Chapter 7

Optimizations for simultaneous mobility

In this chapter, I analyze the problem due to non-receipt of binding update that results when both the mobile nodes move simultaneously and propose optimization techniques that increase the successful handover probability under the simultaneous mobility scenario. These optimization techniques could be applied to mobility protocols at several layers - network layer mobility protocols such as MIPv6 [JPA04] and MIP-LR [JRY+99] and application layer mobility protocol such as SIP-based mobility [SW00].

7.1 Introduction

Stoica et al. [ZLS+05] propose seven properties that are needed to fully realize the promise of ubiquitous mobility. These properties also include simultaneous mobility. It is expected that non-simultaneous mobility in most scenarios would occur more frequently than simultaneous mobility. Non-simultaneous mobility refers to mobility of one end host while the other remains stationary. Nevertheless, simultaneous mobility would happen once in a while and must be handled properly by the mobility protocols.

Simultaneous mobility problem occurs when two mobile nodes that are part of a communication session in normal state, and they both move such that the binding updates that they send to each other are both lost through belated arrival of binding update, and such
that the communication session never returns from interrupted state to normal state. More precisely, simultaneous mobility problem can be defined as the problem of losing a binding update from one mobile node because it is sent to a previous address of the other mobile node that is also moving at around the same time. The disruption caused by the simultaneous mobility problem may far exceed the disruption caused by non-simultaneous mobility.

Thus, the optimization techniques related to simultaneous mobility is a special type of optimization for binding update as discussed in Chapter 5. Any solution for simultaneous mobility should ensure that the end hosts should be able to move simultaneously without breaking an ongoing session between them due to delayed binding update.

7.1.1 Analysis of simultaneous mobility

In this section, I analyze the simultaneous mobility event and describe several concepts associated with simultaneous mobility.

I primarily limit the analysis of simultaneous mobility to layer 3 handoff only, i.e., where IP addresses of the mobile nodes change. A binding update carries the information about the location of the sending mobile host including its new IP address. A binding update is lost if it does not arrive at its intended recipient mobile host. It makes a belated arrival if it arrives at a network where the destination address used to be valid for the intended recipient Mobile Host but it is no longer valid at the moment of arrival. Binding updates do not contain information about future moves of the sending mobile host. While two mobile hosts are in a communication session, they get information on the location of the other Mobile Host only from binding updates. In other words, they do not actively seek the location of the other mobile host, but only passively accept binding updates. Binding updates are sent directly to the most current known address (known by the sender) of the intended recipient mobile host. In general, the latency associated with the binding updates is assumed to be much smaller than the average inter-handoff time, therefore, it is extremely unlikely that a binding update would be sent and the recipient mobile host would move
twice before the binding update arrives at the previous network of the recipient.

The most basic version of simultaneous mobility problem is shown in Figure 7.1. There are two nodes, A and B. Time is in the vertical direction (and flows downward), whereas spatial location is in the horizontal direction. Node A moves from domain A1 to A2 while Node B moves from domain B1 to domain B2. After their respective moves, these two nodes send binding updates to the other node, and both binding updates are lost. Additionally, there are proxies and servers in the network as well.

![Simultaneous mobility scenario](image)

**Figure 7.1: Simultaneous mobility scenario**

Standard mobility protocols like the original Mobile IP (MIP) handle simultaneous mobility adequately, because of non-mobile home agents. The home agent of the Mobile Host functions as an anchor point for the Mobile Host. No matter where the Mobile Host moves, packets for it always go first to its home network for interception and are tunneled by its home agent. If it turns out that the Correspondent Host is also mobile, it will also have a home agent and packets from the Mobile Host will similarly be intercepted and tunneled to the appropriate network by its home agent. Since both home agents are stationary and can always be reached through IP routing, simultaneous mobility does not present a problem to MIPv4.

However, simultaneous mobility problem occurs for scenarios when the end hosts can send the binding updates directly to each other. Therefore, I analyze the simultaneous mobility problem for network layer mobility protocols such as MIPv6 and MIP-LR, and application layer mobility protocol such as SIP, and propose a common framework for the
solution. It is important to note that the problem of simultaneous mobility is very similar
in these protocols because these protocols allow binding updates to be sent to the commu-
icating hosts directly. My proposed solutions are designed to impose minimal changes
on the existing protocols while efficiently dealing with the simultaneous mobility prob-
lems. I focus on situations where the handoff rate of a mobile node is such that consecutive
handoffs of the same mobile node are non-overlapping. I do not focus on the situations
of overlapping consecutive handoffs of the same mobile node, where one handoff has not
completely finished before the next one begins, e.g., there has not been enough time after
the acquisition of an IP address for binding updates to reach their destination networks.

The following are the reasons for assumptions:

1. The problems encountered with overlapping consecutive handoffs are not so much
   a problem of simultaneous mobility as one of excessive handoff rate. There will
   be severe problems leading to complete deadlock situations when the mobile node
   changes its IP address before binding updates for its previous IP address have even
   arrived at their destinations.

2. For the foreseeable future, the extreme case of handoff rates high enough for overlap-
   ping consecutive handoffs is highly improbable. Hence, I assume that consecutive
   handoffs of the same mobile node are non-overlapping, and I focus on overlap of
   handoffs of different mobile nodes, i.e., simultaneous mobility.

7.2 Illustration of the simultaneous mobility problem

In this section, I illustrate how simultaneous mobility problems are encountered for SIP-
based mobility, MIPv6 and MIP-LR.

Figure 7.1 is easily adapted to illustrate the simultaneous mobility problem with SIP
and has been shown in Figure 7.2.
As shown in Figure 7.2, the binding updates are denoted as re-INVITE messages. The main difference is that there are two additional SIP servers with proxy and redirect functionality in each network, one for each mobile node in its home network. In general, when the re-INVITE messages are lost, the servers do not intervene, but I have proposed a solution that could use these servers and solve the simultaneous mobility problem in SIP-based mobility. These proposed solutions are explained in Section 7.6. Furthermore, Figure 7.2 could also be used to illustrate MIP-LR’s simultaneous mobility problem. In MIP-LR, the SIP servers are replaced with MIP-LR home location registers (HLR), and re-INVITE messages are replaced with MIP-LR binding updates.

Similarly, MIPv6 is vulnerable to the simultaneous mobility problem because of the direct binding updates and associated return routability procedures. The direct binding updates from the mobile node to the correspondent nodes pose a security problem. Thus, the return routability procedure allows the mobile node and correspondent node to set up a shared key in a “reasonably” secure manner. In the return routability procedure, a mobile node sends two messages to the correspondent node, namely the Home Test Init (HTI) and the Care-of Test Init messages (CTI). These messages are sent through the Home Agent (reverse tunneled to the Home Agent from the mobile node, and then forwarded to the cor-
CHAPTER 7. OPTIMIZATIONS FOR SIMULTANEOUS MOBILITY

The correspondent node replies by sending two tokens to the mobile node, one directly to the mobile node addressed to its care-of address (the Care-of Test message), and the other with the home address of the mobile node (the Home Test message). The mobile node needs both the tokens to be able to generate the shared key. Thus, the return routability procedure ensures that the mobile node is who it claims to be by testing that it is reachable on both the direct path and through its home address. Subsequently, the correspondent node can accept binding updates directly from the mobile node.

