handoffs. The level of transparency to handoffs that these applications can achieve depends on the mobility management protocol used to carry out the handoff.

2.2 Qualitative Handoff Performance Analysis of Existing Mobility Management Protocols

The effect of handoffs on different application classes can be specified in terms of the following parameters.

- *Handoff latency:* This is the time duration between handoff initiation and handoff completion. Real-time applications using real-time transport protocol (RTP) over UDP such as Internet telephony and multimedia applications that belong to Class D and Class E require minimum handoff latency.
- *Packet loss during handoff:* Class D and Class E applications run over UDP. As UDP is not a reliable protocol, the packets that are lost during the handoff cannot be recovered. Thus, Class D and Class E applications experience packet loss during handoffs. Class B and Class C applications run over TCP. As TCP is a reliable protocol, the packets that are lost during a handoff are recovered through TCP's retransmission mechanism. Therefore, Class B and Class C applications do not experience packet loss during a handoff.
- *Throughput degradation time:* For Class B and Class C applications that use TCP as the transport layer protocol, the packets that are lost during a handoff trigger the slow start mechanism of TCP leading to throughput degradation. It may be noted that the lost packets are retransmitted by TCP until they are received at the destination. Therefore, the applications do not experience any packet loss. However, TCP's throughout is effected negatively by these packet losses. The throughput degradation time should be kept minimum.
- *End-to-end delay:* Real-time applications require minimum end-to-end delay. When a mobility management protocol implements redirection of packets such as Mobile IP, the end-to-end delay may increase significantly. Real-time Class D and Class E applications require low end-to-end delay.
- *Transport-layer transparency:* Applications running over TCP require that if the transport layer connections are broken during a handoff, there should be a mechanism to resume them in such a way that applications remain transparent to the handoff. These include Class B and Class C applications. Therefore, mobility management protocols that hide the modifications of the IP-address of the mobile host upon handoff such as Mobile IP and TCP-Migrate are appropriate for these applications.
- *Security:* A particular application may have different levels of security requirements in different network environments. For example, while communicating inside a home network domain, an application does not require strict security mechanisms. On the other hand, while in a foreign domain or while communicating with CNs that are in foreign domains, the same application may require strict security mechanisms. Thus, security is important for all classes of applications.

The above analysis shows that different classes of applications have different expectations from a mobility management protocol. Mobility management protocols operating from different layers such as link layer [2], [15], network layer [17], transport layer [22], and application layer [25] are proposed in the literature [3], [4]. Next, we carry out the qualitative handoff performance evaluation of these protocols for the above handoff performance metrics.

2.2.1 Link Layer (Layer 2) Mobility Management Protocols

Link layer mobility management protocols focus on the issues related to intersystem roaming between heterogeneous access networks with different radio technologies and different network management techniques [4]. The user mobility profile (UMP) is used in [2] to support enhanced mobility management. The concept of intersystem boundary cells is used in [15] to prepare the users for a possible intersystem handoff in advance. Thus, a significant reduction of the intersystem handoff failure probability is achieved. The performance of the link layer mobility protocols is summarized as follows:

- The intersystem handoff latency is high because several functions such as *format transformation and address translation*, *user profile retrieval*, *mobility information related to intersystem movement recording*, and *authentication between systems* are carried during an intersystem handoff [4].
- The large value of handoff latency results in higher packet loss during intersystem handoff.
- After the intersystem handoff, an MN communicates with the new system without the need for any redirection agent. Thus, the end-to-end delay requirement of the applications is respected.
- Since an MN communicates with a new address in the new system, a transport layer connection has to be reestablished after intersystem handoff. Therefore, link-layer mobility management protocols are not transparent to TCP and UDP applications.
- As authentication is carried out during an intersystem handoff, these handoffs are secure.

2.2.2 Network Layer (Layer 3) Mobility Management Protocols

Handoff performance of network layer mobility protocol, Mobile IP [17], is summarized below:

- Mobile IP registration introduces a significant amount of latency during handoff. Hierarchical Mobile IP [9] and other micro-mobility protocols such as Cellular IP [24], IDMP [16], and HAWAII [19] reduce the handoff latency by introducing another layer of hierarchy to the base Mobile IP architecture to localize the signaling messages to one domain.
- The large value of Mobile IP latency results in significant packet losses during a handoff.
- Mobile IP triangular routing results in path asymmetry between a CN and an MN. Additional delay is introduced from the CN to MN path because of packet redirection through the home agent (HA). Measurements in [27] show that Mobile IP increases the end-to-end delay by 45 percent within a campus (from a CN to an MN), which can be expected to increase further in wide area networks. This is not acceptable for delay-sensitive applications [25].
- Through packet redirection during handoff, Mobile IP hides the change of IP address from the applications. Therefore, Mobile IP handoff is transparent to the applications and the transport layer connections are kept intact during a handoff.

- Authentication of Mobile IP registration messages is carried out as a part of the Mobile IP registration [18]. Thus, Mobile IP handoff is secure.

2.2.3 Transport Layer (Layer 4) Mobility Management Protocols

Using transport layer mobility, a TCP peer can suspend an open connection and reactivate it from another IP address. This reactivation of the TCP connection is carried out in such a way that the applications can continue to use an established TCP connection across a handoff [22]. Handoff performance of transport layer mobility management protocols is summarized below:

- Since only the communicating end points are involved in the handoff process, the latency is often lower than that of Mobile IP [22]. It may be noted that the use of a third party, such as an HA in the case of Mobile IP, increases the handoff latency.
- During a transport layer mobility, a TCP connection maintains the same control block and state including the sequence number space [22]. Therefore, any necessary retransmissions can be requested in the standard fashion. Thus, the packets that are lost during the handoff can be recovered. Therefore, transport layer mobility management protocols can be designed to realize zero packet losses during a handoff.
- Since there is no packet redirection, the path between the communicating hosts (i.e., the MN and the CN) is symmetric. Therefore, the end-to-end delay does not increase after handoff. This is in contrast to network layer Mobile IP handoff, where, due to triangular routing, the end-to-end delay increases in the CN to MN path when the MN is away from its home network.
- As a transport layer connection is reactivated upon handoff, the applications remain transparent to mobility.
- Authentication is implicitly included during a transport layer mobility, making it highly secure. The end-to-end approach to mobility simplifies the trust relationships required to securely support end-host mobility compared to the network layer approaches such as Mobile IP [22]. Since no third parties are required or even authorized to speak on the mobile host's behalf in an end-to-end mobility approach, the only trust relationship required for secure relocation is between the MN and the CN [22].

2.2.4 Application Layer (Layer 5) Mobility Management Protocols

Handoff performance of application layer mobility protocol, SIP [25], is summarized below:

- Because redirecting agents such as SIP proxies and SIP redirect servers are used during handoff, the handoff latency of SIP is comparable to that of Mobile IP but is higher than the transport layer mobility protocols.
- The packets that are in transit during the handoff signaling procedures are lost, making handoff packet loss comparable to that of Mobile IP handoff.

TABLE 1
Qualitative Performance of Mobility Management Protocols

Performance parameter	Layer 2	Layer 3	Layer 4	Layer 5
Handoff latency	Worst	Worse	Weak	Worse
Handoff packet loss	Worst	Worse	Weak	Worse
End-to-end delay	Good	Weak	Good	Good
Transport-layer transparency	Weak	Good	Good	Weak
Security	Good	Good	Good	Good

- Once the handoff signaling phase is over, the communicating hosts, i.e., the CN and the MN, communicate directly without any redirection agent. Therefore, end-to-end delay does not increase when a MN is away from its home network.
- SIP cannot support TCP connections [25]. Therefore, SIP mobility is not transparent to TCP protocol.
- Signaling messages that are used during SIP mobility management are secured using different security mechanisms. Thus, SIP-based mobility management is secure.

We summarize the performance of the mobility management protocols operating from different layers of the TCP/IP protocol stack in Table 1, which shows that none of the existing mobility management protocols can support mobility management transparent to all types of applications. Since it is not possible to support transparent mobility management for all types of applications using one particular mobility management protocol in next-generation wireless systems, we advocate the use of a mobility management framework that adaptively selects a mobility management protocol based on applications' requirements. The link layer mobility management protocols alone cannot be used in NGWS because of their inherent scope limitation to a single wireless access technology [4]. Because of the intrinsic technology heterogeneity of different wireless networks, mobility management protocols supporting mobility outside the scope of a particular access technology are suitable for NGWS. These include mobility management protocols operating from network, transport, and application layers. To determine the mobility management protocol that is suitable for a particular class of application, in the next section, we develop analytical models to investigate the handoff performance of the existing mobility management protocols in the context of *Class B*, *Class C*, *Class D*, and *Class E* applications. As mentioned before, *Class A* applications do not require any mobility support.

3 Parameters and Basic Derivations for Analytical Modeling

To develop analytical modeling for the performance analysis of the existing mobility management protocols, we consider that a mobile host (MH[1]) that is away from its home network (HN) moves from an Old Network (ON) to a New Network (NN) in the middle of its communication with a Correspondent Host (CH) as shown in Fig. 1. The

1. MH and CH (Correspondent Host) are synonymous with MN (Mobile Node) and CN (Correspondent Node), respectively.

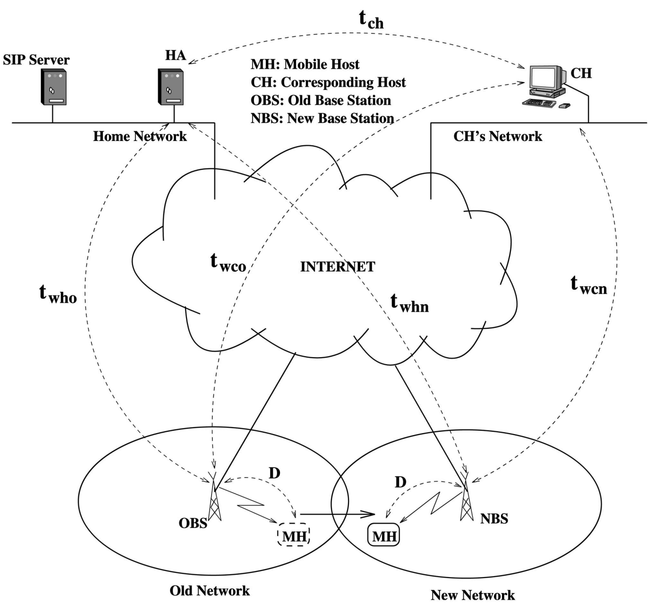


Fig. 1. Reference network model.

network entities that assist the MH for its mobility management such as an SIP [20] server, a Domain Name Server (DNS), and a home agent (HA) are located in the HN as shown in Fig. 1. We define the following parameters that are shown in Fig. 1. t_{ch} is the one-way delay between the CH and the HA. t_{ho} is the one-way delay between the MH and its HA when the MH is in the ON. $t_{ho} = D + t_{who}$, where D is the link-layer access delay as defined in (5) and t_{who} is the one way delay in the wired network between the OBS and the HA. t_{hn} is the one-way delay between the MH and its HA when the MH is in the NN. $t_{hn} = D + t_{whn}$, where t_{whn} is the one-way delay in the wired network between the NBS and the HA. t_o is the one-way delay between the MH and the CH while the MH is in the ON. $t_o = D + t_{wco}$, where t_{wco} is the one-way delay in the wired network between the OBS and the CH. t_n is the one way delay between the MH and the CH while the MH is in the NN. $t_n = D + t_{wcn}$, where t_{wcn} is the one-way delay in the wired network between the NBS and the CH. Next, we carry out some basic derivations that we use for our analytical modeling in the remaining part of this paper.

