Spring 2003

A Framework of Cooperating Agents Hierarchies for Local-Area Mobility Support

Ayman Adel Abdel Hamid
Old Dominion University

A FRAMEWORK OF COOPERATING AGENTS HIERARCHIES FOR LOCAL-AREA MOBILITY SUPPORT

by

Ayman Adel Abdel Hamid
M.Sc. Computer Science, August 1998, Alexandria University, Egypt
B.Sc. Computer Science, June 1993, Alexandria University, Egypt

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirement for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
May 2003

Approved by:

Hussein Abdel-Wahab (Director)

Kurt Maly (Member)

Christian Wild (Member)

Larry Wilson (Member)

Frederic McKenzie (Member)

ABSTRACT

A FRAMEWORK OF COOPERATING AGENTS HIERARCHIES
FOR LOCAL-AREA MOBILITY SUPPORT

Ayman Adel Abdel Hamid
Old Dominion University, 2003
Director: Dr. Hussein Abdel-Wahab

Host mobility creates a routing problem in the Internet, where an IP address reflects the network's point of attachment. Mobile IP, relying on a mapping between a home address and a care-of address, and a home registration process, is widely accepted as a solution for the host mobility problem in wide-area mobility scenarios. However, its home registration requirement, upon each change of point of attachment, makes it unsuitable to handle local-area mobility, resulting in large handoff latencies, increased packet loss, and disrupted services. In this dissertation, we introduce a local-area mobility support framework for IPv4 based on the deployment of multiple cooperating mobility agents hierarchies in the foreign domain. First, we introduce a hierarchy model offering a backward compatible mode to service legacy mobile hosts, unaware of local-area mobility extensions. Second, for intra-hierarchy handoffs, we identify several design deficiencies within the current Mobile IP hierarchy extension proposal, and present an enhanced regional registration framework for local handoffs that encompasses a replay protection identification value dissemination mechanism. In addition, we present two novel registration frameworks for home registrations involving local handoffs, in which we identify the dual nature of such registrations, and attempt to emphasize the local handoff aspect. One technique, maintains tunneling of data packets to the MH (Mobile Host) through an old path until a home registration reply is received to set up the new path. In contrast, the other technique adopts a more proactive bold approach in switching immediately to the new path resulting in a reduction of the handoff latency. Third, for inter-hierarchy handoffs, we present a scalable, configurable, and cooperation based

framework between mobility agents hierarchies to reduce the handoffs latencies. An attempt is made to exploit the expected network proximity between hierarchies within the foreign domain, and maintain a mobile host's home-registered care-of address unchanged while within the same foreign domain. In addition, the involved registration signaling design requires a reduced number of security associations between mobility agents belonging to different hierarchies, and copes with the fact that the mobile host's home-registered care-of address might not be reachable. The proposed mechanisms are evaluated qualitatively and analytically, and their performance is investigated through network simulations using our extension of Columbia University's IP Micro-mobility software (CIMS), an *ns*-2 source code extension. The intra-hierarchy handoffs processing mechanisms achieve a sizable reduction in UDP packet loss, and maintain better TCP throughput in the case of a distant home agent, versus a base Mobile IP implementation. Moreover, the cooperation-based intra-hierarchy handoffs processing techniques are successful in achieving the same results (UDP packet loss reduction, and better TCP throughput) versus a non-cooperative approach for a distant home agent. Fourth, The need to evaluate the performance of our mobility support mechanisms prompted us to design and implement a local-area mobility network simulation framework. We extended CIMS to include the capability to model a true foreign domain with multiple mobility agents' hierarchies with an arbitrary number of levels. In addition, our implementation encompasses several implementation enhancements including the support for regional registrations, periodical home registrations, and the smooth handoff mechanism.

ACKNOWLEDGMENTS

This dissertation would not have been successfully completed without the contributions of a number of people. My deepest gratitude and appreciation are due to Dr. Hussein Abdel-Wahab for his continued guidance and support throughout this work. I am indebted to him for long hours of motivating discussions, constructive feedback, and thorough review of this dissertation. In addition, I would like to extend my thanks to all my committee members: Dr. Kurt Maly, Dr. Chris Wild, Dr. Larry Wilson, and Dr. Frederick McKenzie for their guidance and fruitful feedback concerning this dissertation.

My motivating supportive family has always been a source of encouragement during my study. My wife, Sahar, my daughter, Rana, and my son, Mohamed, inspired me to achieve my goal. My parents and brother always supported me during this journey, although that meant we would be separated thousands of miles for extended periods of time. My utmost thanks for my entire family for their everlasting support and encouragement.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

SECTION I

INTRODUCTION

The increasing availability of wireless communication technologies and the proliferation of portable computing devices have made realistic a mobile computing paradigm: *users, on the move, can seamlessly access network services and resources, from any-where, at any time*. However, host mobility presents a challenge for the TCP/IP based Internet, since an IP address reflects a host's point of attachment to the network. The Internet Engineering Task Force (IETF) standardized Mobile IPv4 [43] and is in the process of issuing a standard for Mobile IPv6 [33] as network layer solutions for the host mobility problem for IPv4 [50] and IPv6 [20], respectively. Mobile IP can handle wide-area mobility (macro-mobility) and local-area mobility (micro-mobility), although more suited to handle the former, since a mobile host is required to inform its, possibly distant, home network whenever it changes its point of network attachment. Such requirement results in unnecessary large protocol-signaling overhead, large handoff latencies, and potential packet loss in the local-area mobility scenario, which disrupts ongoing connections directly affecting a mobile host's applications and services. The resulting overhead and handoff latencies are even higher when the mobile host experiences high handoff frequencies, e.g., in wireless networks with small size cells, where the mobile host crosses cell boundaries more frequently. In this dissertation, we present our view and efforts in developing a network layer mobility support solution, within the Mobile IP framework, capable of efficiently handling local-area mobility.

1.1 Overview

Host mobility refers to the function of allowing a mobile host to change its point of attachment to the network, without interrupting IP packet delivery to/from that host [39]. Host mobility presents a challenge for the TCP/IP based Internet, since an IP address reflects a host's point of attachment to the network. During an active TCP session, if the

---

The journal model for this dissertation is the IEEE/ACM Transactions on Networking.

source IP address, or the destination IP addresses change, due to a change of point of attachment, the TCP session breaks. If no special handling is provided to deal with host mobility, packets addressed to a mobile host will be routed to the mobile host's home network, not to its current location. This problem occurs because an IP address serves a dual purpose: a *routing directive* in the network layer and an *end point identifier* in the transport layer [7].

The Primary design goal of mobile host protocols is to allow true mobile operation, so that the mobile host can remain in almost continuous contact with the network resources needed by its applications. Using these protocols, neither the system, nor any of the applications running on the system need to be reinitialized or restarted, even when network connectivity is frequently broken and reestablished at new points of attachment [44]. A change of access point while connectivity is maintained is typically called a *handoff*. Note that solutions that require mobile hosts to be restarted after migration support *portability* and not *mobility*.

Several solutions have been proposed for the host mobility problem, and some attempts have been made to contrast and compare such solutions [4], [23], [56]. In general, such solutions can be classified as *network layer*, *application layer*, and *end-to-end* solutions. Network layer solutions have the advantage of being fully transparent to upper protocol layers but require modifying the deployed IP base. Such solutions generally adopt a two-level addressing architecture with a *home address* and a *care-of address*. A recent survey of network layer mobility solutions can be found in [7] where the key mechanisms of any network layer solution have been identified. The identified mechanisms include an *address translation mechanism* to map the home address to the care-of address, a *packet forwarding mechanism* to tunnel the packets destined to the home address to the location of the care-of address, and a *location management mechanism* to update the mobile host's location. An application layer solution for the host mobility problem has been proposed in [65]. The solution is based on the Session Initiation Protocol (SIP) [58] which is an application-layer protocol used to set up and tear down unicast and multicast multimedia sessions. This approach aims at efficiently supporting real-time communication by providing mobility support through SIP, but proposes using Mobile IP for TCP connections. Recently, an end-to-end approach to host

mobility [59] has been proposed exploiting the secure dynamic updates feature within the Domain Name System (DNS) in order to update a mobile host's current point of attachment at its home network. In addition, a new end-to-end TCP option has been introduced to support secure established connection migration while faced with an IP address change.

Mobile IP [43], [33] presents a network layer solution for the host mobility problem in the Internet for both wired and wireless networks. For wireless networks, it assumes that the *Mobile Host* (MH) can communicate over a wireless link with a *Base Station* (BS), which is statically connected through a fixed wired infrastructure to the Internet.

Mobile IPv4 [43] deploys *Mobility Agents* (MA) in the home network and the visited network. The MH is associated with two IP addresses: Its permanent home IP address which serves as an end point identifier, and a transient care-of IP address which reflects its current point of attachment, and serves as a routing directive at the network layer. The care-of address can be the address of a *Foreign Agent* (FA) in the visited network, or can be a *co-located care-of address*, which the mobile host may dynamically acquire on the visited network through the Dynamic Host Configuration Protocol (DHCP) for example. The FA is a router in the foreign network that acts as a mobility agent. Whenever a mobile host is away from home, it registers its current care-of address with its *Home Agent* (HA) and is responsible for renewing such home mobility binding. The HA is a router that acts as a mobility agent in the home network, and intercepts any datagrams destined to the mobile host's home address, and tunnels them to the registered care-of address. A host in the Internet communicating with the MH is termed a *Correspondent Host* (CH). Mobile IPv4, in its base form, suffers several drawbacks including the requirement that the MH informs its HA at every change of care-of address, and the routing of data packets from a CH to the MH's HA, which in turn tunnels such packets to the current care-of address, while data packets originating from the MH are routed directly using normal IP routing. The latter drawback is termed *triangular routing*. The route optimization enhancement [47] attempts to alleviate such drawback by introducing mechanisms to inform a CH of the MH's current care-of address. In addition, it includes a *smooth handoff mechanism* by which the old FA is informed about the MH's new FA in order to reduce potential packet loss at the old FA.