![Diagram of simultaneous mobility for MIPv6]

Figure 7.3: Simultaneous mobility for MIPv6

However, the additional message exchange due to return routability procedure adds to the existing simultaneous mobility problem. Figure 7.3 illustrates the simultaneous mobility problem with Mobile IPv6. Following are three different possible scenarios that could possibly result in simultaneous mobility in MIPv6.

1. Both sides’ CTI and HTI messages are lost because of simultaneous mobility. This would look like Figure 7.2 except that CTI and HTI messages are lost instead of re-INVITE (and home agents are used instead of SIP servers).

2. One side actually completes return routability, but then its binding update is lost.
because the other side moves. This interesting asymmetric scenario is illustrated in Figure 7.3.

3. Both sides complete the return routability checks, but then their binding updates are lost due to simultaneous mobility.

I propose solutions to take care of simultaneous mobility problems for SIP-based mobility, MIP-LR and MIPv6 in Section 7.6.

7.3 Related work

There are only a few papers that discuss about simultaneous mobility. Tilak and Gha-za-leh [TAG01] extend the TCP migration mobility protocol [SB00] to handle simultaneous mobility, but there are significant differences between the TCP migration schemes (where mobility is handled at the transport layer) and MIP-related protocols or SIP-based mobility protocols. Dreibholz et al. [DJT03] propose a scheme that handles simultaneous mobility at a layer between the transport and application layer. In that scheme, mobility is handled using stream control transmission protocol (SCTP) extensions. However, no analytical framework or theorems and proofs related to the simultaneous mobility problem for SIP, MIPv6 and MIP-LR have been proposed before.

As part of my research, I have analyzed the simultaneous mobility problems for MIP-LR, SIP and MIPv6 in [WDSY03] and [WD05]. I have also developed some common approaches that could be applied to provide solutions to mobility protocols such as MIPv6, MIP-LR and SIP-based mobility. I have described these results in [DWDSY07].

7.4 Key optimization techniques

Following are some of the key fundamental techniques that should be considered to optimize the handoff event during simultaneous mobility of the communicating hosts.
1. Many of the principles related to optimization for binding update are applicable to simultaneous mobility scenario. However, the handoff rate of the mobile node will determine if any of those techniques can be applied to simultaneous mobility scenario.

2. Reduce the effect of delayed direct binding updates by introducing an anchor point closer to the mobile.

3. Limit the traversal distance of binding updates.

4. Forward the binding updates from the previous network and cache it in a forwarding agent closer to the mobile.

5. Apply retransmission of binding update by the mobile nodes and proxies to complete update.

6. Apply simultaneous binding update by the mobile to reduce the failure probability of reconnection.

### 7.5 Analytical framework

In this section, I introduce an analytical framework to analyze simultaneous mobility problem. I define some of the fundamental concepts that are used to analyze the simultaneous mobility framework.

#### 7.5.1 Fundamental concepts

In this section, I introduce some fundamental concepts that are used to study the analytical framework associated with simultaneous mobility. In particular, I describe the terms such as handoff sequences and binding updates.
**Definition** Two mobile nodes are in a communication session if they are actively exchanging data. A communication session may be in a normal state or interrupted state. The session is in a normal state when data from one node is arriving at the right location for the other node, and vice versa. It is in an interrupted state otherwise.

**Example** A communication session typically is in an interrupted state from the moment a handoff occurs, until data starts arriving again at the new attachment point (e.g., after a binding update is received at the other node). An illustration of this alternation between normal state and interrupted state is shown in Figure 7.4. I explain this figure in more details.

![Figure 7.4: Simultaneous mobility framework notation](image)

### 7.5.2 Handoff sequences

As defined in Appendix C, handoff is a movement of a mobile node from a previous attachment point to a new attachment point. During simultaneous mobility, the handoff time of a handoff instance is the moment in time when it changes from being reachable at the previous attachment point to not reachable at the previous attachment point. Let the handoff time (of a particular handoff instance) be $T$. Then the node needs time for network configuration, so it becomes reachable (with a valid IP address at the new network) at time $T + \gamma$. If there is a correspondent node, then some time later, $T + \gamma + \zeta$, it sends a binding update to the correspondent node. The binding update arrives at time $T + \gamma + \zeta + \Delta$. I use these symbols to represent these differential times after a handoff. For convenience I may write $X(i) = \gamma(i) + \zeta(i) + \Delta(i)$ as shown in Figure 7.4. So $T(i) + X(i)$ denotes the time when the binding update arrives at the other node. Given that time is continuous, I assume that only
one handoff can occur at any given moment in time, i.e., handoff times are unique. It is to be noted that definition of handoff and handoff time may not be applicable to certain types of IP-layer soft handoff or physical layer soft handoff, as in CDMA systems and bicasting or multicasting schemes.

Figure 7.5 shows a scenario where A and B are two mobile nodes that are in a communication session with each other, during which each node performs zero, one or more handoffs.

**Definition** The handoff sequence of A is the ordered set

\[ H_A = T_A(0), T_A(1), ..., T_A(N_A - 1) \] (7.1)

and the handoff sequence of B is the ordered set

\[ H_B = T_B(0), T_B(1), ..., T_B(N_B - 1) \] (7.2)

where \( T_A(i) \) is the handoff time of the \( i \)th handoff of A so that \( T_A(i) < T_A(j) \), \( \forall i,j \) such that \( 0 < i < j < N_A - 1 \) and same holds good for B. The function arguments \( i,j \), are the handoff index number. In general, when necessary, we will use subscripts to indicate the mobile node and we will show the handoff index number in function arguments.

![Figure 7.5: Examples of consecutive handoffs](image)

Two handoffs are consecutive (with respect to a pair of mobile nodes) if neither of the mobile nodes performs another handoff in between the two handoffs. For example, if the two handoffs are at A and B, at times \( T_A(i_0) \) and \( T_B(j_0) \), and suppose that A's handoff is earlier, then saying they are consecutive is equivalent to saying \( T_A(i) \in H_A : T_A(i_0) < T_A(i) \)
< T_B(j_0) = \emptyset \) and \( T_B(j) \in H_B : T_A(i_0) < T_B(j) < T_B(j_0) = \emptyset \). As defined, then, consecutive handoffs could be at the same mobile node or at two different mobile nodes. Figure 7.5 shows two examples of consecutive handoffs, one in which the two handoffs are at different mobile nodes and one in which they are at the same mobile node.