3.1 End-to-End Packet Loss Probability

The path between the MH and the HA (or the CH) contains two parts: the wireless link connecting the MH and the BS and the wired link between the BS and the HA (or the CH). Then, the end-to-end packet loss probability p between the MH and the HA (or the CH) is given by

$$p = 1 - (1 - p_w)(1 - p_c), \tag{1}$$

where p_w and p_c are the packet loss probabilities in the wireless link and the wired link, respectively.

Next, we derive the expressions for p for both without Radio Link Protocol (RLP) and with RLP scenarios. We denote by L_p and L_f the length of a packet (typically an IP packet) and the length of a link-layer frame, respectively. Therefore, the number of frames per packet is $K = \lceil \frac{L_p}{L_f} \rceil$. For the case of without RLP, the packet loss probability in the wireless link becomes $p_{wnr} = 1 - (1 - p_f)^K$, where p_f is the link layer frame error rate (FER). Therefore, the end-to-end packet loss probability p_{nr} between the MH and the HA (or the CH) without RLP can be derived by using $p = p_{nr}$ and $p_w = p_{wnr}$ in (1),

$$p_{nr} = 1 - (1 - p_f)^K (1 - p_c). \tag{2}$$

For the case with RLP, the packet loss probability in the wireless link p_{wr} is given by [6]

$$p_{wr} = 1 - \left[1 - p_f((2 - p_f)p_f)^{\frac{(n^2+n)}{2}}\right]^K, \tag{3}$$

where n is the maximum number of trials that the RLP carries out before aborting the attempt to transmit a frame over the link layer. Typically, $n = 3$ for RLP. The end-to-end packet loss probability p_r between the MH and the HA (or the CH) with RLP is obtained from (1) by using $p_w = p_{wr}$ and $p = p_r$,

$$p_r = 1 - \left[1 - p_f((2 - p_f)p_f)^{\frac{(n^2+n)}{2}}\right]^K (1 - p_c), \tag{4}$$

where p_f is the link layer FER and K is the number of link layer frames per packet.

3.2 End-to-End Packet Transportation Delay

The end-to-end packet transportation delay between the MH and the HA (or the CH) is the sum of packet transportation delay over the wireless link from the MH to the BS and the packet transportation delay in the wired link between the BS and the HA (or the CH). When no RLP is used, there is no frame retransmission in the link layer. Therefore, the end-to-end packet transportation delay, T_{nr}, between the MH and the HA (or the CH) is given by

$$T_{nr} = D + t_w, \tag{5}$$

where D is the link-layer access delay and t_w is the delay in the wired link between the BS and the HA (or the CH).

The one-way frame transportation delay T_f between the MH and the BS with RLP is given by [6]

$$T_f = D(1 - p_f) + \sum_{i=1}^{n} \sum_{j=1}^{i} P(C_{i,j})(2iD + 2(j - 1)\tau), \tag{6}$$

where p_f is the link layer FER and τ is the link layer interframe interval, which is typically around 20 ms. $P(C_{i,j})$ is the probability that the first frame transmitted by the MH is received correctly by the BS, being the ith retransmitted frame at the jth retransmission trial. The expression for $P(C_{i,j})$ is given by [6]

$$P(C_{i,j}) = p_f(1 - p_f)^2((2 - p_f)p_f)^{\left(\frac{i^2-i}{2}+j-1\right)} \quad \text{for } i = 1, 2, \ldots, n \text{ and } j = 1, 2, .., i. \tag{7}$$

Therefore, when RLP is used, the end-to-end packet transportation delay, T_r, between the MH and the HA (or the CH) is then

$$T_r = T_f + (K - 1)\tau + t_{w,} \tag{8}$$

where K is the number of link layer frames per packet as defined in Section 3.1.

3.3 Average Signaling Packet Transportation Delay Using UDP

When UDP, which does not support reliable packet transport, is used to transport signaling packets, the sender starts a retransmission timer. If the sender does not receive a reply for its transmitted packet, it retransmits the packet when the retransmission timer expires. (It may be noted that this retransmission of the signaling packet is different from TCP's retransmission, where the retransmissions are handled by the transport layer.) Thus, the average one-way signaling packet transportation delay, D_p, between the MH and the HA (or the CH) is

$$D_p = \sum_i^{\infty} p_i T_i, \tag{9}$$

where T_i is the packet transportation delay when the packet is successfully transferred between the MH and the HA (or the CH) in the ith retransmission trial and p_i is the probability that a packet is successfully transferred between the MH and the HA (or the CH) in the ith retransmission trial. p_i is computed by

$$p_i = q^{i-1}(1-q), \tag{10}$$

where q is the end-to-end packet loss probability between the MH and the HA (or the CH). $q = p_{nr}$ when no RLP is used and $q = p_r$ when RLP is used. The expressions for p_{nr} and p_r are derived in (2) and (4), respectively. The formulation for T_i is as follows:

$$T_i = \begin{cases} \Delta + \gamma\Delta + \gamma^2\Delta + \ldots + \gamma^{i-2}\Delta + B & i \leq m \\ \Delta + \gamma\Delta + \gamma^2\Delta + \ldots + \gamma^{m-2}\Delta & \\ \quad +(i-m)\gamma^{m-2}\Delta + B & i > m, \end{cases} \tag{11}$$

where the special cases are $T_1 = B$ and $T_2 = \Delta + B$; m is an integer such that, after the mth retransmission timeout, the retransmission timer is frozen. $B = T_{nr}$ when no RLP is used and $B = T_r$ when RLP is used. The expressions for T_{nr} and T_r are derived in (5) and (8), respectively. Δ is the initial value of the retransmission timer, which is large enough to account for the size of the messages, twice the round trip time between the MH and the HA (or the CH), and at least an additional 100 ms to allow for processing the messages at the MH and the HA (or the CH). γ is the factor by which the retransmission timeout duration is incremented after each failed retransmission. Typically, $\gamma = 2$.

Now, using the formulations for p_is and T_is from (10) and (11), respectively, we simplify (9) to obtain

$$\begin{aligned} D_p &= \sum_i^{\infty} p_i T_i = p_1 T_1 + \sum_{i=2}^{m} p_i T_i + \sum_{i=m+1}^{\infty} p_i T_i \\ &= (1-q)B + \sum_{i=2}^{m} q^{i-1}(1-q) \\ &\quad [\Delta + \gamma\Delta + \gamma^2\Delta + \ldots + \gamma^{i-2}\Delta + B] + \sum_{i=m+1}^{\infty} q^{i-1}(1-q) \\ &\quad [\Delta + \gamma\Delta + \gamma^2\Delta + \ldots + \gamma^{m-2}\Delta + (i-m)\gamma^{m-2}\Delta + B] \\ &= (1-q)\Big\{B + A\sum_{i=2}^{m} q^{i-1}(\gamma^{i-1}-1) + \sum_{i=m+1}^{\infty} q^{i-1} \\ &\quad [A(\gamma^{m-1}-1) + (i-m)\gamma^{m-2}\Delta\Big\}, \end{aligned} \tag{12}$$

where $A = \frac{\Delta}{\gamma - 1}$.

3.4 TCP Retransmission Timeout Duration

TCP maintains a retransmission timer, whose duration is equal to TCP's retransmission timeout (RTO), for every packet that it sends. If it does not receive the ACK for a packet before the expiry of the packet's retransmission timer, TCP retransmits the packet. After retransmitting the lost packet, TCP increases the RTO duration by a factor of λ and waits for the ACK. If the timer for the retransmitted packet is also lost, then TCP again retransmits the packet and increases the RTO duration by a factor of λ. When the number of retransmissions for a packet becomes higher than a predefined number s, TCP does not increase its RTO. TCP continues this behavior until the packet is received correctly at the destination. If the lost packet is received by at the destination after the Nth retransmission, then the time difference between the first transmission of the packet to its Nth retransmission is

$$\begin{aligned} T_{rto} &= \begin{cases} TO_1 + \lambda TO_1 + \ldots.. + \lambda^N TO_1 & \text{if } N \leq s \\ TO_1 + \lambda TO_1 + \ldots.. + \lambda^p TO_1 & \text{if } N > s \\ \quad +(N-s)\lambda^s TO_1 & \end{cases} \\ &= \begin{cases} TO_1 \frac{\lambda^{N+1}-1}{\lambda - 1} & \text{if } N \leq s \\ TO_1 \frac{\lambda^{s+1}-1}{\lambda-1} + (N-s)\lambda^s TO_1 & \text{if } N > s, \end{cases} \end{aligned} \tag{13}$$

where TO_1 is the initial RTO.

3.5 Time for TCP Slow Start

In slow start, TCP starts from a initial congestion window size and gradually increases its congestion window to the steady state value, CW_s. We assume that the initial congestion window size is 1. TCP doubles its congestion window after every round trip time (RTT). Assuming that there is no packet loss before TCP reaches its steady state, if the total number of round trips for TCP to reach its steady state from slow start is i, then

$$CW_s = 1 + 2 + 2^2 + 2^3 + \ldots.. + 2^i = 2^{(i+1)} - 1. \tag{14}$$

Therefore, $i = log_2(1 + CW_s) - 1$. The time required for TCP to reach its steady state is

$$T_s = [log_2(1 + CW_s) - 1]RTT. \tag{15}$$

4 Handoff Performance of *Class B* and *Class C* Applications (Mobile IP and TCP-Migrate)

As *Class B* and *Class C* applications use TCP, we consider a TCP connection between a CH and MH to investigate their handoff performance. The handoff performance of *Class B* and *Class C* applications is synonymous with the handoff performance of a TCP connection. We consider a scenario where the MH while in the Old Network (ON) starts to download a file using FTP from the CH and moves into the New Network (NN) in the middle of this file transfer. We assume that the size of the file is long enough for the TCP connection to continue from the ON to the NN. We further assume that CH's FTP application creates packets continuously such that CH's TCP sends full-sized segments (packets) as fast as its congestion window allows. Moreover, we assume that the window size advertised by the receiver (the MH in this case) is always larger than the congestion window size. Therefore, the sending window size is always limited by the congestion window. We assume that, while the MH is in the ON, the TCP connection between the CH and the MH operates in a steady state. During this steady state, TCP state parameters, e.g., congestion window size and round trip time (RTT), are decided by the path between the CH and the MH. To maintain the highest throughput performance in different types of wireless networks characterized by different p_f and D and achieve fairness to the wired TCP sources sharing the same bottleneck, we consider the adaptive congestion control proposed in [1] that dynamically adjusts additive-increase multiplicative-decrease (AIMD) parameters (α, β) according to the current wireless link conditions. The expression for α is given by [1]

$$\alpha = \frac{bp(1-\beta)}{2(1+\beta)}\left[\hat{T}(2R + 3T_0 p(1+32p^2)(1+\beta))\right]^2, \qquad (16)$$

where p is the end-to-end packet loss probability and $\hat{T}$ is the throughout achieved by a wired TCP source experiencing p_c, which is the packet loss probability due to congestion in the wired network, and R_c, which is the end-to-end RTT in the wired network. R is the end-to-end RTT between the CH and the MH, T_0 is the initial retransmission timeout (RTO) for the TCP connection, and b is the number of data packets acknowledged with a single ACK. The numerical value of β can be set to be 0.75, 0.80, and 0.85 for a WLAN, a 3G cellular network, and a satellite network, respectively [1]. p is the end-to-end packet loss probability between the MH and the CH. The steady state congestion window size of TCP depends on the end-to-end packet loss probability and is given by [26]

$$E[W] = \frac{\alpha + b(1-\beta)}{2b(1-\beta^2)} + \sqrt{\left(\frac{\alpha + b(1-\beta)}{2b(1-\beta^2)}\right)^2 + \frac{2\alpha(1-p)}{bp(1-\beta^2)}}. \qquad (17)$$

We use (16) to determine the additive-increase parameter of a TCP connection and (17) to calculate the steady state congestion window size when the MH is in the ON and the NN.