Mobile IPv6 [33] only deploys a HA in the home network since foreign agents are no longer needed because of IPv6 features which allow mobile nodes to operate in any location without any special support from local routers. The MH informs the HA and its corresponding hosts about its current care-of address through a binding update mechanism. Although base Mobile IPv6 does not deploy mobility agents in the foreign domain, several extensions have been proposed to reduce the number of exchanged binding updates and improve the handoff delay by deploying mobility agents in the foreign domain, e.g., Hierarchical Mobile IPv6 (HMIPv6) from INRIA [14], and the Mobile IPv6 regional registration approach from Nokia [38].

A special class of network layer solutions takes advantage of multicast technology [19], which provides a mechanism for location independent addressing and packet delivery to a group of hosts that subscribe to a multicast group. In addition, it introduces efficient mechanisms for hosts to dynamically join or leave a multicast group. As stated earlier, a network layer solution for the host mobility problem involves specifying mechanisms for address translation, packet forwarding, and location management of mobile hosts. Recognizing the similarity in the fundamental nature between the two problems, Mysore and Bharghavan [41] suggest using multicast communication as the sole mechanism to provide addressing and routing services for mobile hosts. In contrast, the Deadulus approach [6] uses multicasting from the HA to the BSs in the vicinity of the MH to achieve fast handoffs and to reduce packet loss during handoffs. Hence, multicasting is used in a more restricted fashion while preserving the two-level addressing architecture with the care-of address being a multicast address. The major drawback of such solutions is the reliance on the ubiquitous deployment of multicast technology and supporting mechanisms, e.g., multicast address allocation schemes [63].

1.2 Motivation and Objective

One of the main drawbacks of base Mobile IP is being unsuitable to handle local-area mobility (Intra-domain mobility, mobility within a defined local domain, or local coverage area) because of the requirement that the MH must inform the home network upon a change of care-of address. Such requirement induces large protocol signaling overhead with a possibly distant HA, resulting in large handoff latencies, increasing the

possibility of potential packet loss at the old care-of address until the HA is informed of the new care-of address. Such drawback has prompted researchers to design host mobility protocols more suited to better handle a local-area mobility pattern (Intra-domain mobility) while relying on base Mobile IP to handle the wide-area mobility scenario (Inter-domain mobility). A common idea exploited in all such protocols is to move some of the HA functionality to a designated mobility agent located within the local domain that the MH is currently visiting. Consequently, a change of care-of address results in protocol signaling confined to the local area at hand, reducing handoff latencies and potential packet loss. Such local-area mobility management has been recently denoted as *Localized Mobility Management* (LMM), and its main requirements have been identified in [68].

A number of proposals, within the Mobile IP framework, exist to handle local-area mobility, without incurring any large handoff latencies, e.g. [21], [30], and [38]. Other researchers have optimized their local-area mobility solutions towards the wireless network environment, e.g., [9] and [55]. In addition, Cellular IP [64] targets the local-area mobility problem in an IP based cellular network

The regional registration approach [30], a LMM scheme for Mobile IPv4, deploys one or more foreign agent hierarchies within the foreign domain. The presence of a mobility agents hierarchy creates an old path and new path (and hence, an old FA and a new FA) towards the MH during a handoff, hence permitting the closest capable foreign agent, the *crossover FA*[1], in the hierarchy to respond and handle the MH's handoff, in contrast to a distant HA managing each individual handoff. Fig. 1 depicts an example of a MH's handoff within a foreign domain deploying a single FA hierarchy. We analyzed and modeled the operation of the regional registration approach for Mobile IPv4, which enabled us to identify potential race conditions and shortcomings within its registration processing framework, e.g., its full dependence on the smooth handoff mechanism [47] being supported by both the MH and the FA hierarchy, and the lack of an efficient scalable approach to handle inter-hierarchy handoffs. Modeling and overview of the

[1] The crossover FA is the first common FA between the old and new path.

regional registration approach for Mobile IPv4 is presented in section II, while its critique and shortcomings are presented in section III.

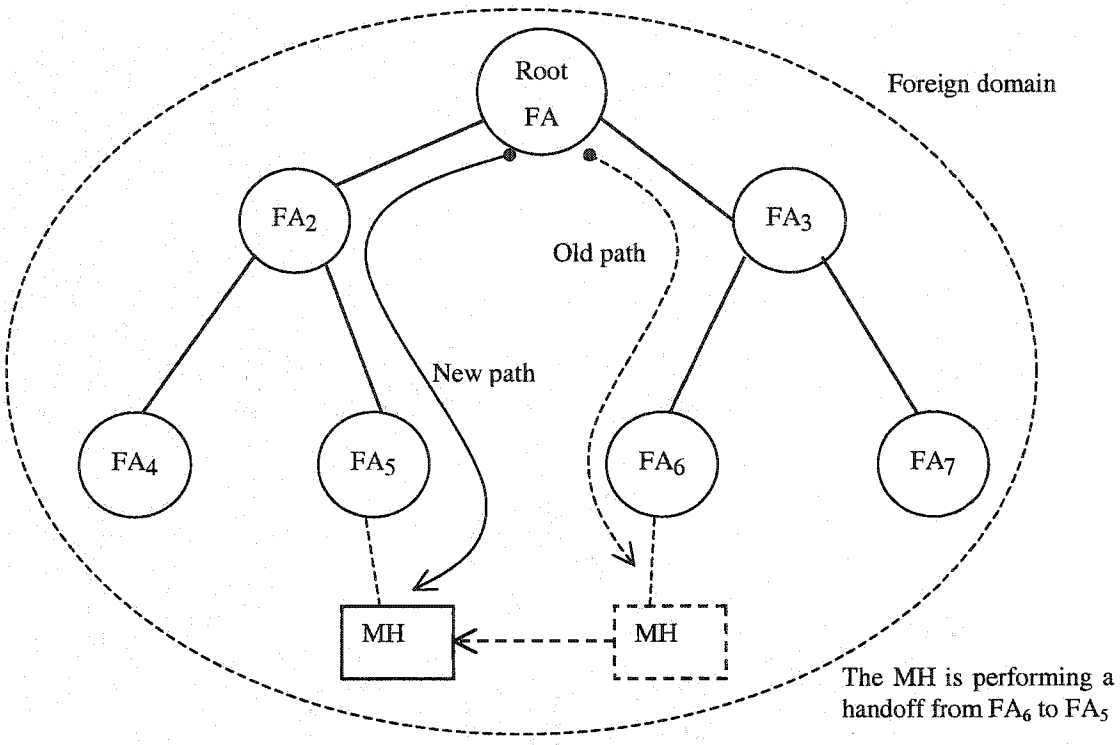


Fig. 1. A mobile host's handoff within a foreign agents hierarchy.

The objective of our work is to provide a local-area mobility support framework, within the Mobile IP framework, capable of efficiently handling local movement scenarios, alleviating the expected large protocol signaling overhead and large handoff latencies experienced with base Mobile IP. In addition, we require that the framework provide the same level of security as base Mobile IP by offering authentication and replay protection for all protocol messages. Furthermore, we do not assume the availability of the smooth handoff mechanism as an integral component of our framework.

We choose to adhere to the design guidelines of Mobile IP, in using generic mobility agents in the foreign domain, not restricting the placement or required functionality of such agents to any specific access technology. Moreover, we reuse the idea of deploying mobility agents' hierarchies in the visited domain albeit introducing our own framework for hierarchy-optimized processing of the MH's handoff for intra-hierarchy handoffs. In addition, we propose a novel cooperation-based approach between such hierarchies within the same foreign domain reducing inter-hierarchy handoff latencies as well. Our local-area mobility support framework was designed based on Mobile IPv4, although the ideas of hierarchy-optimized processing and cooperation between hierarchies are generic enough to be adapted to fit the specifics of a hierarchy-based Mobile IPv6 LMM scheme, e.g., the Mobile IPv6 regional registration approach [38].

1.3 Contributions

Our local-area mobility support framework deploys multiple cooperating foreign agent hierarchies within a foreign domain, e.g., a university campus, or a corporate site, where a novel hierarchy-cooperation feature permits reducing inter-hierarchy handoff latencies (Fig. 2). The roots of the FA hierarchies can be used as care-of addresses for the MH within such foreign domain. The framework relies on a regional processing paradigm that localizes the required registration processing when handling a MH's local handoff. In brief, the MH uses a regional registration message for intra-hierarchy handoffs, or a specially formulated home registration message for inter-hierarchy handoffs in lieu of a home registration message, to signal the fact that a local handoff is in effect. For intra-hierarchy handoffs, the regional registration message is processed by regional foreign agents within the current hierarchy without involving the HA. For inter-hierarchy handoffs, the specially formulated home registration message is processed using a cooperation approach between the new FA hierarchy and the FA hierarchy having its root as the current MH's care-of address. If deemed necessary, e.g., due to the failure of the current care-of address, the home registration is actually forwarded to the HA for regular registration processing.

Our framework attempts to alleviate the observed shortcomings with the regional registration approach for Mobile IPv4 [30], albeit reusing the nucleus idea of supporting

mobility by deploying a number of mobility agents' hierarchies in the visited domain, and adopting a regional processing paradigm, whenever possible, even for inter-hierarchy handoffs. Our aim is to take advantage of the expected network proximity between FA hierarchies belonging to the same foreign domain, in order to optimize handoff delay for inter-hierarchy handoffs.