### 7.5.3 Binding updates

**Definition** A binding update is lost if it does not arrive at its intended recipient node.

**Definition** A binding update makes a belated arrival if it arrives at a network where the destination address used to be valid for the intended recipient node but is no longer valid (for the intended recipient) at the moment of arrival. For example, if A is the sender and B is the intended recipient, and we are considering the binding update for A’s \( i \)th handoff, then if B’s next handoff is its \( j \)th, then the binding update makes a belated arrival if and only if \( T_A(i) + \gamma_A(i) + \zeta_A(i) + \delta_{A\rightarrow B}(i,j) > T_B(j) \).

**Definition** A binding update is lost through belated arrival if it makes a belated arrival and is consequently lost.

A node can be lost not necessarily through belated arrival, but through other possible causes of lost binding updates, such as network congestion, node failure, link failure. Conversely, a node can make a belated arrival and not be lost, e.g., if there is an agent in that network that can forward the binding update to the current location of the intended recipient.

Furthermore, the following assumptions are made about binding updates for simultaneous mobility.

1. Binding updates cannot and do not contain information about future moves of the sending node.

2. While two nodes are in a communication session, they get information on the location of the other node only from binding updates, i.e., they do not actively seek the
location of the other node, but only passively accept binding updates.

3. Unless otherwise stated, a binding update is sent directly to the most current known address (i.e., known by the sender) of the intended recipient.

4. Regarding the relative timings of binding update latencies and consecutive handoffs of a receiving mobile node, the timescale of the latencies for binding updates is assumed much smaller than the average inter-handoff time. In other words, \( \delta \ll E(T(i + 1) - T(i)) \), where \( E(.) \) denotes expectation.

5. It is extremely unlikely that a binding update would be sent and the recipient moves twice before the binding update arrives at the previous network of the recipient.

6. It is also assumed that if there is a forwarding location proxy (defined in Section 7.5.4) in the previous network of the recipient, it will correctly forward the binding update to the recipient, which would only have moved once from the previous network.

### 7.5.4 Location proxies and binding update proxies

Here I introduce two kinds of stationary proxies for mobility signaling. These proxies, if used carefully, can help prevent the simultaneous mobility problem. These proxies are abstract proxies - the definitions are more about network functionality than specific implementations as network elements.

Thus, it can be seen how familiar network elements like Home Agents can be described as having certain proxy functions, or can be enhanced for such purposes. The abstraction of these proxies will allow general problems and solutions (related to simultaneous mobility) to be discussed without unnecessarily being bogged down by details of specific mobility protocols. It is also assumed that these proxies should be stationary, not mobile.
7.5.4.1 Location proxy

A location proxy (of a mobile node) is a network function that is used to locate the mobile node. There can be three kinds of location proxies. A forwarding location proxy will forward messages (including binding updates) to the most recent location that it knows for the mobile node. A redirecting location proxy will redirect (e.g., by responding to a query with the latest address) messages to the most recent location that it knows for the mobile node. An intercepting location proxy intercepts, and may act on (forwards or redirects), messages in packets not addressed to it. A non-intercepting location proxy only acts on messages in packets addressed to it. The fundamental differences between the types of proxies are shown in Figure 7.6. It is important to note that whereas a forwarding location proxy will pass along messages toward the final destination, a redirecting location proxy will not do this, but just return location information that can be used to send the message toward the final destination. A location proxy is up-to-date with respect to a particular mobile node, if that mobile node continually updates the proxy with its latest address after each move.

7.5.4.2 Proactive location proxy

A proactive location proxy keeps a copy of mobility related signaling messages (typically, binding updates, but possibly other messages like Care-of-Test Init, when procedures like the return routability are used before the binding update is sent). It keeps the messages for a short while, $X$, after receiving and acting on them (i.e., after redirecting and/or forwarding the message). The messages are kept in the location proxy cache and indexed by the destination node. The messages are discarded after time $X$ has elapsed. If during this period of time, the pro-active location proxy receives a binding update from any one of the destination nodes in its location proxy cache, it either (a) redirects to the new address (if it is a redirecting pro-active location proxy); or (b) forwards the corresponding saved message(s) to it, (if it is a forwarding pro-active location proxy).
CHAPTER 7. OPTIMIZATIONS FOR SIMULTANEOUS MOBILITY

Here are examples of how some of the mobility components within different mobility protocols behave as different kinds of proxies. The Mobile IP home agent is a forwarding location proxy (of the intercepting kind). DNS servers are non-intercepting redirecting location proxies. MIP-LR Home Location Registers are non-intercepting redirecting location proxies. SIP proxy servers are non-intercepting proxies that can be either forwarding location proxies known as proxy servers in SIP terminology or redirecting location proxies known as redirect servers in SIP terminology. Except for DNS servers, the other examples given here are typically used in mobility schemes as up-to-date location proxies. In TCP migration, though, DNS servers are part of the mobility scheme, and so they are up-to-date location proxies in that scheme. Most existing location proxies are not proactive location proxies. However, proactive location proxies (defined in Section 7.5.4) may be useful to provide solutions to the simultaneous mobility problem. The current solutions take into account signaling only. It is assumed that if the mobility signaling gets to its intended recipient, the mobility schemes should, and must, take care of the data traffic correctly. However, in some cases, e.g., with Mobile IP, the Home Agent forwards both signaling and data, whereas in other cases, e.g., MIP-LR and SIPMM, the location registers or SIP servers are only involved in the signaling.

![Figure 7.6: Abstract functions of the location proxies](image)
7.5.4.3 Binding update proxy

A binding update proxy acts on behalf of a mobile node to send its binding updates to its correspondent nodes’ latest addresses. It would typically engage the services of a location proxy of each correspondent node either for redirection to the correspondent node’s latest address, or for forwarding the relevant binding update. At the same time, it also forwards a copy of the message to the latest address it knows for the correspondent node. A mobile node on whose behalf a binding update proxy acts may be referred to as a master of that binding update proxy.

7.5.4.4 Proactive binding update proxy

A proactive binding update proxy not only queries for the latest addresses of the correspondent nodes of its master(s); it also keeps the binding updates for a short while \( \alpha \) after receiving and forwarding them. The messages are kept in the binding update proxy cache and indexed by destination node. The messages are discarded after time \( \alpha \) has elapsed. If during this period of time, the pro-active binding proxy receives a redirection regarding any one of the destination nodes in its binding update proxy cache, it forwards the corresponding saved binding update(s) to it.

7.6 Analyzing the simultaneous mobility problem

I now prove four lemmas that cover the two cases of what happens when there is a pair of handoffs at two mobile nodes, and the binding update from the earlier one arrives (a) later than the time the other node moves (Lemmas 3.1 and 3.2); or (b) earlier than the time the other node moves (Lemmas 3.4 and 3.5).