As TCP is a reliable protocol, there is no packet loss during a handoff as lost packets are recovered through retransmissions after the handoff is completed. Therefore, the handoff performance of a TCP connection can be represented by two parameters: 1) throughput degradation time and 2) handoff latency. Next, we investigate the performance of a TCP connection when Mobile IP [17] is used as the mobility management protocol followed by when TCP-Migrate is used.

4.1 Handoff Performance Analysis of a TCP Connection When Mobile IP Is Used

Fig. 2 shows the Mobile IP [17] handoff process for a TCP connection when the MH moves from the ON to the NN. As shown in Fig. 2, the HA intercepts the packets for the MH. Then, the HA tunnels the packets to the MH. The parameters t_{ch}, t_{ho}, t_{hn}, t_{hn}, and t_n shown in Fig. 2 are defined in Section 3. In Fig. 2, time A is the time the MH enters the NN and starts the layer 2 handoff (L2 handoff) to the NN. As pointed out earlier, before time A, the TCP connection operates in the steady state corresponding to the ON. We denote the congestion window size of this steady state as CW_1. We assume that all packets received by the MH before time A are properly ACKed and all these ACKs are received by the CH. We denote the sequence number of the packet received at time A as n. Therefore, the MH is expecting the packet with sequence number $n+1$ next. As shown in Fig. 2, the MH starts layer 2 handoff to the NN and IP address acquisition from the NN at time A. These procedures are completed at time B. Then, at time B, the MH starts Mobile IP [17] registration with its HA. The new care-of-address (CoA) of the MH gets successfully registered at the HA at time instant C. Thus, packets received by the HA after time C are correctly forwarded to the MH in the NN. The packets received by MH's HA from the CH between time G and C are lost as they are forwarded to MH's old CoA. The last ACK sent by the MH from the ON is received by the CH at time E. Therefore, the CH transmits all packets in its congestion window, i.e., CW_1 number of packets, after E and waits for ACKs. One of the following scenarios may occur:

- *Case A:* The new CoA of the MH is registered at the HA after the HA receives the packet transmitted by the CH at time F. In this case, all packets in the congestion window (from E to F) are lost as the HA tunnels these packets to MH's old CoA. Therefore, the CH does not receive the ACKs for these packets and waits until the RTO of the packet transmitted at time E to occur. Then, it reduces the congestion window to one and retransmits the packet for which RTO occurs at time RTO_1. If the new CoA of the MH is not registered at the HA by the time this retransmitted packet reaches the HA, the HA sends the packet to MH's old CoA and the packet is lost again. Then, CH's TCP updates the value of RTO, waits until the second RTO, and retransmits the packet when the second RTO expires. If the retransmitted packet after the Nth RTO reaches the HA after MH's new CoA is registered at the HA, then the HA tunnels the packet to MH's new CoA. In this case, the retransmitted packet is successfully received by the MH in the NN.
- *Case B:* The new CoA of the MH is registered at the HA before the HA receives the packet transmitted by the CH at time F. In this case, the packets that belong to the congestion window (from E to F) and arrive after the registration of MH's new IP address at the HA are tunneled to MH's new CoA. TCP takes one RTT to transmit all the segments in one congestion window. Typically, the Mobile IP handoff latency is larger than the RTT. Therefore, this case occurs very rarely.

We determine the handoff latency and throughput degradation time of a TCP connection for Case A as described below.

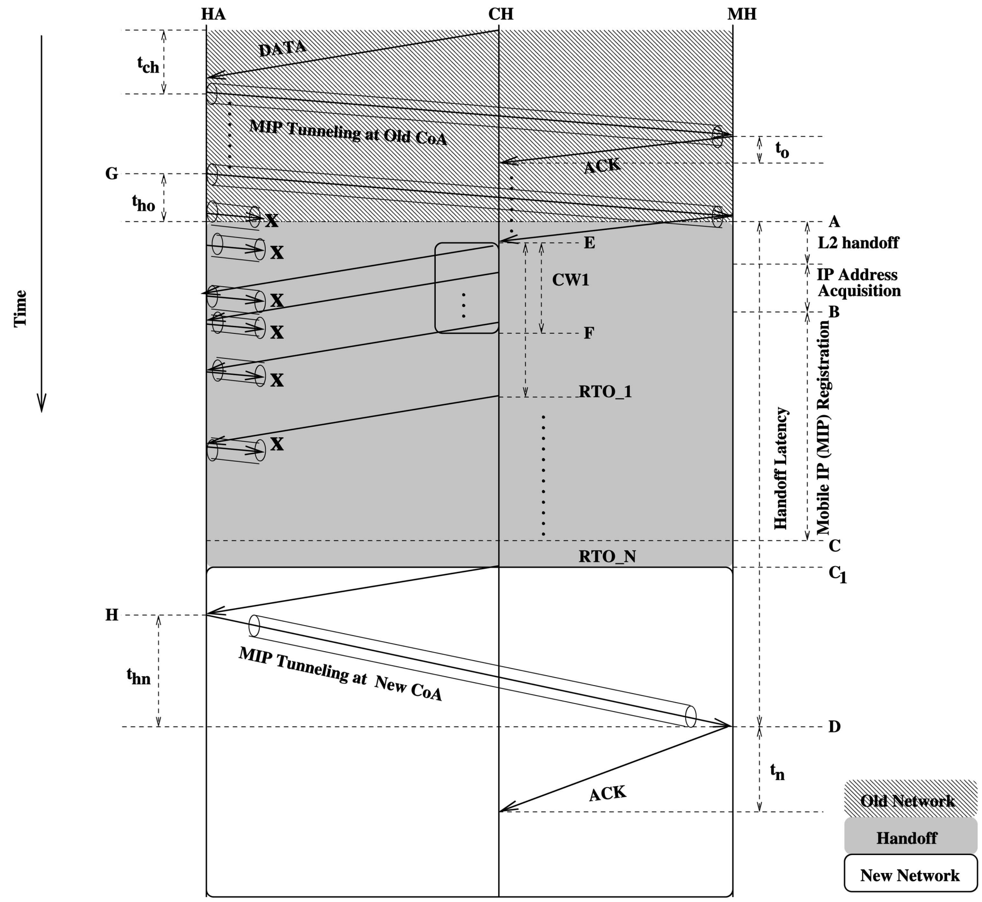


Fig. 2. Handoff of a TCP connection using Mobile IP.

4.1.1 Handoff Latency

To calculate handoff latency (the time interval between the receipt of the packet with sequence number n by the MH in the ON and the receipt of the packet with sequence number $n+1$ by the MH in the NN), we first determine the time during which the packets transmitted by the CH are lost. This time is given by

$$T = C - A = \tau_{L2} + \tau_a + \tau_m, \tag{18}$$

where τ_{L2} is the time required for MH's Layer 2 handoff to the NN, τ_a is the time required for new IP address acquisition by the MH in the NN, and τ_m is the time required for Mobile IP [17] registration. If T is such that the sending window of CH's TCP is exhausted (which is usually the case), then CH's TCP goes through timeouts as described earlier. If N number of timeouts occur before the HA receives the new CoA of the MH, then the handoff latency T_{h1} is given by

$$T_{h1} = D - A = D - C_1 + C_1 - A. \tag{19}$$

$C_1 - A$ depends on the number of TCP timeouts N that occur before the HA receives MH's new CoA. From Fig. 2, $C_1 - A = C_1 - E + E - A$, where $C_1 - E$ is the time TCP spent in N timeouts and $E - A = t_o$. Therefore, using (13), $C_1 - A$ is

$$C_1 - A = \begin{cases} TO_1 \frac{\lambda^{N+1}-1}{\lambda-1} + t_o & \text{if } N \leq s \\ TO_1 \frac{\lambda^{s+1}-1}{\lambda-1} + (N-s)\lambda^s TO_1 + t_o & \text{if } N > s. \end{cases} \tag{20}$$

Now, using (20), (19) can be expressed as

$$T_{h1} = \begin{cases} TO_1 \frac{\gamma^{N+1}-1}{\gamma-1} + t_o + t_{ch} + t_{hn} & \text{if } N \leq s \\ TO_1 \frac{\gamma^{s+1}-1}{\gamma-1} + (N-s)\gamma^s TO_1 & \\ \quad + t_o + t_{ch} + t_{hn} & \text{if } N > s, \end{cases} \tag{21}$$

where TO_1 is the initial RTO for the TCP connection when the MH is in the ON and is given by $TO_1 = \xi RTT_o$, where ξ is a constant weighting factor and RTT_o is the RTT of the TCP connection when the MH is in the ON. s and λ are defined in (13).

T in (18) is required to determine the number of retransmission timeouts, N, that CH's TCP undergoes before the HA receives the new CoA of the MH. Once N is determined, the handoff latency can be calculated using (21). τ_{L2} and τ_a in (18) are usually constant for a particular wireless system such as a WLAN, 3G, satellite network, etc. On the other hand, τ_m depends on the distance between the MH and its HA and on the wireless link conditions.