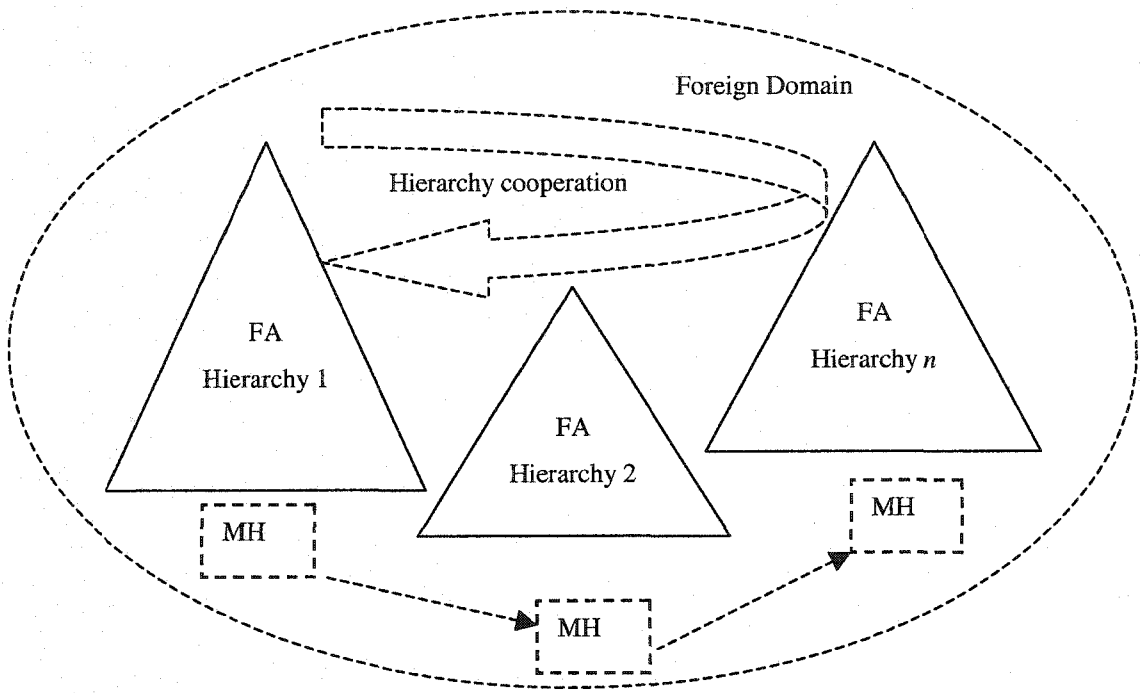


Fig. 2. A number of cooperating foreign agent hierarchies in the foreign domain.

The desired local-area mobility support framework requires addressing the following issues: foreign agent hierarchy model, registration processing for intra-hierarchy handoffs, registration processing for inter-hierarchy handoffs within the same foreign domain, and performance evaluation of the resulting local-area mobility support

framework. We briefly present our approach to resolve each the aforementioned issues highlighting our contributions.

Foreign Agent Hierarchy Model

The foreign agent hierarchy model defines hierarchy layout and mobility agents' advertisements. In section III, we present our adopted model which hides the structure of the hierarchy from visiting MHs. In addition, it provides a backward compatible mode of operation, if a legacy MH is not equipped to handle local-area mobility extensions [2].

Registration Processing for Intra-hierarchy handoffs

We identify the possible types of MH's registrations as regional, home, and home registrations involving local handoffs[2]. First, we introduce an improved regional registration processing framework, which avoids the identified race conditions within the regional registration framework, while still ensuring the consistency of tunneling state within the FA hierarchy. Moreover, we propose a new replay protection information update mechanism in order to ensure successful future registration processing by upper levels in the FA hierarchy.

Second, we identify a double purpose for home registrations involving local handoffs: a change of local care-of address, and renewal of home mobility binding. Consequently, we treat such registration as a compound home and regional registration and present two new techniques for processing home registrations involving local handoffs. The two new techniques optimize the local handoff processing by exploiting the existence of an old and new path towards the MH during its handoff. The first technique attempts to maintain the old path "alive" until a registration reply is received from the HA to set up the new path. Such task is performed by maintaining the old path active for an estimated amount of time, which is computed as a function of previous response times seen by the HA replies, and the remaining MH's registration lifetime. The second technique follows a non-traditional proactive approach in immediately

---

[2] Although a home registration, old and new paths exist on the FA hierarchy towards the MH during the handoff process.

acknowledging the handoff from the new path to the old path without waiting for the HA reply, hence emphasizing the local handoff aspect, for which the HA should not be involved. Both techniques obey our imposed requirement in providing authentication and replay protection for all exchanged protocol messages through a set of proposed protocol messages' extensions.

Section III presents the design details, analysis, and performance evaluation, through network simulation, of the suggested regional and home registration processing frameworks. Simulation experiments with one foreign agent hierarchy in the foreign domain have demonstrated the effectiveness of the proposed approaches in reducing UDP packet loss, and achieving better TCP throughput when compared to a base Mobile IP approach.

Registration Processing for Inter-hierarchy handoffs

We introduce a novel scalable and configurable cooperation-based approach between FA hierarchies to reduce inter-hierarchy handoff latencies [1], [2]. Our approach relies on configurable cooperation between the root FAs for the deployed hierarchies, in a manner that attempts to localize the registration processing for the MH's handoff. In addition, such cooperation is achieved using a minimum number of security associations [35] between deployed foreign agents in different FA hierarchies; hence scaling with a large number of such hierarchies. Section IV introduces design details, security associations' requirement analysis, and performance evaluation of our cooperation-based approach. Network simulation experiments have demonstrated the cooperation-based approach's effectiveness in reducing UDP packet loss and achieving better TCP throughput when compared to a non-cooperative approach.

Performance Evaluation

We adopted network simulation as the tool to evaluate our mobility support framework. The network simulator *ns*-2 [40] includes an implementation of Mobile IP [43]. *ns*-2 is widely used in the research community because of source availability, and possibility of components reuse and extension. The Columbia IP Micro-Mobility Software (CIMS) [15], an *ns*-2 source code extension, provides implementations for a suite of micro-mobility protocols including a simplified

abstraction of the FA hierarchy approach [30] without an actual implementation of the regional registration mechanism. CIMS implementation allows constructing a 2-level foreign agent hierarchy, and imposes a restriction that the root of the hierarchy must be the MH's HA. To the best of our knowledge, no network simulation tool provided capabilities for the construction of a true foreign domain with a number of deployed foreign agent hierarchies with an arbitrary number of levels. Section V presents a network simulation framework for local-area mobility that extends the design and functionality of CIMS to implement our proposed mechanisms for local-area mobility support. In addition, a novel foreign domain, and FA hierarchy configuration approach and implementation are introduced. Moreover, our CIMS-extension implements the regional registration approach, and the smooth handoff mechanism.

In conclusion, our contributions in this dissertation can be stated as follows.

1. A local-area mobility support framework that deploys multiple cooperating FA hierarchies within a foreign domain. The framework encompasses the following mechanisms providing authentication and replay protection for all mobility protocol messages.
    a. An improved regional registration processing technique for intra-hierarchy handoffs;
    b. Two new home registration processing techniques for home registrations involving local handoffs, emphasizing the local handoff aspect and taking advantage of the compound nature of such registrations;
    c. A novel scalable and configurable cooperation-based approach between FA hierarchies and relevant processing to efficiently handle inter-hierarchy handoffs.
2. A network simulation framework for local-area mobility implemented as *ns*-2 design and source code extensions. The simulation framework implements our local-area mobility support solution and allows modeling a foreign domain

comprised of a number of foreign agent hierarchies, each with arbitrary number of levels.

1.4 Outline

The rest of this dissertation is organized as follows. Section II introduces background information and related work for local-area mobility protocols, and host mobility protocols in general. Section III focuses on intra-hierarchy handoffs, and presents the adopted FA hierarchy model, analysis and critique of a related FA hierarchy approach, an improved regional registration processing mechanism, and two new home registration processing mechanisms when involving local handoffs. In addition, performance evaluation results of the proposed mechanisms are presented through network simulation. Section IV considers inter-hierarchy handoffs and introduces a novel cooperation-based approach between FA hierarchies to handle such handoffs, along with performance evaluation results obtained through network simulation. Furthermore, we highlight how the mechanisms in section III can be applied when moving between FA hierarchies within the same foreign domain. Section V presents a network simulation framework for local-area mobility, based on an extension of the *ns*-2 network simulator. This framework and resulting implementation have been used to evaluate the proposed techniques in sections III and IV. Section VI presents a suite of simulation experiments validating the results obtained through the simulation framework in section V, and evaluating the effects of some the network design parameters for foreign agent hierarchies, such as hierarchy height, and topology, among other factors. Finally, in section VII we conclude this dissertation summarizing our contributions as well as presenting ideas for future extensions.

SECTION II

BACKGROUND AND RELATED WORK

In section I, host mobility problem solutions were classified as network layer, application layer, or end-to-end solutions. In addition, we identified one of the major drawbacks of base Mobile IP, which is its unsuitability to handle local-area mobility due to the requirement to inform the home network upon each change of the care-of address. Consequently, base Mobile IP is assumed as a macro-mobility support framework, while a local-area mobility management protocol is required to handle intra-domain mobility.

In this section, we focus on network layer solutions to the host mobility problem, and present relevant background information including a high level overview of IP addressing and routing (section 2.1). Furthermore, we explore in more detail the operation of Mobile IPv4 as a network layer solution to the host mobility problem (section 2.2). In addition, we present modeling and operational overview of Mobile IPv4 regional registration framework as a reference hierarchy-based local-area mobility solution (section 2.3). Moreover, we classify and survey local-area mobility support solutions highlighting each solution's major merits and drawbacks (section 2.4). Finally, section 2.5 summarizes section II.

2.1 IP Addressing and Routing

The IP layer, the network layer in the Internet, provides a connectionless and unreliable datagram delivery service [50]. IP makes its best effort to deliver an IP datagram to the specified destination but there is no guarantee that the datagram actually makes it to its destination. Upper layers add any desired reliability such as the transport layer in case of TCP, or the application layer in case of UDP.

Communication in the Internet works as follows. The transport layer takes data streams and breaks them up into datagrams. Each datagram is transmitted through the Internet, possibly being fragmented into smaller units as it goes. When all the pieces finally get to the destination machine, they are reassembled by the network layer into the

original datagram. This datagram is handed to the transport layer, which inserts it into the receiving process input stream [62].

One of the most important functions of the IP layer is *routing*. In summary, for each datagram, each router in the Internet determines the next hop by finding the entry in its routing table that best matches the destination IP address. Therefore, the purpose of routing protocols within the Internet is to allow routers to exchange information about the networks they are connecting. Nodes that are not routers typically accomplish the routing objective simply by sending all of their outgoing datagrams to a default router.

An IPv4 address is a 32-bit address that has two components. The first component is a *network ID* that defines the network on which the address resides and is considered as a *routing prefix*. The second component is the *host ID* that occupies the least significant remaining bits of the IP address following the network ID bits. Routers within the Internet know only how to route datagrams based on the network ID of the destination address in each datagram; once the datagram reaches that network, it is then delivered to the correct individual host on that network.