**Lemma 3.1** Given a pair of consecutive handoffs, one for each of the two mobile nodes in a communication session in normal state, in the absence of location proxies for either mobile node, any binding update sent by the earlier moving mobile node will be lost through
belated arrival, if and only if the binding update does not arrive at the other mobile node before it moves.

**Proof** Suppose without loss of generality that node A moves before node B. Let the handoff times be $T_A(i_0)$ and $T_B(j_0)$, so $T_A(i_0) < T_B(j_0)$. Since node A and node B are in a communication session in normal state up till $T_A(i_0)$, then up till $T_A(i_0)$, anything sent by A arrives at B and vice versa. Since the two handoffs are consecutive, then by definition there is no other handoff in the time interval $[T_A(i_0), T_B(j_0)]$. By our third assumption on binding updates, A’s binding update would be addressed to the latest address it has for B. So for time interval $[T_A(i_0), T_B(j_0)]$, anything sent by A will still be addressed to B’s pre-handoff address, and still arrives at B, including A’s binding update. However, as soon as $t \geq T_B(j_0)$, B would no longer be reachable at its pre-handoff address. In some scenarios, a location proxy for B would be able to prevent the binding update from being lost. However, in the absence of a location proxy, the binding update would just go to B’s previous address and disappear there. Hence, it would be lost through belated arrival.

Conversely, suppose A’s binding update is lost through belated arrival. As shown, for time interval $[T_A(i_0), T_B(j_0)]$, anything sent by A will still be addressed to B’s pre-handoff address, and still arrives at B, including A’s binding update. So if it arrives before $T_B(j_0)$, it will not be lost through belated arrival. Thus, A’s binding update cannot arrive before $T_B(j_0)$. Therefore it arrives after B has moved. This lemma and the next are making assertions about cases where the binding update from the earlier-moving mobile node arrives after the later-moving mobile node has moved. This is shown in Figure 7.7.

**Lemma 3.2** Given a pair of consecutive handoffs, one for each of two mobile nodes in a communication session in normal state (up until the first handoff). In the absence of location proxies for either mobile node (or there might be location proxies but they are not used or involved), the simultaneous mobility problem will occur, if and only if the binding update sent by the earlier moving mobile node does not arrive at the other mobile node before it moves.
CHAPTER 7. OPTIMIZATIONS FOR SIMULTANEOUS MOBILITY

**Proof** I use A and B as the first and second mobile nodes again. If the binding update from A does not arrive at the other node before it moves, then by Lemma 3.1, it is lost through belated arrival. Then, at time \( T_B(j) \), B does not have A’s new address. Since A’s binding update is lost, by the time B is sending its binding update (i.e., \( T_B(j) + \lambda_B(j) + \zeta_B(j) \)) it will send it to A’s previous address. Thus, in the absence of location proxies, B’s binding update will also be lost through belated arrival. So both A’s and B’s binding updates are lost through belated arrival. By definition, the simultaneous mobility problem has occurred.

Conversely, if the simultaneous mobility problem has occurred, then by having both the binding updates having lost through belated arrival, so A’s binding update is lost through belated arrival. From the above proof, the following corollary emerges.

![Diagram](image)

**Figure 7.7: Lemma 1 and 2**

**Corollary 3.3** Given a pair of consecutive handoffs, one for each of two mobile nodes in a communication session in normal state (up until the first handoff), in the absence of location proxies for either mobile node (or there might be location proxies but they are not used or involved), if the binding update from the node that moved first is lost through belated arrival, the binding update from the node that moved second will also be lost through belated arrival.

**Lemma 3.4** Given a pair of consecutive handoffs, one for each of two mobile nodes in a
communication session in normal state (up till the first handoff), the simultaneous mobility problem does not occur if the binding update from the node that moved earlier reaches the other node before that node moves.

**Proof** As in the proof of Lemma 3.1, one can argue that for time interval \([T_A(i_0), T_B(j_0)]\), anything sent by A still arrives at B, including A's binding update. Thus, B can then send its binding update correctly to A's new address after it moves at \(T_B(j_0)\). Therefore, the simultaneous mobility problem does not occur.

This lemma and the next are making assertions about cases where the binding update from the earlier-moving mobile node arrives before the later-moving mobile node has moved. This is shown in Figure 7.8. NB: the converse of Lemma 3.1 is not necessarily true, i.e., one cannot say that if the simultaneous mobility problem does not occur, then the binding update from the node that moved earlier reaches the other node before that node moves. The reason this is not necessarily true is that location proxies could be used, as I will demonstrate later. However, I first extend Lemma 3.4 to the case that location proxies are excluded, where I can make a stronger statement.

**Lemma 3.5** Given a pair of consecutive handoffs, one for each of two mobile nodes in a communication session in normal state (up till the first handoff), in the absence of location proxies for either mobile node (or there might be location proxies but they are not used/involved), the simultaneous mobility problem does not occur if and only if the binding update from the node that moved earlier reaches the other node before that node moves.

**Proof** This has been partially proved in the proof for Lemma 3.4. What remains is to prove that if the simultaneous mobility problem does not occur, the binding update from the node that moved earlier reaches the other node before that node moves. Supposing the simultaneous mobility problem does not occur, that means both binding updates arrive at the other node. B’s binding update therefore cannot be lost through belated arrival, so B must have successfully received A’s binding update. By the 3rd assumption on binding updates, A's binding update could not have been addressed to B’s new location since A moved first.
Given that location proxies are not used, there is no way that B could successfully receive A’s binding update after $T_B(j_0)$. Therefore, A’s binding update must have reached B before $T_B(j_0)$, i.e., before B moved.

![Diagram](image)

**Figure 7.8: Lemma 4 and 5**

**Remark** What about if A’s binding update successfully reaches B before it moves, but B’s binding update does not reach A because A’s next handoff happens before the arrival of B’s binding update? Does Lemma 3.4 break down? No, in that case the lemma, applied to $H_A(i_0)$ and $H_B(j_0)$ would correctly show that the problem is not between those two handoffs, but applied to $H_B(j_0)$ and $H_A(i_0 + 1)$ would correctly show that the simultaneously mobility problem occurs then with B as the earlier moving node.

### 7.7 Probability of simultaneous mobility

In this section, I analyze the probability of simultaneous mobility. In [WDSY03] I introduced a simple mathematical model for estimating the probability of occurrence of simultaneous mobility. I briefly describe in this section.

If the probability that any particular handoff (either separately or both at the same time) suffer from simultaneous mobility problem be $P_0$, and the probability that at least one out
of \( N \) handoffs in a given session suffers from the simultaneous mobility problem be \( P_N \).

Thus \( P_N = 1 - (1 - P_0)^N \).