We derive the expression for τ_m as follows: τ_m is equal to the time required for MH's *Mobile IP Registration Request* [17] message to reach the HA and HA's *Mobile IP Registration Reply* [17] to reach the MH, i.e., $\tau_m = 2D_{mh}$, where D_{mh} is the average one-way delay to transport Mobile IP signaling packets between the MH and the HA. Note that Mobile IP signaling messages are transported using UDP [17]. Using steps similar to the derivation of (12), D_{mh} is given by

$$D_{mh} = (1 - q_1)\left\{B_1 + A_1 \sum_{i=2}^{m} q_1^{i-1}(\gamma^{i-1} - 1) + \sum_{i=m+1}^{\infty} q_1^{i-1}[A_1(\gamma^{m-1}-1) + (i-m)\gamma^{m-2}\Delta_1]\right\}, \tag{22}$$

where B_1 is the end-to-end packet transportation delay between the MH and the HA. $B_1 = B_{1nr}$ when no RLP is used and $B_1 = B_{1r}$ when RLP is used. B_{1nr} is computed from (5) by using $T_{nr} = B_{1nr}$ and $t_w = t_{whn}$. B_{1r} is computed from (8) by using $T_r = B_{1r}$, $K = K_m$, and $t_w = t_{whn}$. $K_m = \lceil \frac{L_m}{L_f} \rceil$ is the number of link layer frames per one *Mobile IP Registration Request/Reply* message, where L_m is the length of a *Mobile IP Registration Request/Reply* message and L_f is the length of a link-layer frame. t_{whn} is defined in Section 3. q_1 is the end-to-end packet loss probability between the MH and the HA. $q_1 = q_{1nr}$ when no RLP is used and $q_1 = q_{1r}$ when RLP is used. q_{1nr} is computed from (2) by using $p_{nr} = q_{1nr}$ and $K = K_m$. q_{1r} is computed from (4) by using $p_r = q_{1r}$ and $K = K_m$. Δ_1 is the initial value of the retransmission timer for Mobile IP signaling messages. γ and m are as defined in (11).

4.1.2 Throughput Degradation Time

As discussed earlier, the HA receives the Nth retransmission packet after the successful registration of MH's new CoA. Therefore, the HA tunnels the Nth retransmitted packet and subsequent packets transmitted by CH's TCP to MH's new CoA. CH's TCP resumes TCP slow start operation at time C_1 as shown in Fig. 2. Then, it increases the congestion window to the steady state value of the NN denoted by CW_2. Using (15), the time required by TCP to increase its congestion window size from 1 to CW_2, τ_s, is given by

$$\tau_s = [log_2(1 + CW_2) - 1]RTT_n, \tag{23}$$

where RTT_n is the RTT when the MH is in the NN. The time for which the TCP connection experiences throughput degradation T_{t1} is equal to $T_t = (C_1 - A) + \tau_s$. Using (20) and (23), the expression for T_{t1} is

$$T_{t1} = \begin{cases} TO_1 \frac{\lambda^{N+1}-1}{\lambda-1} + t_o & \text{if } N \le s \\ +[log_2(1 + CW_2) - 1]RTT_n & \\ TO_1 \frac{\lambda^{s+1}-1}{\lambda-1} + (N-s)\lambda^s TO_1 & \\ +t_o + [log_2(1 + CW_2) - 1]RTT_n & \text{if } N > s. \end{cases} \tag{24}$$

4.2 Handoff Performance Analysis of a TCP Connection When TCP-Migrate Is Used

We select TCP-Migrate [22] as the representative transport layer mobility management protocol as it requires minimum change in the network infrastructure, whereas other solutions such as MSOCKS [14] require the introduction of an additional network entity such as a proxy to split the TCP connection [5]. Next, we briefly explain the operation of TCP-Migrate [22] during a handoff.

The MH and the CH negotiate a *token* through the Migrate option as described in [22] during the initial TCP connection establishment. Thus, a TCP connection can be uniquely identified at the MH and the CH by either

$$<MH\text{'s address}, MH\text{'s port}, CH\text{'s address}, CH\text{'s port}>$$

4-tuple or a new $<CH\text{'s address}, CH\text{'s port}, token>$ triple [22]. When the MH moves to the NN and receives a new IP address, it sends a SYN segment containing its new IP address and a Migrate Option to the CH. This SYN segment includes the *token* computed during the initial connection establishment in the Token field. The CH identifies the connection corresponding to this *token* and changes the address and port to match MH's new IP address. Then, the CH resets the congestion-related states of the connection to the initial values and resumes the connection from the slow start operation of TCP. Further details about the operation of TCP-Migrate can be found in [22].

Fig. 3 shows the TCP-Migrate handoff process of a TCP connection. At time A, the MH starts the handoff process to the NN. We assume that all packets received by the MH before time A are properly ACKed and all of them are received by the CH. We denote the sequence number of the packet received at time A as n. Therefore, the MH is expecting the packet with sequence number $n + 1$ next. As shown in Fig. 3, the MH starts layer 2 handoff to the NN and IP address acquisition from the NN at time A. These procedures are completed at time B. Then, the MH starts the TCP-Migrate handoff process that is completed at time C_1. Then, the CH resumes the TCP connection from slow start at time C_1 as shown in Fig. 3. The slow start ends at time D, i.e., the TCP connection reaches the steady state corresponding to the NN. We determine the handoff latency and throughput degradation time of the TCP connection as described below.

4.2.1 Handoff Latency

From Fig. 3, the TCP-Migrate handoff latency, T_{h2}, is given by

$$T_{h2} = C - A = \tau_{L2} + \tau_a + E[L] + t_n, \tag{25}$$

where τ_{L2} and τ_a are defined in (18). t_n is defined in Section 3. $E[L]$ is the average delay for the transportation of TCP-Migrate signaling messages. Next, we derive the expression for $E[L]$.

First, the MH sends a SYN packet with TCP-Migrate options containing the MH's new IP address to the CH $i \ge 0$ times unsuccessfully until the $(i+1)$th SYN arrives successfully at the CH. Then, the CH repeatedly retransmits its SYN/ACK until it receives an ACK from the MH. Let CH send SYN/ACK $j \ge 0$ times unsuccessfully and the $(j+1)$th SYN/ACK successfully arrive at the MH. Then, the MH retransmits the ACK to the CH that gets successfully transmitted in the $(k+1)$th trial $(k \ge 0)$. Therefore, the probability $P_h(i,j,k)$ that the TCP-Migrate handoff is completed after the exchange of i unsuccessful SYNs, followed by one successful SYN, followed by exactly j SYN/ACK failures, followed by one successful SYN/ACK, followed by k unsuccessful ACKs, followed by one successful ACK, is given by

$$P_h(i,j,k) = p_1^i(1-p_1)p_2^j(1-p_2)p_2^k(1-p_2) \quad \text{for } i,j,k = 0,1,2,\ \ldots, N_m - 1, \tag{26}$$

where N_m is such that TCP abort connection establishment attempts after N_m number of retransmissions. p_1 is the end-to-end packet loss probability between the MH and the CH

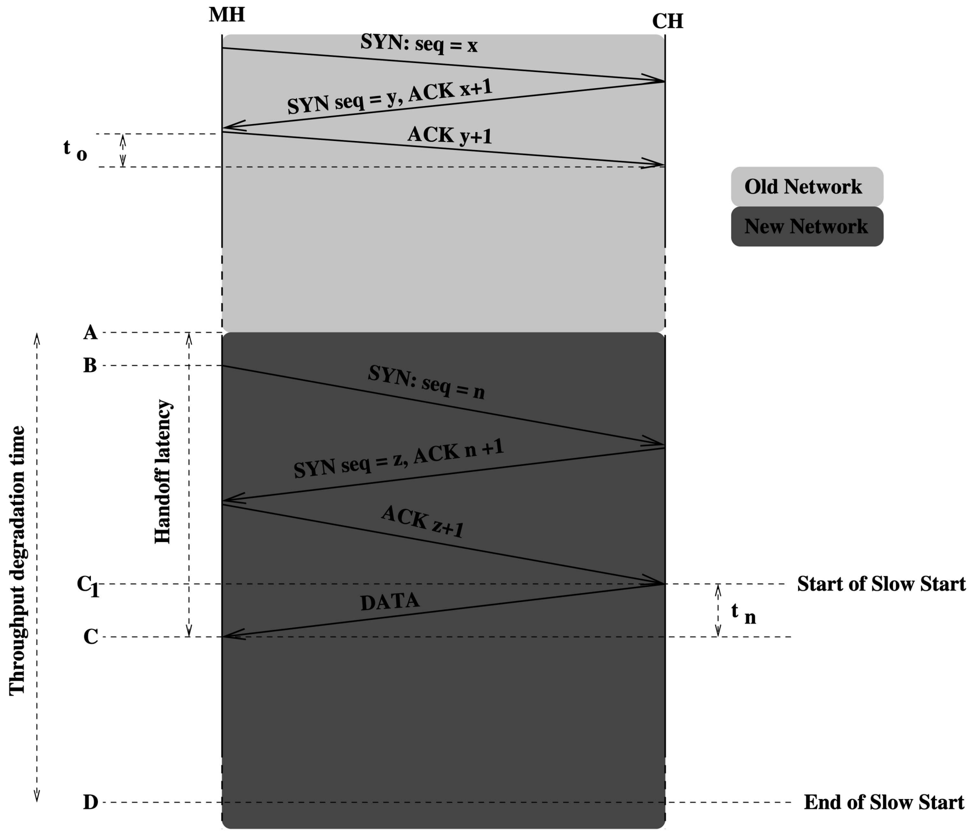


Fig. 3. Diagram showing the operation of TCP-Migrate.

for a SYN packet and p_2 is the end-to-end packet loss probability between the MH and the CH for a SYN/ACK or ACK packet. $p_1 = p_{1nr}$ when no RLP is used and $p_1 = p_{1r}$ when RLP is used. p_{1nr} is computed from (2) by using $p_{nr} = p_{1nr}$ and $K = K_1$. p_{1r} is computed from (4) by using $p_r = p_{1r}$ and $K = K_1$. $K_1 = \lceil \frac{L_1}{L_f} \rceil$ is the number of link layer frames per one SYC packet. L_1 is the length of the SYC packet and L_f is the length of a link-layer frame. Similarly, $p_2 = p_{2nr}$ when no RLP is used and $p_2 = p_{2r}$ when RLP is used. p_{2nr} is computed from (2) by using $p_{nr} = p_{2nr}$ and $K = K_2$. p_{2r} is computed from (4) by using $p_r = p_{2r}$ and $K = K_2$. $K_2 = \lceil \frac{L_2}{L_f} \rceil$ is the number of link layer frames per one SYN/ACK or ACK packet. L_2 is the length of the SYN/ACK or ACK packet. The handoff latency for the above scenario is given by

$$\begin{aligned} L_h(i,j,k) &= 1.5RTT_n + \sum_{m=0}^{i-1} 2^m RTO + \sum_{m=0}^{j-1} 2^m RTO \\ &\quad + \sum_{m=0}^{k-1} 2^m RTO \\ &= 1.5RTT_n + (2^i + 2^j + 2^k - 3)RTO \\ &\quad \text{for } i,j,k = 0,1,2,\ldots,N_m - 1, \end{aligned} \tag{27}$$

where RTO is the initial retransmission time out for the TCP connection, $RTO = \xi RTT_o$, and RTT_o is the RTT in the ON. Therefore, the average TCP-Migrate handoff latency is

$$E[L] = \sum_{i=0}^{N_m-1} \sum_{j=0}^{N_m-1} \sum_{k=0}^{N_m-1} P_h(i,j,k) L_h(i,j,k). \tag{28}$$

4.2.2 Throughput Degradation Time

As shown in Fig. 3, TCP resumes slow start operation at time C_1 and reaches its steady state operation in the NN at time D. Therefore, the TCP connection experiences throughput degradation from time A to D. Using (15) and (25), the expression for handoff degradation time, T_{t2}, is given by