Historically, IP addresses were divided into *five* classes: A, B, C, D, and E, with class D addresses reserved for multicast addresses. This classification implied the boundaries between the network ID and the Host ID in the IP address. With the advent of *classless addressing*, the distinction between IP addresses classes can be ignored. Instead, whenever an IPv4 network address is assigned to an organization what is assigned is a 32-bit network address and a corresponding 32-bit *netmask*. Bits of 1 in the mask cover the network address and bits of 0 in the mask cover the host. Hence, the address mask can be specified as a *prefix length* that denotes the number of contiguous bits of 1 in the mask starting from the left. IPv4 network addresses are normally written as a dotted-decimal number, followed by a slash, followed by the prefix length.

With the introduction of *IP subnetting*, the IP address hierarchy becomes a *three-level hierarchy*: a network ID (assigned to the site), a subnet ID (chosen by the site), and a host ID (chosen by the site). The boundary between the network ID and the subnet ID is fixed by the prefix length of the associated network address, whereas the boundary between the subnet ID and the host ID is chosen by the site. All the hosts on a given subnet share a

common *subnet mask*. Bits of 1 in the subnet mask cover the network ID and subnet ID, and bits of 0 cover the host ID. For instance, an assigned network address "206.62.226.0/24" implies that the leftmost 24 bits are used to identify the network ID. A subnet address "206.62.226.32/27" implies that the 27 leftmost bits are used to identify the network ID and subnet ID, where the rightmost 3 bits out of these 27 bits are used to identify the subnet ID. Hence, out of 32 bits, 5 bits are left to designate the host ID within a subnet.

The hierarchy in IP addressing and routing prevents datagrams from being routed correctly to a MH while it is away from its home network. Since a host's address logically encodes its location, without special handling for mobility, datagrams addressed to a MH will be routed by the IP layer only to the MH's home network.

2.2 Mobile IP: A Network layer Host Mobility Problem Solution

Mobile IP[3] [43] is a modification to IP that allows MHs to continue to receive datagrams no matter where they happen to be attached to the Internet. It is intended to enable hosts to move from one IP subnet to another. In general, Mobile IP specifies mechanisms to perform the following three functions: *Agent Discovery*, *Registration*, and *Tunneling*. A high level outline of the Mobile IP protocol follows [44]:

1. ***Agent Discovery***: Mobility Agents (HAs and FAs) may advertise their availability on each link for which they provide service. In contrast, a MH can send an agent solicitation on the link to learn if any Mobility Agents are present.
2. A MH uses the agent advertisements to determine whether, it is on its home network or a foreign network. When the MH detects that it is located on its home network, it operates without mobility services. When the MH detects it has moved to a foreign network, it obtains a care-of address on the foreign network. The care-of address can be a FA care-of address provided by a FA through its agent advertisement messages. Alternatively, the care-of address can be a co-located care-of address that is a local IP address on the visited subnet.

[3] Unless otherwise stated, the term "Mobile IP" in this section refers to Mobile IPv4 [43].

3. ***Registration***: When away from home, the MH registers its care-of address with its HA. The registration process can be performed either directly (co-located care-of address) or through a FA, which forwards the registration to the HA (FA care-of address). The registration process entails the exchange of a registration request and registration reply message.
4. ***Tunneling***: When the MH is away from home, the HA intercepts any datagrams sent to the MH's home address, and tunnels them to the MH's care-of address. When an FA care-of address is used, the FA is the endpoint of the tunnel and, on receiving tunneled datagrams, decapsulates them and delivers the inner datagram to the MH. When a co-located care-of address is used, the MH itself is the endpoint of the tunnel and performs decapsulation of the datagrams tunneled to it.
5. In the reverse direction, datagrams sent by the MH may be delivered to their destination using standard IP routing, without necessarily passing through the HA.

Mobile IP provides authentication for registration messages. Each MH, HA, and FA is required to support a *Mobility Security Association* (MSA) indexed by their *Security Parameters Index* (SPI) [35]. For example, each MH and corresponding HA are required to have a pre-configured MSA. When the MH sends a registration request to its HA, it computes an authenticator value, using the pre-configured MSA, and includes this authenticator value in an authentication extension to the registration request. In such manner, the HA is able to authenticate the MH's registration request.

Mobile IP provides two styles of replay protection between the HA and MH: *timestamp replay protection*, and *nonce replay protection*. The style of replay protection is part of the pre-configured MSA. The registration request contains an identification field that guarantees the freshness of registration messages. In addition, the registration reply contains a corresponding identification field that is formulated based on the identification field in the corresponding registration request. Details of either style of replay protection can be found in [43].

Base Mobile IP suffers from *triangular routing* (Fig. 3 [43]). Datagrams destined for a MH will be routed to the MH's home network, and then tunneled to the MH's current care-of address by the MH's HA, whereas datagrams originating from the MH are routed directly through normal IP routing. The *route optimization* enhancement to the base Mobile IP [47] attempts to alleviate this problem by maintaining binding caches within hosts communicating with the MH. If the MH's HA deduces that the source of a datagram destined to the MH has no binding cache entry for the MH, it should send a *binding update message* to the original source of the datagram informing it of the MH's current care-of address. The next time, this source wishes to send a datagram to the MH, it uses the care-of address, hence eliminating the triangular routing problem.

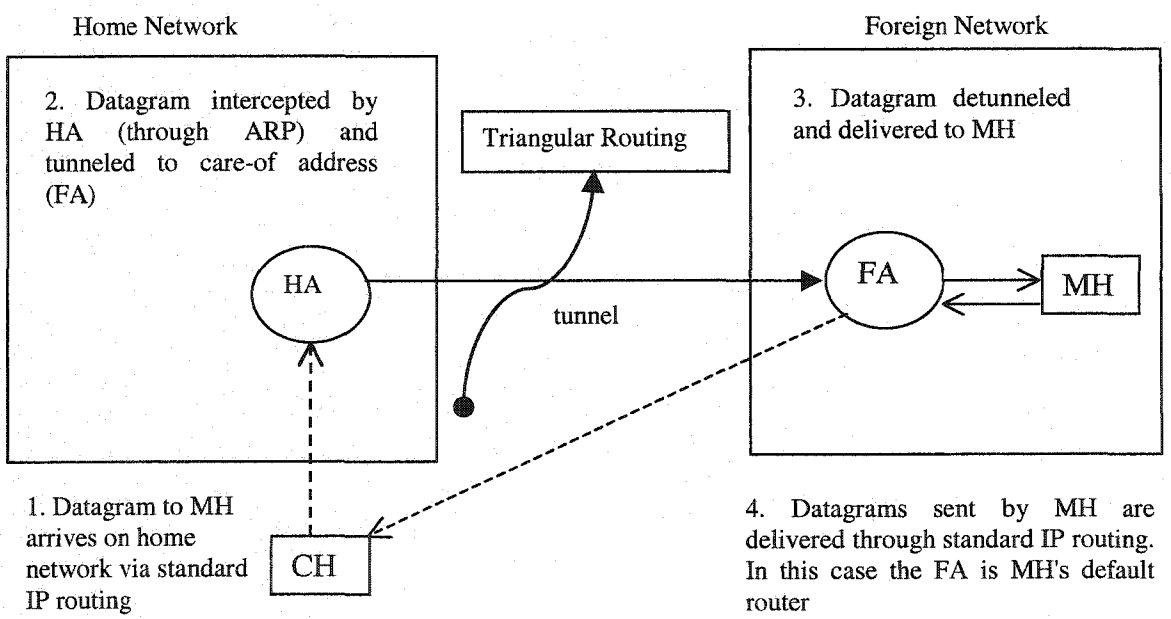


Fig. 3. Triangular routing in base Mobile IP.

The route optimization enhancement specifies mechanisms to achieve smooth handoffs between FAs. When registering with a FA, a MH may establish a registration key "session key" for the duration of its registration with its FA. When the MH later moves and registers a different care-of address, it may notify this previous FA by sending it a *binding update message* that is authenticated using the previously established registration key. Notifying the previous FA of the new care-of address for the MH allows datagrams in flight to this FA, as well as datagrams tunneled from correspondent hosts with out-of-date binding cache entries to be forwarded to the MH's new care-of address. Various methods have been proposed to establish a registration key [48]. For example, if no pre-configured MSA exists between the FA and the MH, or none can be established dynamically, the HA might act as a *Key Distribution Center* (KDC) to distribute registration keys to be used between the FA and the MH.

2.3 Overview of Mobile IPv4 Regional Registration Framework

In this section, we model and overview the operation of Mobile IPv4 regional registration framework [30], as a reference hierarchy-based micro-mobility protocol. We introduce the concept of regional registrations, and present a classification of the MH's registration messages while in the foreign domain. For simplicity, we restrict our presentation to the case of a single FA hierarchy in the foreign domain. Moreover, we explain registration signaling for different registration messages, and point out the need for a hierarchy tunneling consistency mechanism. In section III, we critique the regional registration framework, and identify race conditions within its registration processing mechanisms. This critique is presented as motivation for introducing our own framework for hierarchy-optimized registration processing for intra-hierarchy handoffs.