The system gets into an interrupted state if either of the mobiles or both of the mobiles are subjected to simultaneous mobility problem. Inter-handoff times for mobile1 (A) and mobile2 (B) are \( \lambda_1 \) and \( \lambda_2 \), respectively.

\( \alpha \) is the time taken for A’s binding update to reach B. \( \beta \) is the time taken for B’s binding update to reach A.

- Probability that mobile 1 contributes to the simultaneous mobility problem is \( P_1 = \beta/\lambda_1 \).
- Probability that mobile 2 contributes to the simultaneous mobility problem is \( P_2 = \alpha/\lambda_2 \).

Thus, probability that there is a simultaneous mobility problem due to handoff by mobile 1, mobile 2 or both is as follows

\[
P_0 = P_1 + P_2 - (P_1 \times P_2) = \frac{\beta}{\lambda_1} + \frac{\alpha}{\lambda_2} - \frac{[\beta \times \alpha]}{[\lambda_1 \times \lambda_2]}
\]

If inter-handoff time for both the mobiles are same, thus \( \lambda_1 = \lambda_2 = \lambda \), then

\[
P_0 = \frac{[\alpha + \beta]}{\lambda} - \frac{[\beta \times \alpha]}{\lambda^2}
\]

where \( \alpha \) and \( \beta \) are the amount of time needed for a binding update to reach from A to B and vice-versa and \( \lambda \) is the average inter-handoff time.

Thus, the simultaneous mobility problem is affected by a combination of end-to-end latency of the packet and inter-handoff time. As part of initial results, I have conducted a preliminary analysis of simultaneous mobility of IP hosts for SIP, MIPv6 and MIP-LR-based mobility protocols.

Consider two consecutive handoffs, one each at mobile nodes A and B. According to Lemma 3.2, the simultaneous mobility problem occurs if and only if the binding update from the earlier moving node arrives after the other node has moved. Mathematically, this is written as \( T_A + \gamma + \zeta + \delta_{B \rightarrow A} > T_B \) (if A is the earlier moving node) or \( T_B + \gamma + \zeta + \delta_{B \rightarrow A} > T_A \) (if B is the earlier moving node).

Putting the two inequalities together, the following equation is obtained
\[ T_A - \alpha < T_B < T_A + \beta \] 

(7.3)

where \( \alpha = \gamma + \zeta + \Delta_B \rightarrow A \) and \( \beta = \gamma + \zeta + \Delta_A \rightarrow B \) are convenient short forms.

Then I define the concept of “vulnerability interval” \( \beta + \alpha \), which is the time around a handoff during which the two mobile nodes are vulnerable to the simultaneous mobility problem if another handoff occurs at the other mobile node.

It is reasonable to model the handoff times for A and B as independent Poisson processes. In this model, the intervals between consecutive handoffs at A, \( \Gamma_A(k-1)_A = T_A(k)-T_A(k-1) \), \( \Gamma_A(k) = T_A(k+1)-T_A(k) \) etc., are independent exponentially distributed random values, and similarly for the corresponding intervals between consecutive handoffs at B. Then it is easy to argue that the probability of the simultaneous mobility problem occurring can be estimated as the following equation.

\[
P_0 = \frac{E(\alpha + \beta)}{E(\Gamma)} - \frac{E(\alpha \times \beta)}{E(\Gamma^2)}
\]

(7.4)

If there are \( N \) handoffs occurring at each of the two mobile nodes, then the probability of the simultaneous mobility problem occurring can be estimated by

\[
P_N = 1 - (1 - P_0)^N
\]

(7.5)

Based on experimental measurements, \( E[\alpha+\beta] \) ranges from 50 ms to 500 ms, while \( \lambda \) may range from 5 seconds (movement at vehicular speeds across pico-cells of a few hundred meters in diameter) to 500 seconds or more (larger cells, slower speeds, non-linear movement pattern). \( E(\Gamma) \) is the average value of inter-handoff time that can be equated to \( \lambda \). In Figure 7.9, I plot for approximately this range of \( E[\alpha+\beta] \) and \( \lambda \). Figure 7.9 shows how probability of simultaneous mobility \( P_0 \) is affected due to binding update latency and mean handoff time of the mobile based on the Equation 7.4. Figure 7.9(a) shows that for a given inter-handoff time (500 seconds), probability of failure increases as the one-
way latency of the binding update increases. While Figure 7.9(b) shows that for a given one-way-delay (50 ms) probability of failure decreases as the inter-handoff time decreases. Figure 7.10 shows the probability of simultaneous mobility $P_3$ when the total number of handoff is 3 based on Equation 7.5. For the same values of inter-handoff time and one-way latency, probability of failure due to simultaneous mobility increases as the number of handoff is increased to 3. As expected, as shown in Figure 7.9(a) the highest probability of simultaneous mobility is when one-way packet latency is the largest and average inter-handoff time is smallest. Thus, the effect of the simultaneous mobility problem could be quite significant. Without fixing the problem, the binding updates of both the Mobile Hosts would never reach the other host, and so the connection would be lost. It is important to note that this analysis is optimistic as it is assumed that the binding update from A to B would not be lost. Since there is a small chance that this binding update might also be lost, the values computed in this analysis could be viewed as merely providing a lower bound on the likelihood of the simultaneous mobility occurring. For the case of simultaneous mobility during session initiation signaling, the probability of failure also depends upon the mobility rate of the mobiles. From the lab measurements, it takes about 200-300 ms to complete the whole session initiation signaling sequence. A complete registration will take about 150 ms. Hence, the probability of simultaneous mobility occurring during session initiation signaling is non-trivial.

7.8 Solutions

Kravets et al. have hinted at the benefits of some kind of proxies in fixed locations to enable communication to continue even when both end-hosts move simultaneously [KCM01]. However, as far as I know, previous work has not analyzed the problem to the level I have described in Section 7.3 nor has a systematic analysis of solutions applied to a range of mobility protocols been previously provided. I describe some of the solution mechanisms
in this section. Solution mechanisms are specific mechanisms and functions that could be used (typically in conjunction with other mechanisms and functions) to provide solution for the simultaneous mobility problem for a given mobility protocol.

The proposed techniques can broadly be classified into three: soft-handoff, sender-based, receiver-based mechanisms.

### 7.8.1 Soft handoff

Suppose a mobile node can have more than one valid IP address. In such cases, the mobile can have two bindings associated with its home address. This is sometimes referred to as simultaneous mobility bindings, and should not be confused with the simultaneous mobility problem. I call it as the soft handoff approach, since it is similar to soft handoffs in CDMA mobile systems [WL97]. The idea is that if the previous IP address and new IP address can both be used to reach the mobile node during the handoff process, that can solve the simultaneous mobility problem. Binding updates sent to the previous IP address would arrive correctly. Although, most of the current operating system can support multiple concurrent IP addresses for the wireless interface (s), this is not a universally applicable solution, for
the following reasons:

1. The operating system should be able to support multiple concurrent IP addresses for the wireless interface(s).

2. The network interface of the mobile needs to be able to connect simultaneously to multiple base stations that may belong to two different subnets. CDMA technology does provide this ability whereby the network interface can connect to both the network access points simultaneously. However, this is limited to CDMA access technology only.