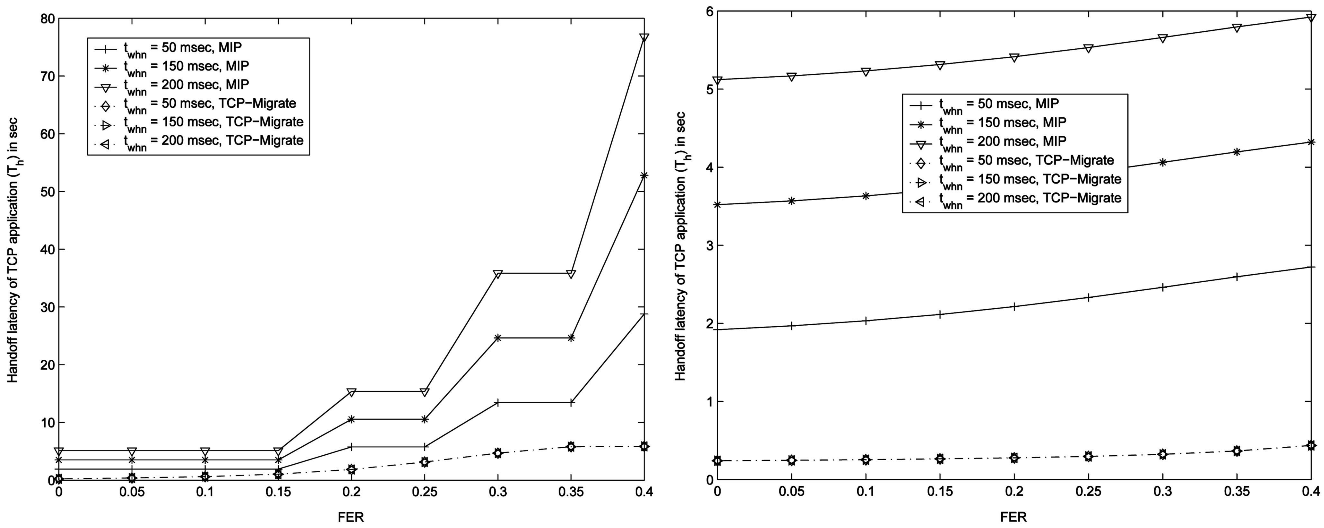
$$\begin{aligned} T_{t2} &= D - A = (D - C_1) + (C - A) - (C - C_1) \\ &= \tau_{L2} + \tau_a + E[L] + [log_2(1 + CW_2) - 1]RTT_n, \end{aligned} \tag{29}$$

where τ_{L2} and τ_a are defined in (18). t_n is defined in Section 3. RTT_n is the RTT when the MH is in the NN. $E[L]$ is given by (28).

4.3 Handoff Performance Comparison of Mobile IP and TCP-Migrate for a TCP Connection

To compare the performance of Mobile IP (MIP) and TCP-Migrate-based handoff for a TCP connection, we assume the following values for different parameters: $\tau_{L2} = 10$ ms and $\tau_a = 20$ ms. τ_{L2} and τ_a are defined in (18). The link layer access delay $D = 10, 50, 150$ ms for WLAN, 3G cellular, and satellite networks, respectively [1]. The length of link layer frame $L_f = 19$ bytes, link layer interframe interval $\tau = 20$ ms, and packet loss probability in the wired network $p_c = 1e-5$. $t_{ch} = 50$ ms, $t_{wco} = 100$ ms, and $t_{wcn} = 100$ ms. We consider $t_{who} = t_{whn}$ and use different values for them in our simulations. t_{ch}, t_{wco}, t_{wcn}, t_{who}, and t_{whn} are defined in Section 3.

Fig. 4a shows the handoff latency comparison of Mobile IP and TCP-Migrate for a TCP connection when no RLP is used in the link layer. Similarly, Fig. 4b shows the handoff

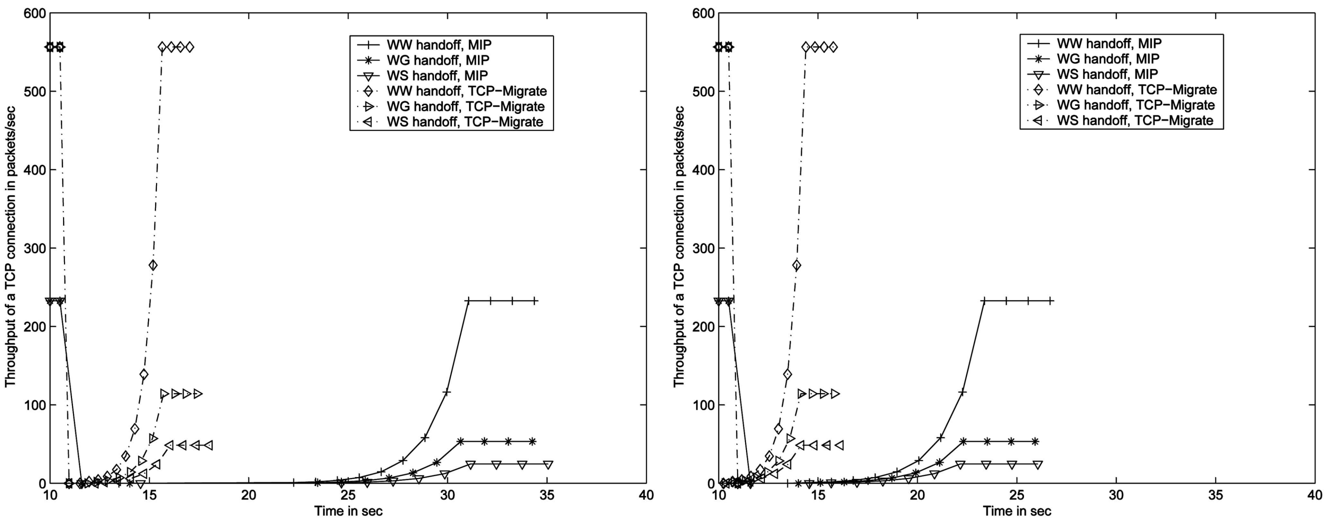


(a) (b)

Fig. 4. Handoff latency comparison of Mobile IP and TCP-Migrate: (a) no RLP and (b) RLP.

latency comparison for Mobile IP and TCP-Migrate when RLP is used. The results show that, for both no RLP and RLP scenarios, the handoff latency of Mobile IP is always greater than that of TCP-Migrate. The reason is twofold. First, the Mobile IP signaling messages are transferred between the MH and its HA, whereas TCP-Migrate signaling messages are transferred between the MH and the CH. Typically, the distance between the MH and its HA is higher than the distance between the MH and the CH. Second, Mobile IP handoff is not transparent to TCP. Therefore, even after MH's new CoA is registered at the HA, the TCP waits until the retransmission timer to timeout before sending a new packet. On the other hand, when TCP-Migrate is used, CH's TCP resumes the TCP connection as soon as it receives the new IP address. The results also show that the handoff latency for Mobile IP and TCP-Migrate increases as the wireless link FER increases. This can be explained as follows: When no RLP is used in the link layer, a higher value of FER increases the probability of erroneous packet transfer across the link layer. Therefore, the handoff signaling messages have to be retransmitted several times before the successful completion of a handoff. Similarly, when RLP is used in the link layer, a higher value of FER requires a higher number of link layer retransmissions for the successful transfer of handoff messages across the link layer. This increases the link layer packet transfer delay and results in higher handoff signaling delay. During a handoff, the MH is around the boundary of a cell coverage and suffers from higher link layer FER. When no RLP is used, higher FER results in very high Mobile IP handoff latency. For an FER of around 0.2, the Mobile IP handoff latency is around five times higher than the handoff latency of TCP-Migrate. Moreover, as shown in Fig. 4a and Fig. 4b, Mobile IP handoff depends on the delay between the MH and its HA (t_{whn}) as the signaling messages are exchanged between them. On the other hand, as expected, the handoff latency of TCP-Migrate depends only on the distance between the MH and the CH.

The throughput of a TCP connection during a handoff is shown in Fig. 5a and Fig. 5b for no RLP and RLP scenarios, respectively. To investigate the throughput performance of Mobile IP and TCP-Migrate, we use $p_f = 0.2$ and $t_{who} = t_{whn} = 200$ ms. Fig. 5a and Fig. 5b show the throughput of a TCP connection when the MH previously in a WLAN moves to a WLAN or 3G cellular, or satellite network. We refer to the handoff from a WLAN to another WLAN network as WW handoff. Similarly, WLAN to 3G cellular and WLAN to satellite network handoffs are referred as WG handoff and WS handoff, respectively. In Fig. 5a and Fig. 5b, the MH moves into the NN at time 10.5 seconds. Therefore, before this time, the TCP connection operates in a steady state corresponding to the ON, which is a WLAN in this case. Then, after MH's movement to the NN (a WLAN, 3G network, or satellite network) until the handoff process is completed the packets destined for the MH are lost resulting in zero throughput. After the successful registration of MH's new CoA at the HA, the MH starts to receive packets in the NN. As TCP starts from slow start after the handoff, it takes a finite amount of time for TCP to reach its steady state in the NN. Fig. 5a and Fig. 5b show that this time is minimum in the case of WLAN to WLAN (WW) handoff and maximum for WLAN to satellite network (WS) handoff. This because the one-way access delay of a WLAN network is the lowest and that of the satellite network is the highest. The dotted lines and solid lines represent the throughput of the TCP connection for TCP-Migrate and Mobile IP, respectively. The results also show that the throughput degradation of the TCP connection lasts longer for Mobile IP than that of TCP-Migrate. The higher handoff latency of Mobile-IP-based handoff results in a longer throughput degradation time compared to TCP-Migrate based handoff. Fig. 5a and Fig. 5b show that, for the parameters considered in our



(a) (b)

Fig. 5. Throughout degradation duration comparison of a TCP connection for Mobile IP and TCP-Migrate: (a) no RLP and (b) RLP.

analysis, the throughput degradation during a Mobile IP handoff is around twice that of TCP-Migrate. The numerical value of the handoff degradation depends on the handoff latency that depends on the numerical value of FER and the distance between the MH and CH and the MH and its HA. However, Mobile IP always has higher handoff latency and higher throughput degradation time compared to TCP-Migrate.

To summarize, the handoff latency and throughput degradation time of Mobile IP depend on the link layer FER (p_f), the delay between the MH and HA, and wireless access technology. Similarly, the handoff latency and throughput degradation time of TCP-Migrate-based handoff depend on the link layer FER (p_f), the delay between the MH and CH, and wireless access technology. TCP-Migrate has lower handoff latency and lower handoff degradation time for *Class B* and *Class C* applications compared to Mobile IP. Therefore, we advocate that TCP-Migrate is suitable for these applications.