2.3.1 Operational Overview

A Foreign Agent hierarchy is rooted by a *Gateway Foreign Agent* (GFA), which has a publicly routable IP address. The MH can use the GFA address as its home-registered care-of address. Any intermediate level FA is termed a *Regional Foreign Agent* (RFA). A

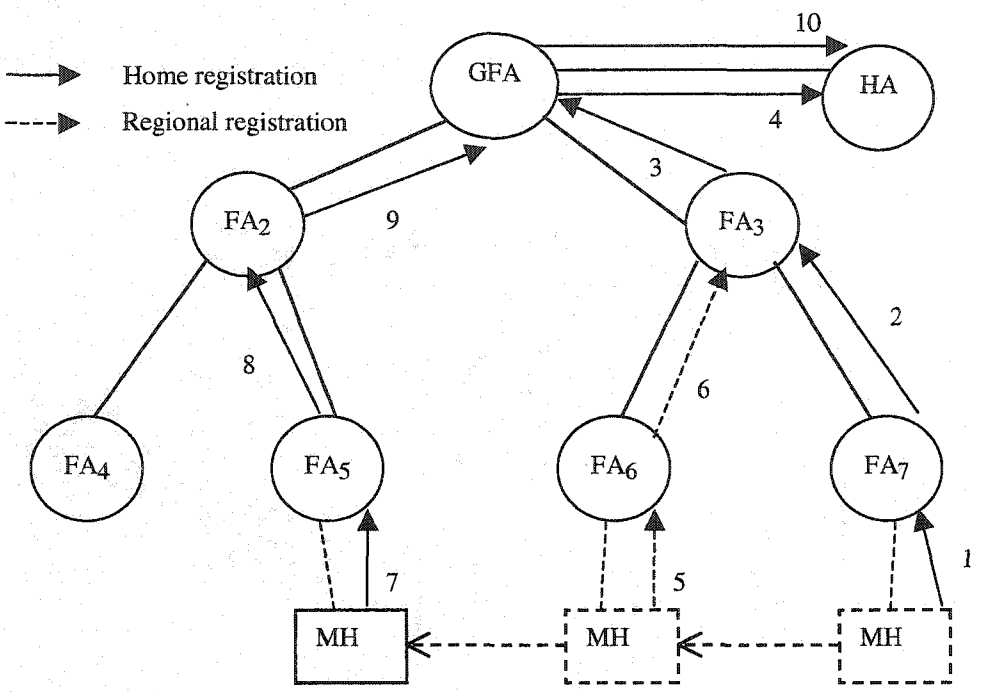
foreign agent might only advertise the GFA address, or all upper FA addresses on the path leading to the GFA, as part of its agent advertisement message.

The MH is required to perform a home registration when it first enters the foreign domain, registering the GFA as its care-of address[4]. Such care-of address does not change when the MH changes FA under the same GFA. The MH is allowed to perform regional registrations as long it is within the same FA hierarchy, i.e., did not change its GFA, changing its local care-of address within the foreign domain (as long as its registration with the HA did not expire). Thus, the MH has the responsibility of periodically renewing its home mobility-binding with its HA by periodically transmitting home registration requests.

A home registration request might be transmitted through a FA that the MH has already established as a local care-of address within the foreign domain by means of a home or regional registration mechanism. On the other hand, a home registration request, from within the current foreign agent hierarchy might coincide with a handoff from one FA to the other. We term such home registration a "Home Registration"-"Local Handoff" (HR-LH). In such case, a local handoff occurs while renewing the home mobility binding, and hence the existence of a crossover FA between the new path followed by the new home registration request to the GFA, and the old established path from the GFA to the MH's local care-of address. Fig. 4 illustrates a sample of home and regional registrations within a foreign domain comprised of one foreign agent hierarchy. Messages {1, 2, 3, and 4} represent the first home registration when the MH enters the foreign domain. The registration reply by the HA establishes {GFA-FA3-FA7} as the path to reach the MH. Messages {5 and 6} represent a regional registration (a local handoff from FA_7 to FA_6), while FA_3 is the crossover FA that generates a regional registration reply. The tunneling path within the foreign domain to reach the MH is now {GFA-FA_3-FA_6}. Messages {7, 8, 9, and 10} represent a home registration involving a local handoff from FA_6 to FA_5, with the crossover FA as the GFA. The home registration reply establishes the tunneling path {GFA-FA_2-FA_5} to reach MH.

---

[4] Alternatively, the MH may choose to home register a co-located care-of address.



{1, 2, 3, and 4}: Home registration when the MH first enters the foreign domain.
{5, 6}: Regional registration with a local handoff from FA_7 to FA_6.
{7, 8, 9, and 10}: Home registration involving a local handoff from FA_6 to FA_5.

Fig. 4. A sample of home and regional registrations within the foreign domain.

If the foreign domain deploys multiple foreign agent hierarchies with different GFAs, the MH is required to perform a home registration whenever it changes its GFA. The MH can detect such change of GFA by inspecting the foreign agent advertisement it receives, and comparing the advertised GFA and its current known GFA. In such case, the home registration involves a local handoff, but one that spans multiple hierarchies. Hence, no crossover FA exists in this case, assuming that foreign agent hierarchies are independently organized. Alternatively, a regional registration can be performed to the current GFA from within the new hierarchy with the possibility that the current FA denies such registration because the requested GFA is unknown [30]. We elaborate more

on registration processing when multiple foreign agent hierarchies are involved, when we present our cooperation-based registration processing approach in section IV.

The MH and the GFA most likely will not share a pre-established security association, and hence a style of replay protection is unknown. Thus, the MH supplies with its first home registration request a *replay protection extension*, informing the GFA of its desired style of replay protection for regional registrations and an initial value. Replay protection can be provided through the usage of an identification value in registration requests (timestamps or nonces) [43], or a challenge-response mechanism by the advertising FA [46].

When the HA receives the first home registration establishing the GFA as the MH's care-of address, it generates a *registration key* and distributes it to both the MH and the GFA as part of a registration reply message. The GFA relays the registration reply and the registration key down its hierarchy to the RFA that forwarded the home registration request. This process repeats at each intermediate RFA in a lower hierarchy level until the MH receives the registration reply and key. In such manner, the foreign agent hierarchy is capable of authenticating future regional requests from the MH. The MH uses the registration key to authenticate its regional registration requests by computing an authenticator value placed in an *authentication extension* (MH-GFA authentication extension, which is a subtype of the Generalized Authentication Extension [46]) appended at the end of the regional request. If the GFA does not distribute the registration key down the hierarchy, then only the GFA is capable of authenticating future MH's regional registration requests.

Regional registration replies are generated by the crossover FA that in some cases might be the GFA itself, e.g., in Fig. 4, FA_3 generates the regional registration reply when the MH is registering with FA_6. The crossover FA distributes the MH's registration key as part of the regional registration reply, to allow for future MH authentication by the RFAs in the new path. In general, a registration reply is propagated through the same set of foreign agents that forwarded the corresponding registration request, albeit in reverse order, establishing the MH as a registered visitor at every involved FA. If a regional FA has the MH as a registered visitor while processing a regional registration request, this

FA is the crossover FA, and hence can generate a corresponding regional registration reply. The granted lifetime in regional registration replies is the remaining MH's registration lifetime at the crossover FA. For every visitor MH, a RFA maintains a visitor entry [43] containing among other attributes, the remaining registration lifetime, the lower level foreign agent that is the tunnel endpoint for this MH, and the style of replay protection in use for this MH. When a data packet destined to the MH arrives at the GFA, it is forwarded to the tunnel endpoint stored in this MH's visitor entry. This process repeats at each intermediate RFA until the packet eventually reaches the MH.

In general, the forwarding of registration requests between foreign agents is performed as follows. If a registration request only contains the GFA address as care-of address, a leaf FA appends its own address to the registration request by placing it in a hierarchical FA extension. Such data extension is protected by using an FA-FA authentication extension. The purpose of such data and authentication extensions is to inform the upper RFA about the address of the forwarding FA in an authenticated manner. The upper RFA (receiving RFA) creates a pending registration entry with a care-of address the forwarding FA address. Before forwarding to a next-level RFA, the current RFA removes such data and authentication extensions, appending respective extensions that provide his own address. Such process repeats until the registration request reaches the GFA or the crossover FA, in the case of a home registration or a regional registration, respectively.

2.3.2 Tunneling Consistency Problem and Solution

The ability of a FA to correctly identify itself as the crossover FA for a regional registration request is crucial to the correct registration processing by foreign agent hierarchies. Such ability might be hindered due to the following specification of the base Mobile IP protocol [43]: a MH is not required to inform a FA that it is no longer registered with it, i.e. that it is currently registering with a new FA, relying on an eventual expiration of registration lifetime. Such approach reduces protocol messages overhead, but creates a *hierarchy tunneling consistency problem* for FA hierarchies: *a RFA not informed that the MH is no longer a current visitor might erroneously decide that it is the*

crossover FA and generate a regional registration reply in response to a regional registration request, although such request should be forwarded to upper level RFAs. Such consistency problem (e.g., see Fig. 5) occurs when the MH is attempting to register with an old FA for which the registration lifetime has not yet expired, and that was not initially informed that the MH is no longer a current visitor. In Fig. 5, the MH was originally registered with FA5, switched to FA6 and then back to FA5. FA5 and FA2 were not informed that the MH is no longer a current visitor when it switched to FA_6. Consequently, FA_2 erroneously generates a regional registration reply for the MH's regional registration request, whereas the request should have been forwarded to the GFA to adjust the GFA's tunnel endpoint for the MH to point to FA_2. Hence, a future data packet received at the GFA, is tunneled to FA_3, whereas it should have been tunneled to FA_2.

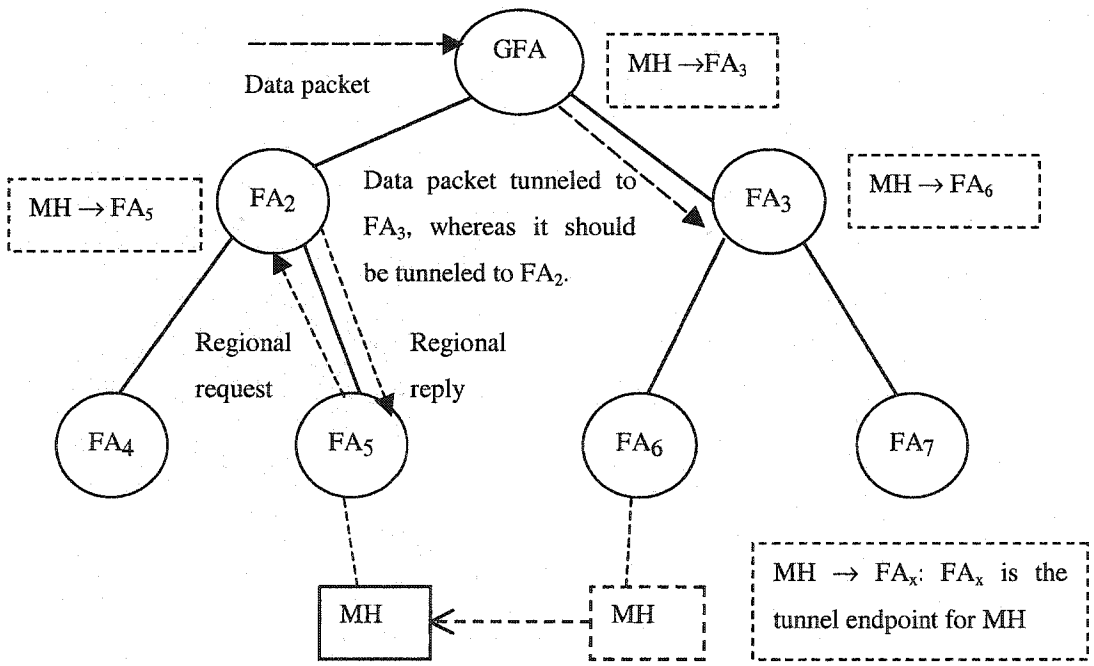


Fig. 5. Foreign agent hierarchy tunneling consistency problem.