3. Resource utilization is not efficient because of redundant allocation of bandwidth resources (on both communication paths) during the period of simultaneous mobility bindings.

4. It is also important to make sure that the simultaneous mobility bindings need to be active to ensure that no problems will occur during simultaneous mobility. The longer it waits, the more this specific scheme uses valuable network resources redundantly. Since this solution must work for any radio network technology, use of simultaneous
7.8.2 Receiver-side mechanisms

Receiver-side mechanisms typically can be deployed in the previous network or home network of the receiver and act on behalf of a receiver to help it to be located. Retransmission, forwarding, redirecting, proactive-forwarding and proactive-redirecting are some of the mechanisms that have been analyzed for two mobility protocols such as SIP and MIP-LR. Details of the proposed mechanisms and results can be found in references [WDSY03], [DWDSY07]. Following is a list of receiver-side mechanisms.

7.8.2.1 Timer based retransmissions

One could imagine a forwarding location proxy automatically retransmitting a binding update if it has not gotten confirmation that the binding update was successfully received by the intended receiver. This location proxy could be located in the receiver’s home network or a visited network (e.g., the previous network or latest network). Location proxies that retransmit based on timeouts are similar to proactive location proxies in that both need to store the message briefly, to retransmit if necessary. The difference is in the conditions for retransmission. A proactive proxy retransmits as soon as a new address is obtained, whereas a timer-based retransmission may be too slow. In the existing implementations, a stateful SIP proxy could retransmit binding updates (re-INVITE) after the expiration of a timer. This could be located in the home network of the receiver or in the visited network of the receiver. In order to ensure that the re-INVITE message (and other signaling) goes through this server, the Record-Route option could be used in the initial INVITE message to add the server to the signaling path.
7.8.2.2 Regular passive forwarding

Forwarding mechanisms on the receiver side allow binding updates to be forwarded from a location proxy in the previous network to the correct new location of the receiver. Forwarding mechanisms from a previous network may also forward data packets (since the location proxy is forwarding packets, anyway, it might as well forward data packets). One could also imagine such a location proxy in the receiver’s home network as well.

Here are some existing implementation where the forwarding agents are in the receiver’s previous network. In MIP-RO (Mobile IP with Route Optimization), the previous Foreign Agent serves this role by forwarding the data packets. Unfortunately, this ability is missing from MIPv6, perhaps because no Foreign Agents are used in MIPv6 (and so, there is no natural forwarding agent present in the previous network). Thus, the problem of simultaneous mobility remains in MIPv6. Similarly, SIP-based mobility management and MIP-LR lack such functionality. Similarly, in some examples, the forwarding agents could be in receiver’s home network. An example is the Home Agent in MIPv4 and MIPv6. A SIP server in the receiver’s home network could also serve in this capacity, e.g., if it places itself in the signaling path using the Record Route field in the initial INVITE message.

7.8.2.3 Pro-active forwarding

Regular passive forwarding may be insufficient to solve the simultaneous mobility problem. A pro-active forwarding location proxy may help where forwarding takes place before the handoff.

7.8.2.4 Redirecting

Redirecting mechanisms on the receiver side can help to get messages like binding updates to the right place. There are some existing implementations where the redirecting agents are placed in the receiver's previous network and home network. In MIP-RO, the previous Foreign Agent serves this role. In MIP-LR, the HLR does this, but only before a media
session begins. Then, it is not involved in control signaling during the communications session. Thus, it does not count as a proper implementation of a solution mechanism for the simultaneous mobility problem.

7.8.2.5 Pro-active redirecting

Regular redirecting may be insufficient to solve the simultaneous mobility problem. A pro-active redirecting location proxy may help in some cases where there is a probability of handover to a number of target networks.

7.8.3 Sender-side mechanisms

Sender-side mechanism typically can be deployed in the home network of the sender, or in the sender itself, and act on behalf of the sender to try to reach the receiver. The receiver may be moving simultaneously with the sender and may not receive the binding update if none of these mechanisms are used. Following is a list of sender-side mechanisms.

7.8.3.1 Timer based retransmissions

A forwarding location proxy automatically retransmits a binding update if it has not gotten confirmation that the binding update was successfully received by the intended receiver. This location proxy could be located in the sender’s home network or even in the sender itself (for end-to-end retransmission).

There can be several existing implementations. A stateful SIP server could retransmit binding updates (re-INVITE) after the expiration of a timer.

7.8.3.2 Forwarding (regular, passive type)

Forwarding mechanisms in the sender’s home network can help to get messages like binding updates to the right place, but are probably less useful than those on the receiver side
because of the time spent by the forwarded signals due to distance between sender’s home network and receiver.

7.8.3.3 Pro-active forwarding

Regular, passive forwarding may be insufficient to solve the simultaneous mobility problem. A pro-active binding update proxy may help in some solutions, where it attempts to find the most current location of the receiving node and re-try the forwarding there.

7.8.3.4 Redirecting

Redirecting mechanisms in the sender’s home network can help to get messages like binding updates to the right place, but are probably less useful than those on the receiver side.

7.8.3.5 Pro-active redirecting

Regular redirecting may be insufficient to solve the simultaneous mobility problem. A pro-active redirecting location proxy may help in some cases when the target network is not deterministic.

Table 7.1 shows the applicability of these solution mechanisms for different mobility protocols. Table 7.2 shows the strengths and weaknesses of different solutions.

7.9 Application of solution mechanisms

In this section, I describe how the solution mechanisms can be applied to different mobility protocols. I illustrate its applicability to few mobility protocols, namely Mobile IPv6, SIPMM (SIP-based mobility) and MIP-LR.
Table 7.1: Comparison of solutions for simultaneous mobility

<table>
<thead>
<tr>
<th>Solutions</th>
<th>MIP RO</th>
<th>MIPv6</th>
<th>SIPMM</th>
<th>MIP LR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prior network</td>
<td>Re-transmission</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Forwarding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pro-active forwarding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Receiver-side</td>
<td>Redirecting</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pro-active redirecting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home network</td>
<td>Retransmission</td>
<td>possible</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Forwarding</td>
<td>Yes</td>
<td>Yes</td>
<td>possible</td>
</tr>
<tr>
<td></td>
<td>Pro-active forwarding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Redirecting</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Pro-active redirecting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sender-side</td>
<td>Retransmission</td>
<td>possible</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home network</td>
<td>Forwarding</td>
<td>possible</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Proactive forwarding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Redirecting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Proactive redirecting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>At sender</td>
<td>Retransmission</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 7.2: Strength and weakness of different solutions