5 Handoff Performance of *Class D* and *Class E* Applications (Mobile IP and SIP)

As *Class D* and *Class E* applications use UDP, we consider a UDP connection between the MH and the CH to investigate their handoff performance. The handoff performance of *Class D* and *Class E* applications is synonymous with the handoff performance of a UDP connection. We consider a voice over IP (VoIP) application that uses RTP over UDP. It may be noted that the same analysis is valid for other real and non-real-time applications using UDP. Out of the different handoff performance parameters discussed in Section 1, since we are considering a UDP connection, we do not consider the *transport-layer transparency*. Both Mobile IP and SIP support secure handoff. Therefore, we also do not consider *security* in our analysis. As a result, we involve the following three metrics to investigate the performance of Mobile IP and SIP for the VoIP application: *handoff latency*, *packet loss during handoff*, and *end-to-end delay*. The *end-to-end delay* corresponds to the transportation delay of the VoIP data packets.

5.1 Handoff Performance of a UDP Connection When Mobile IP Is Used

Fig. 6a shows the Mobile IP [17] handoff process of a UDP connection when the MH moves from the ON to the NN. In Fig. 6a, t_{ch}, t_{hn}, t_n, and t_o are defined in Section 3. As shown in Fig. 6a, the MH starts layer 2 handoff to the NN and IP address acquisition from the NN at time A. These procedures are completed at time B. Then, at time B, the MH starts Mobile IP [17] registration with its HA. The new CoA of the MH gets successfully registered at the HA at time instant C. Thus, packets received by the HA after time C are correctly forwarded to the MH in the NN. The packets received by MH's HA from the CH between time G and C are lost as they are forwarded to MH's old CoA. We refer to handoff latency as the time elapsed after the MH receives the last packet in the ON until the MH receives the first packet in the NN. Next, we derive the mathematical formulations for *handoff latency*, *packet loss during handoff*, and *end-to-end delay*.

5.1.1 Handoff Latency

From Fig. 6a, the handoff latency of the UDP connection is given by

$$T_{h3} = D - A = \tau_{L2} + \tau_a + \tau_m + t_{ch} + t_{hn}, \tag{30}$$

where τ_{L2}, τ_a, and τ_m are defined in (18). t_{ch} and t_{hn} are defined in Section 3.

5.1.2 Packet Loss

From Fig. 6a, the packets that are intercepted by the HA between time G and C are lost. Therefore, if the packet transmission rate of the CH is R, the number of packets that are lost during handoff is given by

$$P_h = R(C - G) = R(\tau_{L2} + \tau_a + \tau_m + t_{ho}). \tag{31}$$

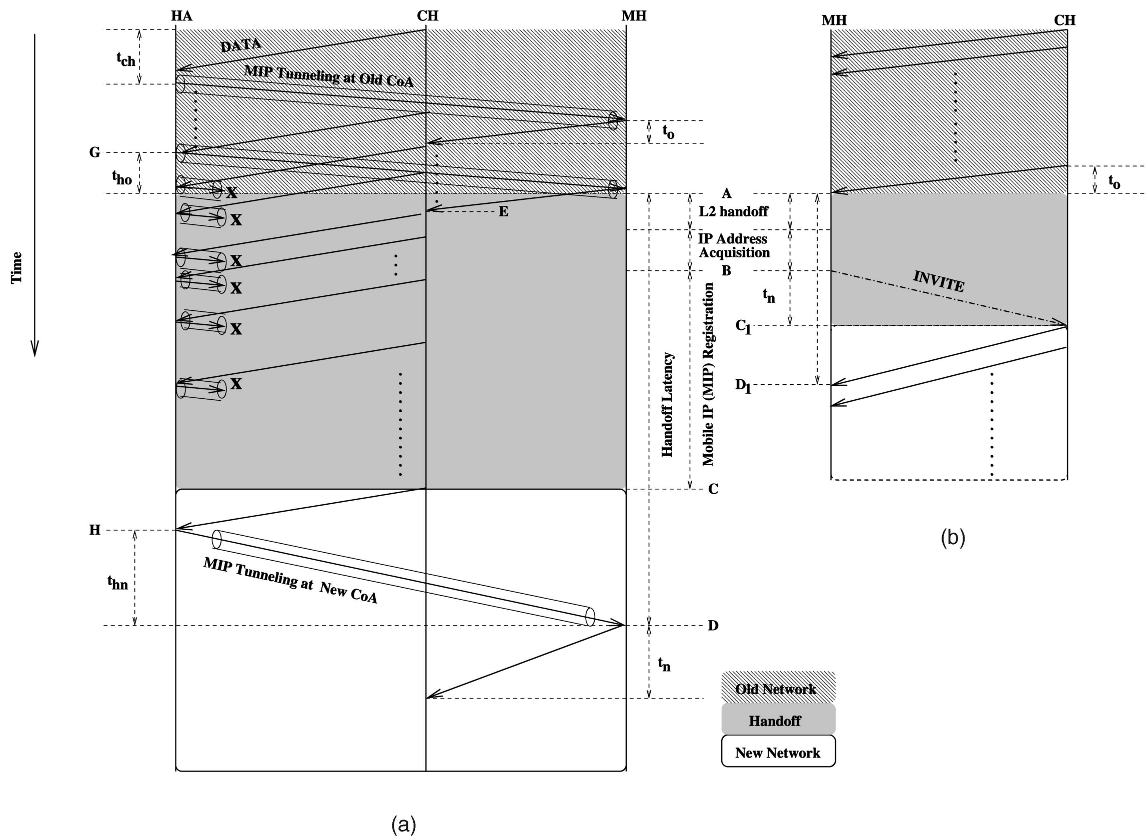


Fig. 6. Handoff of a UDP connection using (a) Mobile IP and (b) SIP.

5.1.3 End-to-End Packet Transportation Delay

The end-to-end packet transportation delay of the VoIP data packets in the path from the MH to the CH D_{fm} without RLP D_{fmnr} and with RLP D_{fmr} are, respectively, given by

$$D_{fmnr} = D + t_{wcn} \quad (32)$$

and

$$D_{fmr} = D + (K_p - 1)\tau + t_{wcn.} \quad (33)$$

D and t_{wcn} are defined in (5) and Section 3, respectively. $K_p = \lceil \frac{L_p}{L_f} \rceil$ is the number of link layer frames per one VoIP data packet, where L_p is the length of one VoIP data packet and L_f is the length of a link-layer frame. Similarly, the end-to-end packet transportation delay from CH to MH path (D_{rm}) without RLP D_{rmnr} and with RLP D_{rmr} are, respectively, given by

$$D_{rmnr} = D + t_{ch} + t_{whn} \quad (34)$$

and

$$D_{rmr} = D + (K_p - 1)\tau + t_{ch} + t_{whn} \quad (35)$$

for no RLP and RLP scenarios, respectively. t_{ch} and t_{whn} are defined in Section 3.

5.2 Handoff Performance of a UDP Connection When SIP Is Used

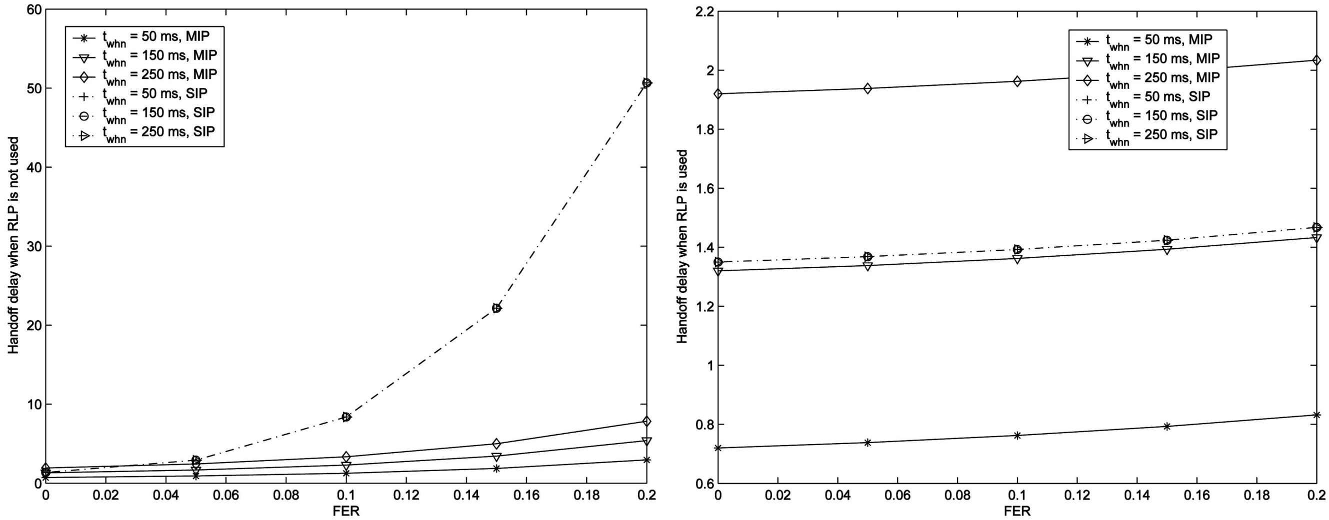
In the case of SIP-based mobility management, when the MH moves from the ON to the NN, it sends a new *INVITE* [20] message to the CH using the same call identifier as in the original call setup as shown in Fig. 6b. The MH puts its new IP address in the *contact* field of SIP *INVITE* message [25]. This new IP address informs the CH about MH's change of network. Therefore, after receiving MH's new IP address, CH sends the VoIP data packets to MH's new address. Fig. 6b shows the SIP [20] handoff process when the MH moves from the ON to the NN. At time A, the MH starts the handoff process to the NN. As shown in Fig. 6b, the MH starts layer 2 handoff to the NN and IP address acquisition from the NN at time A. These procedures are completed at time B. Then, at time B, the MH sends the *INVITE* message to the CH that is received by the CH at time C_1. Thus, packets sent by the CH between time $A - t_o$ and C_1 are lost as they were sent to the old IP address of the MH. Next, we derive the mathematical formulations for *handoff latency*, *packet loss during handoff*, and *end-to-end delay*.

5.2.1 Handoff Latency

From Fig. 6b, the handoff latency when SIP is used is given by

$$T_{h4} = D_1 - A = \tau_{L2} + \tau_a + 2D_{mc},$$

where τ_{L2} and τ_a are defined in (18). D_{mc} is the average one-way delay to transport SIP signaling packets between the MH and the CH. SIP signaling messages can be transferred using either UDP or TCP [20]. For our analysis, we consider that SIP signaling messages are transferred over UDP. Using steps similar to the derivation of (22), D_{mc} is given by



(a) (b)

Fig. 7. Handoff latency comparison of Mobile IP and SIP: (a) no RLP and (b) RLP.

$$D_{mc} = (1 - q_2)\left\{ B_2 + A_2 \sum_{i=2}^{m} q_2^{i-1}(\gamma^{i-1} - 1) + \sum_{i=m+1}^{\infty} q_2^{i-1}[A_2(\gamma^{m-1} - 1) + (i - m)\gamma^{m-2}\Delta_2] \right\}, \tag{36}$$

where B_2 is the end-to-end packet transportation delay between the MH and the CH. $B_2 = B_{2nr}$ when no RLP is used and $B_2 = B_{2r}$ when RLP is used. B_{2nr} is computed from (5) by using $T_{nr} = B_{2nr}$ and $t_w = t_{wcn}$. t_{wcn} is the one-way delay in the wired network between the new BS (NBS) and the CH. B_{2r} is computed from (8) by using $T_r = B_{2r}$, $K = K_s$, and $t_w = t_{wcn}$. $K_s = \lceil \frac{L_s}{L_f} \rceil$ is the number of wireless link layer frames per one SIP *INVITE* message, where L_s is the length of a SIP *INVITE* message and L_f is the length of a link-layer frame. q_2 is the end-to-end packet loss probability between the MH and the CH. $q_2 = q_{2nr}$ when no RLP is used and $q_2 = q_{2r}$ when RLP is used. q_{2nr} is computed from (2) by using $p_{nr} = q_{2nr}$ and $K = K_s$. q_{2r} is computed from (4) by using $p_r = q_{2r}$ and $K = K_s$. Δ_2 is the initial value of the retransmission timer for SIP signaling messages. γ and m are as defined in (11).