Hence, a mechanism is required by which old regional foreign agents are informed that a MH is no longer a current visitor. The regional registration framework requires a smooth handoff mechanism [47] to be performed by the MH and a new FA in order to inform an old FA that the MH is no longer a current visitor. Simply stated, the smooth handoff mechanism requires that the MH supply a *Previous Foreign Agent Notification Extension* (PFANE) along with its registration request to the new FA specifying the new FA as its new care-of address, and a lifetime within which this binding is valid. The new FA uses the information in the PFANE to send a *Binding Update* (BU) message on behalf of the MH to the old FA, informing it of the new whereabouts of the MH. The old FA is required to send back a *Binding Acknowledge* message to the new FA that delivers it to the MH. The old FA is capable of authenticating the BU since it shares a registration key with the MH as a result of the first home registration performed by the MH (section 2.3.1). After authenticating the BU, the old FA deletes any MH's visitor entry, and creates a new binding cache entry for the MH to forward any newly arriving data packets destined to the MH to the new FA.

The regional registration framework further extends this smooth handoff process to ensure tunneling consistency within the hierarchy as follows. The old FA relays the BU message upwards in the hierarchy (to its father FA) specifying itself as the care-of address of the MH, and using the lifetime supplied by the MH in the original BU message. The father FA performs the following steps in response to receiving the BU message.

- Delete its MH's visitor entry,
- Create a binding cache entry for the MH with care-of address the child FA that sent the BU message,
- Relay the BU message upwards in the hierarchy,
- Send back a binding acknowledge message to its child FA.

Such process at each intermediate RFA repeats until the BU message reaches the crossover FA, which at this point generates a binding acknowledge message to the MH and sends it down the old path to the MH. The crossover FA deduces it is the crossover FA, and hence generates the binding acknowledge message to the MH, because it has

received an earlier registration request from the MH's new path beneath it in the hierarchy. The same process is used for regional registrations and HR-LH requests. Fig. 6 provides an example of the tunneling consistency mechanism in the case of a regional registration. The MH was served by FA_5, switched to FA_6, then back to FA_5. Hence, FA_5 is currently the new FA, while FA_6 is the old FA. We can observe that BU messages flow in the old path towards the crossover FA at the same time that registration requests flow in the new path towards the crossover FA. Moreover, the crossover FA identifies its "crossover" status based on receiving an earlier registration request. We elaborate more on the consequences of such signaling design, and identify potential race conditions, when we present a critique of the involved registration processing mechanisms in section III. Note that binding acknowledgement messages are omitted from Fig. 6 for figure simplicity.

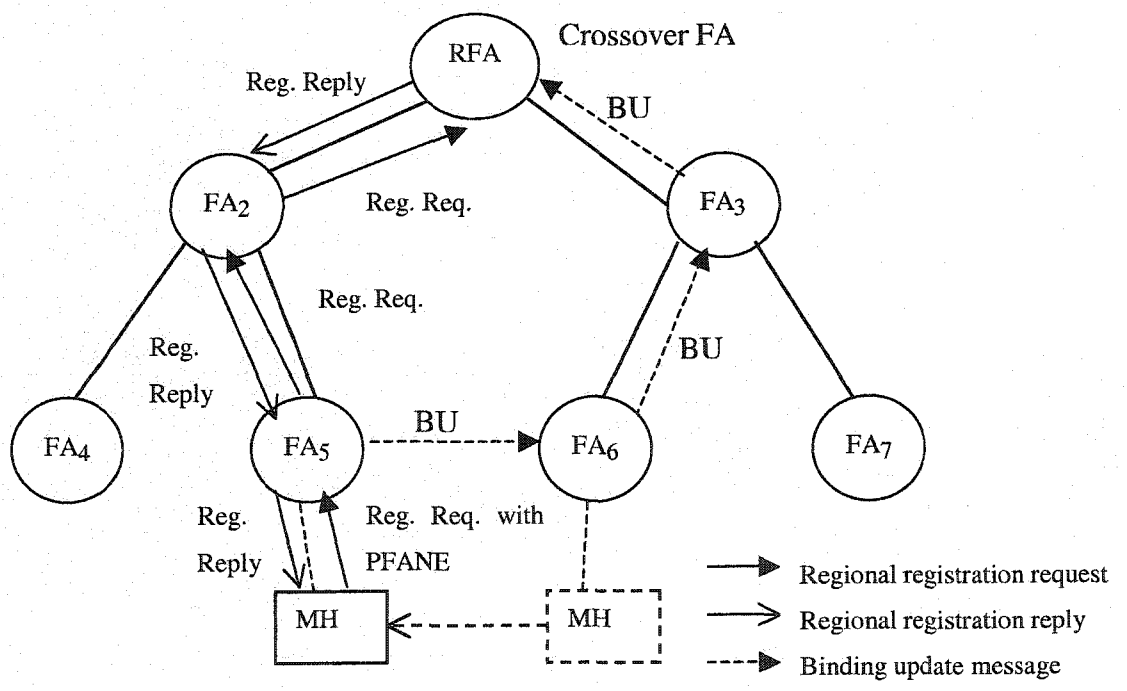


Fig. 6. Tunneling consistency mechanism applied to regional registration.

2.4 Local-area Mobility Protocols: A Taxonomy and Survey

Local–area mobility protocols aim at reducing home signaling overhead, and improving handoff latency to reduce potential packet loss. Such objectives are achieved by localizing the effect of the MH handoff such that handoff processing is confined to the local domain at hand. Several local-area mobility protocols have been suggested in the literature. Researchers attempted to compare and contrast such protocols qualitatively, and through network simulations [11], [12], [57]. The existing protocols can be classified as belonging to the following categories.

- *Extensions to Mobile IP.* Such solutions attempt to adhere to the Mobile IP framework, while localizing the MH's registration processing. One well-known idea is to extend the notion of a mobility agent into a hierarchy of mobility agents. Such hierarchy can be viewed as a mobility support overlay network within the foreign domain, using hierarchical tunneling to forward packets to the MH. Our proposed local-area mobility support framework belongs to this category.
- *Host-based forwarding schemes.* Such solutions install host-based routing entries within the foreign domain, requiring path set-up mechanisms in order to update such entries. Such approaches attempt to avoid the decapsulation and re-encapsulation overhead associated with hierarchical tunneling.
- *Multicast-based schemes.* Such solutions exploit the usage of multicast technology by assigning a multicast address to the MH within the foreign domain. Handoffs are handled through standard multicast join and prune messages.

Recently, a new approach for localized mobility management [34] has been proposed which advocates exploiting existing optimized handoff mechanisms, e.g., the forwarding mechanism from the previous foreign agent in the route optimization extension for Mobile IPv4 [47], instead of introducing additional mobility agents in support of local-area mobility. Such solutions do not reduce the home signaling overhead, but attempt to

improve handoff latency to reduce possible packet loss. The following sections present representative protocols for each of the aforementioned categories.

2.4.1 Mobile IP Extensions

Protocols in this category can all be abstracted as adopting a mobility agents' hierarchy architecture in the visited domain. Such protocols include the Mobile IPv4 regional registration approach [30], and the local and indirect registration approach [21]. Related solutions, not considered pure mobile IP extensions, adopting a mobility agents' hierarchy approach, include TeleMIP [18], and the fast and scalable handoffs approach by Careres et al. [9]. For completeness, we mention Mobile IPv6 targeted solutions, which include Hierarchical Mobile IPv6 [14], and Mobile IPv6 regional registration framework [38].

Mobile IPv4 regional registration approach

In section 2.3, we modeled and overviewed the regional registration approach [30]. Such approach is sometimes referred to as hierarchical Mobile IP, because of its reliance on one or more hierarchy of mobility agents in the foreign domain. In summary, the MH uses a regional registration message when performing intra-hierarchy handoffs, while a crossover FA, the first common FA between the old and new path, is responsible for handling and replying to the MH's registration request.

A number of FA hierarchies might be deployed in the same domain. When changing GFA, i.e. moving between FA hierarchies, the MH is required to register with its HA. Nevertheless, existing prototype implementations, and simulations have considered deploying only one FA hierarchy in the foreign domain [27], [49]. The FA hierarchy approach is sensitive to FA failures. In addition, the GFA is required to maintain a visitor entry for every MH currently registered within its hierarchy. Nevertheless, it is independent of any physical network placement of FAs and offers the same level of security as the base Mobile IP by extending the Mobile IP registration and authentication process.

A relevant issue is making the FA hierarchy more fault-tolerant. Omar et al. [42] attempt to achieve this target by suggesting two approaches to recover from a regional FA failure. The first approach reverts to a non-hierarchical FA setup. Affected MHs are informed to perform a registration with the corresponding HAs, changing their registered care-of address, the GFA IP address, to their current FA. The second approach heals the broken hierarchy, by making an upper level FA remove the faulty FA from the hierarchy, and point to a lower FA in the hierarchy that has less probability of failure. The two suggested approaches do not deal with GFA failures.

El Malki and Soliman [24] introduce fast handoffs in a "flat" network setting through a "bicasting" approach to support data forwarding to the previous and new foreign agents. The "bicasting" approach is enabled through simultaneous bindings, which the MH explicitly requests in its registration request message. The authors explain how to apply the fast handoff approach within a hierarchical Mobile IP setting, and suggest routing improvements for data traffic between MHs within the same domain.