<table>
<thead>
<tr>
<th>Solutions</th>
<th>Strengths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timer-based retransmission of lost messages</td>
<td>Can be easily implemented with a timer</td>
</tr>
<tr>
<td>Simultaneous bindings</td>
<td>No significant increase in handoff latency to solve simultaneous mobility</td>
</tr>
<tr>
<td>Forwarding mechanisms from previous network</td>
<td>Effective if not just data but also signaling is forwarded.</td>
</tr>
<tr>
<td>Stationary proxies</td>
<td>Completely eliminates vulnerability interval</td>
</tr>
<tr>
<td></td>
<td>(a) Difficulty of choosing good time-out values; (b) retransmissions may also be lost</td>
</tr>
<tr>
<td></td>
<td>(a) Not supported by all wireless networks; (b) redundant resource utilization; and (c) not clear how long to keep active the simultaneous bindings</td>
</tr>
<tr>
<td></td>
<td>Handoff latency is slightly increased because of the forwarding from the previous network; still vulnerable to simultaneous mobility, but with reduced vulnerability interval</td>
</tr>
<tr>
<td></td>
<td>Handoff latency is increased</td>
</tr>
</tbody>
</table>

### 7.9.1 Mobile IPv6

Three different solution mechanisms are considered to take care of simultaneous mobility problem in MIPv6. These are described as follows.

#### 7.9.1.1 Forwarding proxy in previous network

As described earlier, for MIP-RO, Foreign Agents in the previous network act as forwarding proxies. However, for MIPv6, Foreign Agents are not used. Thus, ordinary routers in the previous network need to be augmented with forwarding proxy functionality. This would involve significant challenges and modifications to MIPv6. For example, a mechanism would be needed to securely update the router with the latest IP address of the mobile node. However, getting ordinary IPv6 routers to perform this kind of forwarding might pose deployment bottleneck and thus some kind of agent might need to be introduced.
7.9.1.2 Combination of sender-side and receiver-side mechanisms

I proposed a combination of sender-side proactive binding update proxies and receiver-side pro-active redirecting location proxies as a general solution in [WDY+03]. Here I propose that the same technique can also be applied to MIPv6. The Home Agents of the sender and receiver, respectively, can serve as the pro-active binding update proxy and pro-active redirecting location proxy. Return routability has to be modified so that CTI message goes through the sender’s Home Agent. Another modification to MIPv6 is that the binding update must first be reverse-tunneled to a mobile node’s own Home Agent before being forwarded to the correspondent node. The revised MIPv6 update procedure will work as follows. Let there be two mobile nodes A and B. A sends its CTI and binding update messages to B through A’s Home Agent, rather than directly to B’s care-of address. However, A’s Home Agent will then forward these messages to B at B’s care-of address. A’s Home Agent, acting as a pro-active binding update proxy will also keep a copy of any such message for a period $\tau$. It would then query B’s Home Agent (a pro-active location proxy for B) to find out if B has a newer address. B’s Home Agent responds immediately but keeps a copy of the query for a period $\rho$. If before this period is over, B’s Home Agent receives a registration for B at a new address, it pro-actively corrects its query. A’s Home Agent will then forward the message to the new address. This solution is illustrated in Figure 7.11.

The selection of $\tau$ and $\rho$ should be carefully chosen based on reasonable estimates of the appropriate signaling and computational delays of the network. It is clear that $\tau > \rho$, so A’s Home Agent can respond to any query response correction from B’s Home Agent.

7.9.1.3 Receiver-side mechanisms only

Two solutions discussed so far are not good since they require significant changes to MIPv6. A more MIPv6-centric solution is preferable. I therefore consider a solution where just the receiver’s Home Agent is involved. A is the sender (of the CTI, HTI or binding update).
A sends all these control messages to B using B’s *home address*, thus forcing B’s Home Agent to be involved. B’s Home Agent will act as a pro-active forwarding location proxy (a slight modification from its usual role as a forwarding location proxy), forwarding the control message to B as usual, but keeping a copy of it for time $\tau$. If it gets any binding updates from B during that time, it pro-actively forwards the message to B. This solution is shown in Figure 7.12. However, it requires some modifications to be made at the home agent. Home agents need to behave like proactive forwarding location proxy in addition to behaving like a forwarding location proxy. The main modification to mobile nodes implementing this solution is also small - to send the CTI and binding update to the home address of the correspondent node instead of directly to its care-of address.

![Diagram](image)

**Figure 7.11: Sender and receiver side mechanism**

### 7.9.1.4 Evaluation

It is clear that the third solution, with receiver-side mechanisms only (sending messages to the other node’s Home Agent) is the cleanest solution with the least changes to MIPv6. The adding (in the second solution) of a query and response capability to Home Agents is quite a drastic change for MIPv6. After the removal of Foreign Agents in going from
MIPv4 to MIPv6, the adding of a forwarding proxy in the previous network with substantial functionality (in the first solution) is not desirable as it increases delay.

7.9.2 MIP-LR

In this section, I describe how these solution mechanisms can be applied to take care of simultaneous mobility problem in MIP-LR.

7.9.2.1 Forwarding proxy in previous network

There are two types of MIP-LR: one with Foreign Agents [JRG98], and one without [JRY99]. The version without Foreign Agents uses advertisement agents. The forwarding proxies use interceptor function that intercepts the binding update and sends it to the new address of the mobile.

For the purposes of placement of the interceptor function, it does not matter whether Foreign Agents or advertisement agents are in use. The point is that there is some kind of agent in each of the foreign networks, and the interceptor function can be placed here. MIP-LR needs modification so the mobile node sends a binding update to the Foreign Agent (or Advertisement Agent) in the previous subnet as soon as it obtains its new IP address.
7.9.2.2 Sender-side and receiver-side mechanisms

In this solution, the binding update sent by a mobile node to its HLR has a list of correspondent nodes and their addresses. The HLR, which already performs the role of a redirecting location proxy, is enhanced to be a pro-active redirecting location proxy. It also acts as a pro-active binding update proxy, since it already obtains the current binding information as part of MIP-LR updating after each handoff. In order to do this, the HLRs must be enhanced to pro-actively retransmit binding updates and to query other HLRs for correspondent nodes’ addresses. In order to minimize changes to MIP-LR, the HLR-initiated binding updates are only sent when necessary, i.e., when the queries return a newer address for a correspondent node than the one provided by the mobile node.

7.9.2.3 Evaluation

It is recommended not to consider a solution using only receiver-side mechanisms as has been done for MIPv6. This is because the HLR would then have to become a pro-active forwarding proxy. Such a change is too much for MIP-LR, one of whose points is that no forwarding location proxies are used (but multiple replicated HLRs are used).