5.2.2 Packet Loss

From Fig. 6b, the packets that are transmitted by the CH between time $A - t_o$ and C_1 are lost. Therefore, if the packet transmission rate of the CH is R, the number of packets that are lost during handoff is given by

$$P_{h1} = R(C_1 - A + t_o) = R(\tau_{L2} + \tau_a + D_{mc} + t_o).$$

5.2.3 End-to-End Packet Transportation Delay

The end-to-end packet transportation delay of the VoIP data packets in the path from MH to the CH and reverse path are same for both no RLP and RLP scenarios and are given by

$$D_{fsnr} = D_{rsnr} = D + t_{wcn} \tag{37}$$

and

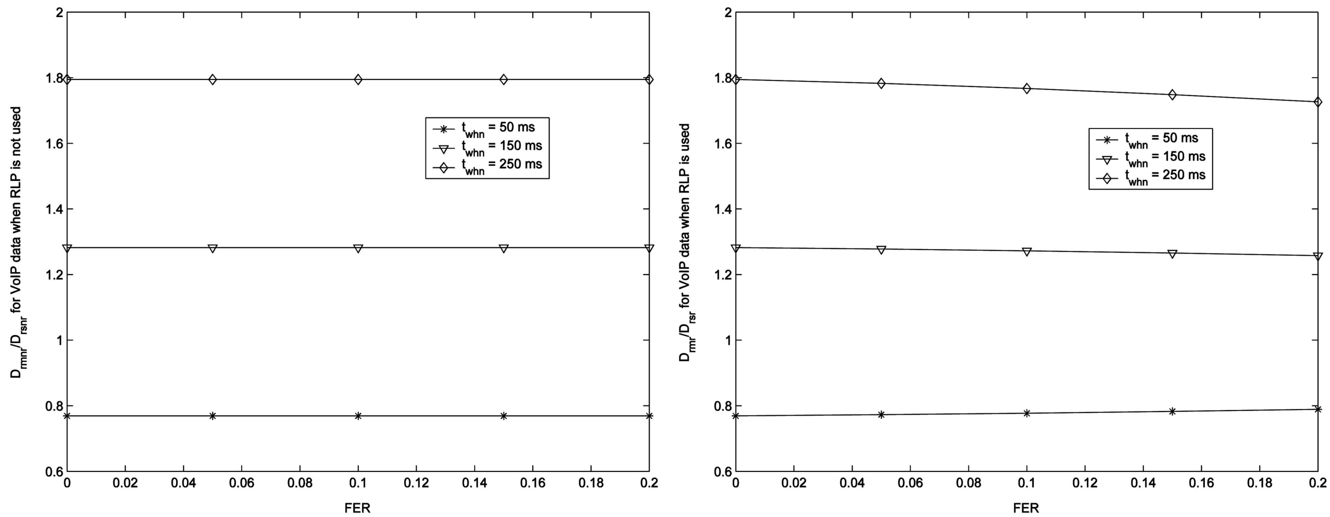
$$D_{fsr} = D_{rsr} = D + (K_p - 1)\tau + t_{wcn}, \tag{38}$$

where D_{fsnr} and D_{fsr} are the end-to-end packet transportation delay of the VoIP data packets in the path from MH to the CH for no RLP and RLP scenarios, respectively. Similarly, D_{rsnr} and D_{rsr} are the end-to-end packet transportation delay of the VoIP data packets in the reverse path for no RLP and RLP scenarios, respectively. Therefore, when SIP is used, the packet transportation delay is symmetric in both directions. This is because there is no packet redirection when SIP is used.

5.3 Handoff Performance Comparison of Mobile IP and SIP for a UDP Connection

To compare the handoff performance of Mobile IP (MIP) and SIP-based mobility management, we assume the following parameters: the lengths of the SIP *INVITE* message (L_s) and the *Mobile IP Registration Request/Reply* message (L_m) are 140 bytes and 56 bytes [5], respectively. We consider the numerical values specified in Section 4.3 for other parameters. We consider that the length of one VoIP data packet is 87 bytes [21], which includes 20 bytes of IP header, 14 bytes of IP options, 8 bytes of UDP header, and 45 bytes of RTP message (33 bytes of voice data and 12 bytes of RTP header). The 33 bytes of voice data are generated by a GSM codec in every 20 ms. When Mobile IP is used, packets are tunneled from the HA to the MH. This adds another 20 bytes of IP header making the total IP packet of length 107 bytes.

Fig. 7a shows the handoff latency comparison of Mobile IP and SIP for different values of FER (p_f) when no RLP is used in the link layer. It shows that, for smaller values of p_f, the handoff latency of SIP is lower than that of the Mobile IP. On the other hand, for a larger value of p_f, the handoff latency of SIP is higher than that of Mobile IP. This can be explained as follows: There are two factors that decide the numerical value of handoff delay. One of them is the delay



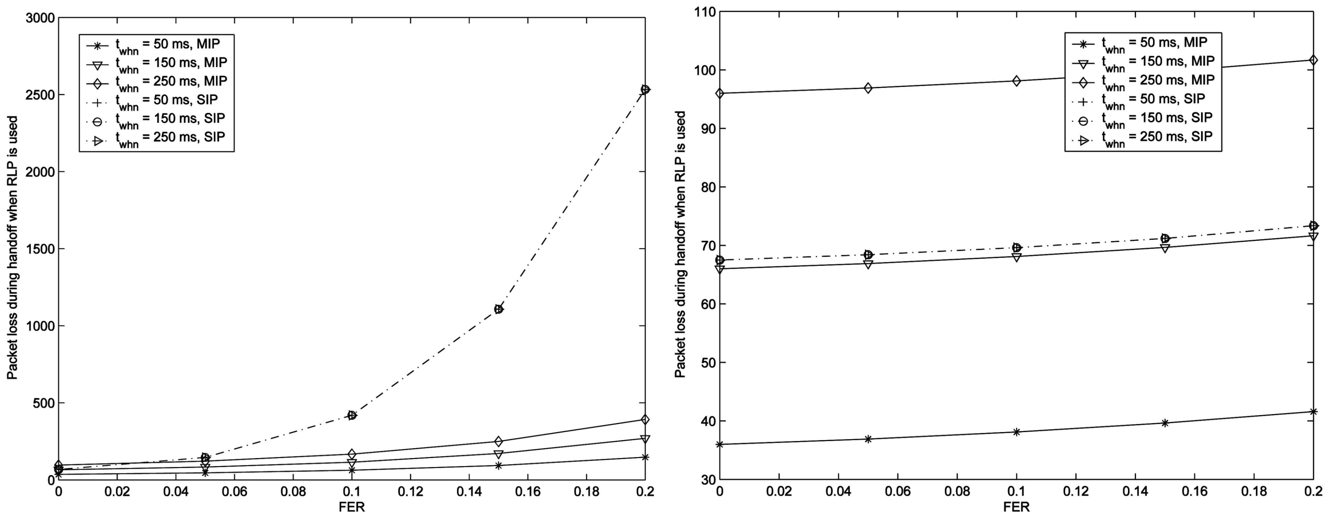
(a) (b)

Fig. 8. End-to-end packet transportation delay comparison of Mobile IP and SIP: (a) no RLP and (b) RLP.

to transfer a handoff signaling message across the link layer and the other is the delay to transfer the handoff signaling message in the wired network. The delay across the wireless link depends on the number of link layer frames and the numerical value of link layer FER. The larger the size of a packet the higher is, the probability that it gets erroneous during its transfer over the link layer. Therefore, the number of retransmissions required for the successful transfer of the signaling message increases. This increases the average signaling delay. As the size of a SIP handoff signaling message is larger than that of the Mobile IP handoff signaling message (the lengths of the SIP *INVITE* message (L_s) and the *Mobile IP Registration Request/Reply* message (L_m) are 140 bytes and 56 bytes [5], respectively), the SIP messages require a higher number of retransmissions. This results in higher handoff latency for SIP compared to Mobile IP. Moreover, the difference in the handoff latency between SIP and Mobile IP becomes larger as the link layer FER increases. The other part of the handoff latency that is incurred because of the handoff signaling transportation delay over the wired network depends on the distance between the entities involved in the handoff process. In the case of SIP, the handoff signaling messages are exchanged between the MH and the CH, whereas, in the case of Mobile IP, the handoff signaling messages are exchanged between the MH and the HA. In most cases, the distance between the MH and the HA is larger than the distance between the MH and the CH. Therefore, the wired part of the handoff latency is larger for Mobile IP than SIP. When the wireless link FER is low or its effect is reduced through the use of link layer RLP, the delay in the wired network influences the overall handoff latency. Therefore, for such scenarios, Mobile IP has higher handoff latency. On the other hand, for a higher value of wireless link FER, the delay over the wireless link plays a major role in making the handoff latency of SIP larger than that of Mobile IP. The results shown in Fig. 7a and Fig. 7b verify this. Fig. 7a shows that the handoff latency for SIP is lower than that of Mobile IP for lower values of FER and higher for higher values of FER. On the other hand, when the effect of link layer FER is reduced through the use of link layer RLP, SIP sometimes has lower handoff latency compared to Mobile IP as shown in Fig. 7b.

The number of packets lost during a handoff is proportional to the handoff latency. Therefore, the number of lost packets for SIP and Mobile IP have a similar nature to the handoff delay. These results are shown in Fig. 9a and Fig. 9b. When no RLP is used, SIP suffers from higher packet loss during handoff for higher values of FER. This is because, as an SIP *INVITE* message is larger than the *Mobile Registration Request/Reply* message, the probability that an SIP *INVITE* message is lost over the link-layer is higher. This increases the average handoff delay resulting in higher packet loss. However, when RLP is used as this link layer loss is compensated by link layer retransmissions, the longer length of the SIP *INVITE* message does not come into the picture. Since most of the current wireless systems implement RLP and the distance between the MH and the HA is usually higher, Mobile IP is expected to suffer from higher packet losses.