In a related approach, Avancha et al. [5] suggest the use of forwarding pointers between domain foreign agents (DFA) to implement a fast handoff scheme. A three-level mobility support hierarchy is assumed. The hierarchy is comprised of a DFA at the root (similar to the GFA), subnet FAs at the second level to represent subnets, and FAs at the leaf level to act as BSs. The MH is responsible for registering with the previous DFA, when handing off to a new DFA, without informing the HA. Hence, a forwarding chain is maintained amongst DFAs to point to the current domain where the MH is located. When the chain of forwarding pointers reaches a certain limit, the current DFA is responsible for collapsing the chain by sending a registration request to the HA. In such approach, the length of the forwarding chain increases the observed packet latency, hence affecting delay-sensitive applications. In addition, security implications of the proposed approach are not discussed.

Local and indirect registration (Anchor FA approach)

This approach [21] aims at reducing handoff latencies within the visited domain. Handoff latencies are attributed to three components: latency in Mobile IP registration,

delay incurred to set up FA-HA dynamic keys, and latency incurred in setting up FA-HA secure tunnels. Two approaches are suggested: *local registration*, and *global indirect registration*. Either approach requires the MH to perform a *global registration* with its HA upon entering the visited domain. In the *local registration* approach, it is assumed that the current FA and the MH establish a shared security association. Later on, the current FA acts as an *Anchor FA* for this MH, authenticating the MH while it moves within the same domain. When the MH changes FA within the same domain, the new FA performs local registration with the Anchor FA. This approach requires a shared security association between every two FAs in the domain. In this manner, the handoff latency is reduced to registering locally with the Anchor FA. The *global indirect registration* approach is used when no security association could be established between the current FA and the MH, requiring the HA to always authenticate the MH registration. Again, the current FA acts as an Anchor FA for this MH. Any new FA directs the MH registration towards the Anchor FA, the Anchor FA relays the registration to the HA, which authenticates the registration. This approach eliminates the latency due to FA-HA dynamic key establishment, and the latency due to FA-HA secure tunnel establishment. This approach can be abstracted as dynamically creating a two-level FA hierarchy rooted by the anchor FA, while all other FAs become leaves in such hierarchy. The anchor FA can change from one MH to the other. However, security association requirements for such approach represent a scalability issue with increased number of deployed FAs. Every two MHs need to have a pair of security associations, one in each direction, in order to authenticate any inter-FA messaging.

Fast and scalable handoffs

This approach [9] suggests also the use of FA hierarchies. However, distinguishes the *local mobility* case where a MH moves between BSs on the same IP subnet, and does not rely on standard Mobile IP signaling. The *Address Resolution Protocol (ARP)* proxy and gratuitous ARP messages are used in the IP subnet to maintain the illusion that the MH resides on the wired link in this subnet. The handoff protocol between the old and the new BS uses a retransmission buffer. The size of the retransmission buffer size is tuned to

the number of expected packet losses during a handoff, and the complete buffer is retransmitted from the old to the new BS after every handoff to reduce packet loss. Movement between IP subnets is handled by *subnet FAs*. A MH uses a *domain FA* IP address as its care-of address in its Mobile IP home registration. Use of proxy and gratuitous ARP represents a major security problem in this approach.

Telecommunications-enhanced Mobile IP (TeleMIP)

TeleMIP [18] is an IP-based architecture to handle intra-domain mobility in cellular wireless networks. A two-level mobility agents' hierarchy is proposed in the foreign network. At the root level, a Mobility Agent (MA) provides a global care-of address for the MH, while at leaf levels, subnet Agents (SA) provide local care-of addresses to the MH. Conceptually, TeleMIP is a special case of the general FA hierarchy approach, specialized for wireless access networks, although standard Mobile IP registration signaling is not used. The base intra-domain mobility management protocol (IDMP) in TeleMIP is further extended to support fast handoffs and paging [17].

2.4.2 Host-based forwarding schemes

Protocols in this category install host-specific routing entries in the foreign domain to alleviate the hierarchical tunneling overhead observed with the mobility agents' hierarchy approach. Special techniques are required to maintain and update such entries. Examples of such update techniques include snooping on the MH's data packets to update the current whereabouts of the MH, and explicit message signaling by the MH. Protocols include Cellular IP [64], and the Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) [55]. A generic host-based routing scheme was introduced in [69] while comparing the performance of the aforementioned schemes with the hierarchical Mobile IP approach. In addition, Campbell et al. [12] recently compared the performance of hierarchical Mobile IP, cellular IP, and HAWAII through network simulation by using the Columbia IP Micro-Mobility Software (CIMS) [15].

Cellular IP

This approach [13], [64] suggests handling local-area mobility through a wireless access network. A wireless access network primarily consists of base stations interconnected by wired links, and other nodes that have no radio device, but can have mobility support functions. The network is connected to the Internet through routers, called *gateway routers* (GW). The gateway router can act as a HA or a FA.

The wireless access network is partitioned into *Paging Areas*. BSs transmit which paging area they belong to as part of their periodic beacon signals. Packets addressed to a MH are routed to its current BS on a hop-by-hop basis. To accomplish that, two types of caches are deployed within the wireless access network: *Paging Caches* maintained for idle mobile hosts, and *Routing Caches* maintained for MHs currently receiving or expecting to receive data. Paging caches are deployed in selected nodes of the wireless access network, while it is expected that most of the nodes will have a routing cache.

While idle, the MH is required to send a *paging-update packet* whenever it enters a new paging area. If the MH roams within the same paging area it periodically sends the paging-update packet only when a specific timer expires. These packets are routed hop-by-hop towards the GW where they are eventually discarded. Meanwhile, they update any paging caches along the way. When data packets are available to the MH, the GW sends a *paging packet* that is routed to the MH. At any node, if no up-to-date route cache mapping is available, the paging cache is used to route the packet. If a node does not contain a paging cache, it forwards the paging packet over all its downlinks. If a node contains a paging cache with no mappings for this MH, the packet is discarded. When the MH receives this paging packet, it responds with a *route-update packet* that configures routing caches along the way to the GW. Hence, the nodes in the network do not exactly know the location of the MH, until data packets are available. In such case, a paging process is performed and routing caches are configured through a control packet sent by the MH. Any data packets transmitted by the MH are routed to the GW on a hop-by-hop basis. Nodes that contain routing caches monitor these passing data packets and use them to update the routing mappings. If the MH is not transmitting, only receiving or expecting

to receive, it is required to periodically send route-update packets to keep the Routing caches current.

In such approach, the GW presents a single point of failure. In addition, when the number of MHs increases, the number of control packets needed to keep the mappings current increases possibly overloading the wireless access network.

HAWAII

This approach [54], [55] suggests partitioning the wireless access network into administrative domains with domain gateway routers named the *Domain Root Routers*. When a MH is moving within its home domain, it retains its IP address. Packets destined to the MH reach the Home Domain Root Router based on the subnet address of the domain and are forwarded over special dynamically established paths to the MH. In such manner, The HA functionality is not needed while the MH is moving within its home domain.

When a MH is visiting a foreign domain it is required to obtain a co-located care-of address within the foreign domain. The MH keeps this care-of address as long it is within the same foreign domain. Nevertheless, it is required to register with a BS within the domain to better handle handoffs. The BS in turn informs the MH's HA about the MH's co-located care-of address through the Mobile IP registartion process. The HA forwards any datagrams for the MH to its care-of address. These datagrams reach the foreign domain root router through normal IP routing, and are forwarded over dynamically established paths until they reach the MH. When the MH powers up, it sends a Mobile IP registration message to its nearest BS. The BS propagates a HAWAII *path setup update* message to the Domain Root Router using a configured default route. Each router in the path between the MH and the Domain Root Router adds a forwarding entry for the MH. Finally, the domain root router sends back an ACK to the BS, which sends a Mobile IP registration reply to the MH. The host-based forwarding entries are soft-state entries that are kept alive by periodic hop-by-hop HAWAII *refresh* messages.

Two variants of path setup schemes are proposed motivated by two types of wireless networks. The *Forwarding scheme* is optimized for networks where the MH is able to

listen/transmit to only one BS. The *Non-Forwarding* scheme is optimized for networks where the MH is able to listen/transmit to two or more BSs simultaneously for a short duration. In the Forwarding scheme, packets are first forwarded from the old BS to the new BS before they are diverted at the crossover router. Whereas, in the Non-Forwarding scheme, as the path setup message travels from the new BS to the old BS, data packets are diverted at the crossover router to the new BS, resulting in no forwarding of packets from the old BS.

The problem we envision with such approach is the requirement that the MH must acquire a new co-located care-of address whenever it changes domains. This requirement stresses the already depleted IPv4 address space. In addition, all the routers in the domain must maintain host-based entries to efficiently implement the path setup scheme. Nevertheless, the proposed approach takes into account the different types of wireless networks suggesting two corresponding path setup schemes. In addition, HAWAII has been extended to include paging functionality [53].

2.4.3 Multicast-based schemes

A number of proposals have investigated the use of multicast technology as a solution for the host mobility problem. The proposals range from using multicast as the sole mechanism to provide addressing and routing services to MHs [41], to the HA pre-assigning the MH a multicast address as suggested in the Daedalus approach [6]. The previous approaches are not considered as local-area mobility support solutions as they require handling issues such as allocating unique multicast addresses across the wide area network. Multicast-based local-area mobility solutions exploit the usage of multicast within a confined domain as an efficient mechanism to achieve fast handoffs through standard multicast join and prune mechanisms [28], [32], [61].

To achieve fast handoffs in connection-oriented picocellular (in-building) networks, Ghai and Singh [28] propose a three level hierarchy. At the lowest level is a dense collection of MHs that communicate with each other and with stationary hosts in the network as they move between cells. *Mobile Support Stations* (MSSs) handle communication in each cell. The MSSs are all connected to an assigned supervisor

machine called the *Supervisor Host* (SH). The SH has the responsibility of tracking MHs and maintaining their connections within its subnet. Picocells in the vicinity of the MH are dynamically formed into multicast groups and packets for a MH are multicast by the SH to all MSSs within the group. The shape and composition of the group is determined by the architecture of the building, the speed at which the MH moves between cells and the direction of motion. The group is updated each time the MH moves between cells and all the MSSs that belong to the new group are informed by the SH of the group identity and the identity of the corresponding MH.