7.9.3 SIP-based mobility

SIP allows much flexibility in placement and usage of SIP servers in the signaling path between two mobile nodes. The Record-Route field is an optional field in the SIP header that allows SIP servers to remain in the signaling path between two SIP end-nodes during a communications session. It is assumed that often there will be a SIP server in each mobile node’s home network that serves as an up-to-date location proxy for it. An up-to-date location proxy keeps a record of the most recent location of the mobile. SIP also provides inbuilt retransmission technique.
7.9.3.1 Timer-based retransmission

SIP has an in-built retransmission capability, where messages are retransmitted after a time-out if the acknowledgement is not received. During mid-session mobility, a re-INVITE may get lost even if it goes through the SIP server that keeps the most recent registration status of the destination. However, SIP allows for automatic retransmissions of INVITEs (including re-INVITEs) by SIP UAs if a response (OK message) is not received within a specified time. Stateful SIP servers can also retransmit (re-)INVITEs.

One problem with timer-based retransmissions is that significant latency could be added to the handoff when messages are lost due to simultaneous mobility.

Another problem is that there is no guarantee that the retransmission would not also be lost. For example, the retransmission may be sent directly to the old address of the Correspondent Host, bypassing network elements (e.g., the relevant SIP servers, or the Home Agent of the Correspondent Host) that know the latest address of the Correspondent Host. Figure 7.13 shows how a server assisted retransmission technique can be useful to solve the simultaneous mobility problem.

Figure 7.13: Server-assisted retransmission mechanism
7.9.3.2 Forwarding proxy in previous network

Like the first proposed solution for MIPv6, I consider adding a forwarding location proxy to the previous network of a mobile node. For SIP-based mobility, the most natural choice of the forwarding proxy could be an entity similar to an RTP (Real-time Transport Protocol) translator [Sch], since these are already used to forward media traffic, among other things. Having a SIP server in the previous network that can forward this signaling is one solution. However, if one keeps on adding SIP servers in previous networks to the signaling path (using Record-Route), the signaling path becomes inefficient as the mobile node moves and the signaling path goes through more and more SIP servers in previous networks. Although the existing RTP translator can only forward data traffic from the previous network, similar mechanism can be deployed that can intercept the signaling traffic and forward it to the new location of the mobile. For receiving update signaling, the RTP translator can be enhanced to act as a SIP signaling translator without generally translating the RTP.

7.9.3.3 Receiver-side mechanisms

For SIP-based mobility scheme, the receiver-side home network SIP server already has some location proxy functionality that can be modified to act as pro-active location proxy. I first consider the case where it acts as a forwarding location proxy (it can also be a redirecting location proxy, which I will consider in the next paragraph). The SIP server immediately retransmits the re-INVITE upon receiving a REGISTER message from the destination of a pending re-INVITE. This is basically the same solution as the third one for MIPv6 (the preferred solution). Hence, the Figure 7.11 also applies here, by replacing B’s Home Agent with B’s home network SIP server, and binding update with re-INVITE. A difference is that, in order to get the SIP server to be in the signaling path for the re-INVITE request, the Record-Route field can be used. No modifications are needed on the mobile nodes, since SIP conveniently already has the Record-Route feature, unlike with MIPv6, where they have to be slightly modified to send control signaling to the home address of
the other node rather than directly to the care-of address. The conversion of the SIP server to a pro-active one is in some ways easier than the conversion of a MIPv6 Home Agent to a pro-active forwarding proxy. This is because there is already the notion of stateful SIP servers that can retransmit messages like the re-INVITE if no acknowledgement has been received by the time a timer expires.

### 7.9.3.4 Sender-side and receiver-side mechanisms

If the home network SIP server is modified to become a pro-active redirecting location proxy, instead of a pro-active forwarding location proxy, then it needs to interact with a pro-active forwarding proxy closer to the sender in the signaling path. In particular, when there is a SIP server in each mobile node’s home network, there needs to be a pro-active forwarding proxy in the sender’s home network. This is similar to the chosen solution for MIP-LR, where the two HLRs were involved in this way. One difference is that the Record-Route feature will be needed to keep both SIP servers in the signaling path.

### 7.9.3.5 Evaluation

It would appear that either the 2nd or 3rd solution is equally simple to implement, given that SIP servers of both types (forwarding and redirecting) are available. With MIPv6, on the other hand, the clear preference was for the receiver-side solution, given that Home Agents are forwarding location proxies only.

### 7.10 Discussion

Although the original MIP did not suffer from the simultaneous mobility problem, newer mobility management protocols like MIP-LR, SIP-based mobility and MIPv6 do face this problem. In this chapter, I have identified the problem of simultaneous mobility, introduced a new analytical framework, and then used the framework to prove some new theorems, an-
analyze solution mechanisms, and propose and compare solutions for simultaneous mobility for MIPv6, SIP-based mobility and MIP-LR. I also conduct a probability analysis on the likelihood of occurrence of simultaneous mobility.

The problem is further compounded by the expected rise in popularity of at least two of the three protocols we considered, namely MIPv6 and SIP-based mobility. Additionally, with the rise of smaller pico-cells in certain segments of the wireless market and higher mobility rates, there may be more frequent occurrences of simultaneous mobility in the future. I have explored a number of approaches to deal with the simultaneous mobility problem. In some of the protocols, there is existing functionality that partially helps solve the simultaneous mobility problem, or that can be modified to handle simultaneous mobility. For example, with SIP-based mobility, forwarding entity similar to RTP can be used to forward signaling, including binding updates, that might have been sent to the previous network.

Most recently, I have introduced the effect of simultaneous mobility problem for MIPv6 in the MEXT working group within the IETF. Realizing that there is lack of solution to deal with the simultaneous mobility, a new section has been added in RFC 3775 bis [JPA09] to take care of simultaneous mobility problem arising out of return routability procedures. As an alternative to the solutions discussed in section 7.9, following modification has been added to the draft.

In some scenarios, such as simultaneous mobility, where both correspondent host and mobile host move at the same time, or in the case where the correspondent node reboots and loses data, route optimization may not complete, or relevant data in the binding cache might be lost.

1. Return routability signaling MUST be sent to the correspondent node’s home address if it has one (i.e., not to the correspondent node’s care-of address if the correspondent node is also mobile.)
2. If Return routability signaling timed out after `MAX_RO_FAILURE` attempts, the mobile node MUST revert to sending packets to the correspondent node’s home address through its home agent.

3. The mobile node may run the bidirectional tunneling in parallel with the return routability procedure until it is successful. Exponential backoff SHOULD be used for retransmission of return routability messages.

The return routability procedure may be triggered by movement of the mobile node or by sustained loss of end-to-end communication with a correspondent node (e.g. based on indications from upper-layers) that has been using a route optimized connection to the mobile node. If such indications are received, the mobile node MAY revert to bi-directional tunneling while re-starting the return routability procedure.