Fig. 8a and Fig. 8b show the end-to-end packet transportation delay comparison of Mobile IP and SIP for different values of FER (p_f). The results show that Mobile IP has a higher end-to-end packet transportation delay for all values of FER. This is because the packets follow triangular routes instead of straight routes between the MH and the CH when Mobile IP is used. On the other hand, packets follow the direct path between the MH and the CH when SIP is used. This is one of the major disadvantages of using Mobile-IP-based mobility management for real-time applications. The results show that the end-to-end packet transportation delay of Mobile IP can be 80 percent higher than that of SIP (the actual value depends on the particular network conditions). This would go higher depending on the distance between the MH and the HA. Since real-time applications such as VoIP require minimum end-to-end



(a) (b)

Fig. 9. Packet loss during handoff comparison of Mobile IP and SIP: (a) no RLP and (b) RLP.

delay, SIP-based mobility management is preferred over Mobile IP. When RLP is not implemented in the link layer, the packet transportation delay across the link layer remains independent of the numerical value of link layer FER. Therefore, the end-to-end packet transportation delay ratio for Mobile IP and SIP remains the same for all values of link layer FER. This is verified by the results shown in Fig. 8a. On the other hand, when RLP is implemented at the link layer, the number of retransmissions that are required for the successful transfer of a VoIP data packet across the link layer increases as the link layer FER increases. Therefore, the effect of a triangular routing of Mobile IP on the end-to-end packet transportation delay reduces. This reduces the ratio of end-to-end packet transportation delay as the FER increases as shown in Fig. 8b.

6 Summary and Conclusions

To summarize, our analysis shows that the handoff performance of a mobility management protocol depends on the following factors:

- **Type of application:** Different applications use different transport layer protocols. As the operating principles of different transport layer protocols are different, they react differently to the handoff. Therefore, the performance of a particular mobility management protocol is different for different types of applications. For example, as discussed earlier, the handoff latency of Mobile-IP-based handoff is larger for applications using TCP than applications using UDP. This is because, when packets are lost during the handoff, TCP went through retransmission timeouts before retransmitting the lost packets.
- **Link layer frame error probability:** Our analysis shows that the handoff latency, end-to-end packet transportation delay, and packet loss during handoff depend on the link layer frame error probability (p_f), both when no RLP is used and when RLP is used.
- **Signaling delay:** Handoff latency and packet loss during handoff depend on the signaling delay between the network entities that are involved in a handoff, e.g., MH and HA in the case of Mobile IP and MH and CH in the case of SIP and TCP-Migrate.
- **Link layer access technologies:** As observed in our analysis, different types of link layer access technologies such as the use of RLP also influence the numerical value of handoff parameters. Moreover, the link layer access delay that is different for different access technologies also influences the handoff performance.

Based on our handoff performance investigation, we advocate the use of TCP-Migrate for applications using TCP, i.e., *Class B* and *Class C* applications. SIP is suitable for real-time applications using UDP. However, SIP is standardized only for real-time applications; therefore, Mobile IP can be used for non-real-time applications that use UDP. In summary, different mobility management protocols operating from different layers of the classical protocol stack are suitable for different classes of applications. The use of application-adaptive mobility itself is not enough to support seamless mobility management. This is revealed in our analysis where we observe that the handoff performance depends heavily on link layer FER, the delay between different network entities that are involved in the handoff, and the wireless access technology. Therefore, we advocate information sharing between different layers to enhance the performance of mobility management. This cross-layering approach will eliminate the negative effects of different parameters such as link layer frame error rate and signaling delay on the handoff performance of mobility management protocols.

References

[1] O.B. Akan and I.F. Akyildiz, "ATL: An Adaptive Transport Layer for Next Generation Wireless Internet," *IEEE J. Selected Areas in Comm. (JSAC)*, vol. 22, no. 5, pp. 802-817, June 2004.

[2] I.F. Akyildiz and W. Wang, "A Predictive User Mobility Profile for Wireless Multimedia Networks," *IEEE/ACM Trans. Networking,* vol. 12, no. 6, pp. 1021-1035, Dec. 2004.
[3] I.F. Akyildiz et al., "Mobility Management for Next Generation Wireless Systems," *Proc. IEEE,* vol. 87, no. 8, pp. 1347-1384, Aug. 1999.
[4] I.F. Akyildiz, J. Xie, and S. Mohanty, "A Survey on Mobility Management in Next Generation All-IP Based Wireless Systems," *IEEE Wireless Comm.,* vol. 11, no. 4, pp. 16-28, Aug. 2004.
[5] N. Banerjee, W. Wu, S. Das, S. Dawkins, and J. Pathak, "Mobility Support in Wireless Internet," *IEEE Wireless Comm.,* vol. 10, no. 5, Oct. 2003.
[6] G. Bao, "Performance Evaluation of TCP/RLP Protocol Stack over CDMA Wireless Links," *ACM/Springer J. Wireless Networks (WINET),* no. 3, pp. 229-237, 1996.
[7] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP Latency," *Proc. INFOCOM,* Mar. 2000.
[8] P. Conrad et al. "SCTP in Battlefield Networks," *Proc. IEEE Military Comm. Conf. (MILCOM),* pp. 289-295, Oct. 2001.
[9] E. Gustafsson, A. Jonsson, and C.E. Perkins, "Mobile IPv4 Regional Registration," IETF Internet draft, work in progress, Nov. 2004.
[10] E. Gustafsson and A. Jonsson, "Always Best Connected," *IEEE Wireless Comm.,* pp. 49-55, Feb. 2003.
[11] T.R. Henderson, "Host Mobility for IP Networks: A Comparison," *IEEE Network,* vol. 17, no. 6, pp. 18-26, Nov./Dec. 2003.
[12] H.-Y. Hsieh, K.-H. Kim, and R. Sivakumar, "An End-to-End Approach for Transparent Mobility Across Heterogeneous Wireless Networks," *ACM/Kluwer Mobile Networks and Applications J. (MONET),* special issue on integration of heterogeneous wireless technologies, vol. 9, no. 4, pp. 363-378, Aug. 2004.
[13] P. Karn, "Qualcomm White Paper on Mobility and IP Addressing," http://people.qualcomm.com/karn/papers/mobility.html, Feb. 1997.
[14] D. Maltz and P. Bhagwat, "MSOCKS: An Architecture for Transport Layer Mobility," *Proc. INFOCOM,* pp. 1037-1045, Mar. 1998.
[15] J. McNair, I.F. Akyildiz, and M. Bender, "An Inter-System Handoff Technique for the IMT-2000 System," *Proc. INFOCOM,* Mar. 2000.
[16] A. Misra, S. Das, A. Dutta, A. McAuley, and S. Das, "IDMP-Based Fast Handoffs and Paging in IP-Based 4G Mobile Networks," *IEEE Comm. Magazine,* pp. 138-145, Mar. 2002.
[17] C. Perkins et al., *IP Mobility Support for IPv4,* IETF RFC 3344, Aug. 2002.
[18] C. Perkins, P.R. Calhoun, and J. Bharatia, "Mobile IPv4 Challenge/Response Extensions (Revised)," IETF Internet draft, work in progress, May 2003.
[19] R. Ramjee, K. Varadhan, L. Salgarelli, S.R. Thuel, S.-Y. Wand, and T. La Porta, "HAWAII: A Domain-Based Approach for Supporting Mobility in Wide-Area Wireless Networks," *IEEE/ACM Trans. Networking,* vol. 10, no. 3, pp. 396-410, June 2002.
[20] J. Rosenberg et al., *SIP: Session Initiation Protocol,* IETF RFC 3261, June 2002.
[21] S. Seol, M. Kim, C. Yu, and J. Lee, "Experiments and Analysis of Voice over Mobile IP," *Proc. 13th IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm. (PIMRC '02),* Sept. 2002.
[22] A.C. Snoeren and H. Balakrishnan, "An End-to-End Approach to Host Mobility," *Proc. MobiCom,* Aug. 2000.
[23] R. Stewart et al., *Stream Control Transmission Protocol,* IETF RFC 2960, Oct. 2000.
[24] A. Valko, "Cellular IP: A New Approach to Internet Host Mobility," *ACM SIGMOBILE Computer Comm. Rev.,* vol. 29, no. 1, pp. 50-65, Jan. 1999.
[25] E. Wedlund and H. Schulzrinne, "Mobility Support Using SIP," *Proc. Second ACM/IEEE Int'l Conf. Wireless and Mobile Multimedia (WoWMoM'99),* Aug. 1999.
[26] Y.R. Yang and S.S. Lam, "General AIMD Congestion Control," *Proc. IEEE 2000 Int'l Conf. Network Protocols (ICNP '00),* Nov. 2000.
[27] X. Zhao, C. Castelluccia, and M. Baker, "Flexible Network Support for Mobility," *Proc. MobiCom,* Oct. 1998.

Shantidev Mohanty received the BTech (Hons) degree from the Indian Institute of Technology, Kharagpur, India, in 2000. He received the MS and PhD degrees from the Georgia Institute of Technology, Atlanta, Georgia, in 2003 and 2005, respectively, both in electrical engineering. He is currently working with Intel Corporation, Santa Clara, California. His current research interests include wireless networks, mobile communications, mobility management, WiMAX networks, and cross-layer protocol design. From 2000 to 2001, he worked as a mixed signal design engineer for Texas Instruments, Bangalore, India. He worked as a summer intern for Bell Labs, Lucent Technologies, Holmdel, New Jersey, during the summers of 2002 and 2003 and for Applied Research, Telcordia Technologies, Piscataway, New Jersey, during the summer of 2004. He is a member of the IEEE.

Ian F. Akyildiz received the BS, MS, and PhD degrees in computer engineering from the University of Erlangen-Nuernberg, Germany, in 1978, 1981, and 1984, respectively. Currently, he is the Ken Byers Distinguished Chair Professor in the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, and the director of the Broadband and Wireless Networking Laboratory. He is an editor-in-chief of the *Computer Networks Journal* (Elsevier) as well as the founding editor-in-chief of the *Ad Hoc Network Journal* (Elsevier). His current research interests are in next-generation wireless networks, sensor networks, IPN Internet, wireless networks, and satellite networks. He received the "Don Federico Santa Maria Medal" for his services to the Universidad de Federico Santa Maria in 1986. From 1989 to 1998, he served as a national lecturer for the ACM and received the ACM Outstanding Distinguished Lecturer Award in 1994. He received the 1997 IEEE Leonard G. Abraham Prize Award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures" published in the *IEEE Journal of Selected Areas in Communications* (JSAC) in January 1996. He received the 2002 IEEE Harry M. Goode Memorial Award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking." He received the 2003 IEEE Best Tutorial Award (IEEE Communication Society) for his paper entitled "A Survey on Sensor Networks," published in *IEEE Communications Magazine* in August 2002. He also received the 2003 ACM Sigmobile Outstanding Contribution Award with the citation "for pioneering contributions in the area of mobility and resource management for wireless communication networks." He received the 2004 Georgia Tech Faculty Research Author Award for his "outstanding record of publications of papers between 1999-2003." He also received the 2005 Distinguished Faculty Achievement Award from School of ECE, Georgia Tech. He has been a fellow of the ACM since 1996. He is a fellow of the IEEE and a member of the IEEE Computer Society.