Tan et al. [61] proposed assigning a multicast address to the MH within the boundaries of the foreign domain. They describe a foreign domain architecture with a two-level hierarchy, with a *Domain FA* (DFA) at the top of the hierarchy, and BSs as leafs of the hierarchy tree. The DFA performs multicast address allocation to the MH. All MHs are required to register with the FA according to Mobile IP specifications. Multicast is used as the packet forwarding mechanism from the DFA to the BSs. To ensure no multicast address conflict, the authors point out that mechanisms for allocating multicast addresses globally, such as allocating a range of multicast addresses to each domain, must be used. However, The details of the multicast address allocation mechanism are not discussed. In addition, The DFA approach is a centralized approach for multicast address allocation that does not scale well with large numbers of MHs. Moreover, the DFA is required to be the forwarding agent for all MHs within the foreign domain in which case the DFA becomes a bottleneck and a single point of failure.

Helmy and Jaseemuddin [32] suggest allocating a locally-scoped multicast address to the MH within the foreign domain. Data packets are multicast-tunneled to the MH's allocated multicast address within the foreign domain. The authors suggest two architectures for multicast-based mobility support, where a unicast regional care-of address (RCOA), and a multicast address (MCOA) are assigned to the MH. The first architecture is a proxy-based architecture where a mobility proxy (MP) handles the inter-domain handoff and allocates MCOA. The second architecture is based on algorithmic mapping where an Access Router (AR) infers MCOA from RCOA through algorithmic mapping, while the MH informs the HA about its RCOA in its registration request. Such

approach alleviates the need for explicit multicast address allocation. In addition, the authors investigate several multicast state aggregation techniques at routers, since the required number of multicast groups is proportional to the total number of MHs currently in the foreign domain.

Adopting a multicast-based solution is an attractive approach, because of similarities between network layer mobility management and multicast group management issues. However, several multicast-related issues have to be addressed such as efficient multicast address allocation schemes, multicast routing techniques that scale to a large number of multicast groups with a limited number of participants, and efficient techniques for aggregation of multicast groups state within domain routers.

2.5 Summary

In section II, we presented relevant background information about network layer mobility solutions. In addition, we overviewed the operation of Mobile IPv4 as a network layer host mobility problem solution. Furthermore, we modeled and overviewed the operation of Mobile IPv4 regional registration approach as a reference hierarchy-based local-area mobility protocol. We identified the main resulting registration classes, and pointed out the need for a hierarchy tunneling consistency mechanism. Furthermore, we classified and surveyed local-area mobility protocols in general, identifying their main categories as: Mobile IP extensions that leverage the existence of a hierarchy of mobility agents in the foreign domain, host-based forwarding, and multicast-based solutions.

In section III, we present a hierarchy-based local-area mobility support framework for intra-hierarchy handoffs, motivated by identified race conditions and drawbacks within Mobile IPv4 regional registration framework.

SECTION III

A REGISTRATION FRAMEWORK FOR INTRA-HIERARCHY HANDOFFS

In this section, we introduce our framework for Mobile IP registration processing within a foreign domain where local-area mobility is supported through the deployment of foreign agent hierarchies. We present a FA hierarchy model, which hides the structure of the FA hierarchy structure from visiting MHs, while offering a backward-compatible mode of operation suited for legacy MHs, which are unequipped to handle local-area mobility extensions. We focus on intra-hierarchy handoffs and present novel registration processing techniques for such handoffs when associated with either home or regional registrations. The adopted message signaling attempts to overcome shortcomings and potential race conditions identified in the regional registration framework for Mobile IPv4 (MIP_RR) [30], which is a related hierarchy-based local-area mobility solution (see section 2.3). Moreover, an attempt is made to exploit the presence of a mobility support overlay network in the form of a FA hierarchy, and emphasize the local handoff aspect in order to optimize the mobile host registration and regional handoff processing. In section IV, the proposed mechanisms herewith are used as a building block within a local-area mobility solution based on cooperative foreign agents hierarchies to handle inter-hierarchy handoffs. We qualitatively and analytically compare and contrast our proposed mechanisms versus MIP_RR. In addition, we simulated our proposed framework using design and implementation extensions (section V) of the network simulator *ns*-2 [8] and experimented with UDP and TCP traffic. Network simulation results demonstrate that the proposed techniques are effective in reducing UDP packet loss, and achieving better TCP throughput, compared to a base Mobile IP approach, in the case of a distant HA.

Section III is organized as follows. Section 3.1 presents a critique of the regional registration framework for Mobile IPv4, identifying some drawbacks and potential race conditions. Section 3.2 introduces our adopted foreign agent hierarchy model. Section 3.3 presents a regional registration processing approach, which prevents the identified race

conditions and drawbacks in section 3.1. In addition, a new replay protection update mechanism is suggested to propagate to upper levels in the hierarchy, any new identification values assigned to the MH. Section 3.4 presents two novel approaches for processing home registrations associated with local handoffs. Section 3.5 presents performance evaluation results using network simulations comparing the proposed approaches versus other approaches, e.g., base Mobile IP. Finally, section III is concluded in section 3.6.

3.1 Motivation: Critique of The Regional Registration Framework

In section II, we modeled and overviewed MIP_RR [30], introducing the concept of regional registrations, and pointing out the need for a hierarchy tunneling consistency mechanism as an integral part of a registration protocol signaling. Such mechanism is crucial in ensuring consistency of tunneling state within the mobility support overlay network (FA hierarchy), when the MH changes serving foreign agents in an intra-hierarchy handoff (see section 2.3.2). In this section, we identify some drawbacks and potential race conditions within the aforementioned tunneling consistency mechanism. Such drawbacks and race conditions motivated us to introduce our framework for MH's registration processing presented later in section 3.3, and section 3.4.

Evaluating MIP_RR's tunneling consistency mechanism, we point out the following advantages.

- It requires the smooth handoff mechanism [47], hence reduces potential packet loss until handoff completion;
- It introduces the binding acknowledgment of binding update (BU) messages between RFAs to insure message delivery;
- The tunneling consistency mechanism is symmetrical for home and regional registrations.

However, the following disadvantages can be observed.

- A MH that is not smooth handoff enabled would be denied service from within such FA hierarchy.

- A potential race condition exists within the proposed message signaling, which might lead to inconsistent tunneling state within the hierarchy.

The identified race condition can be described as follows. If the BU message propagated through the old path reaches the crossover FA before the registration request propagated through the new path, a crossover FA can not deduce it needs to generate the MH's binding acknowledge message and hence forwards the BU message to its father FA towards the GFA. Such scenario has the following consequences.

- Upper RFAs, higher than the crossover FA, possibly up to the GFA will replace the MH's visitor entries with corresponding binding caches. The MH's will be considered as not currently a visitor for these RFAs.
- The MH's remaining registration lifetime, which the initial lifetime amount was initially granted by the HA, will be replaced with the specified lifetime in the BU message. Tunneling lifetime inconsistencies will exist in the hierarchy.

In the case of a home registration associated with a local handoff (HR-LH), such BU lifetime might expire before a registration reply is received from the HA. In the case of a regional registration, such BU lifetime is not consistent with the lifetime granted by the crossover FA in the generated regional registration reply, thus creating remaining lifetime inconsistencies within the RFAs. MIP_RR's authors suggest that upper level RFAs should ignore the BU forwarded by the crossover FA since it does not supply any new care-of address. We argue that this condition is true for every binding update propagated through the old path and can not be used as a special condition to ignore the BU for RFAs above the crossover FA.

The signaling design flaw highlighted here stems from allowing 2 messages, the BU through the old path, and the registration request through the new path, to simultaneously flow in 2 separate paths towards the crossover FA that identifies its "crossover" status when receiving a registration request through the existence of a visitor entry for this MH. Furthermore, such separate paths may not be symmetric in terms of available bandwidth, link delay, or current congestion status. In Fig. 7, we abstract the tunneling consistency mechanism along with the involved hierarchy links. The MH is handing off from the old FA to the new FA and generating a registration request (home or regional). When the new FA receives the registration request; it initiates the tunneling consistency mechanism

by sending a BU to the old FA, and propagates the registration request upwards in the FA hierarchy. In order to better quantify the conditions favoring the occurrence of the identified race condition, we introduce a set of relevant delay measures as summarized in TABLE 1.

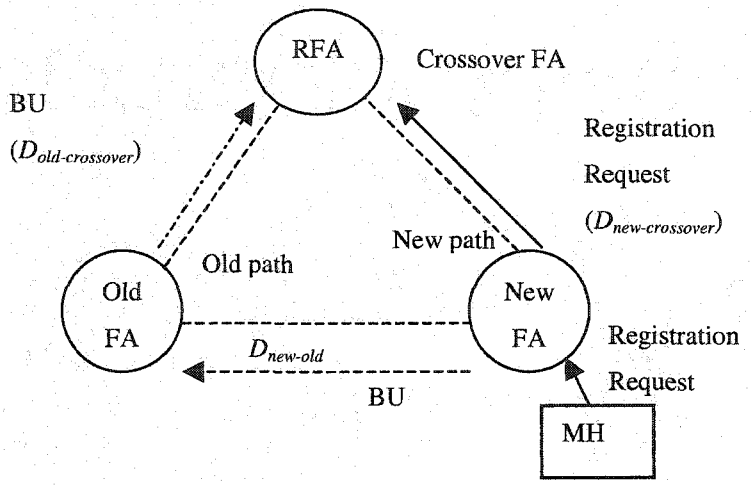


Fig. 7. Abstraction of the tunneling consistency mechanism.

TABLE 1

SUMMARY OF DELAY MEASURES

Delay Measure	Description
$D_{new\text{-}old}$	The delay required for a BU message to be generated and transmitted by the new FA to reach the old FA, when the new FA receives the MH's request.
$D_{old\text{-}crossover}$	The delay for a BU message to be generated and transmitted by the old FA and relayed by each intermediate RFA in the old path to reach the crossover FA, when the old FA receives the BU from the new FA.
$D_{new\text{-}crossover}$	The delay for a MH's registration request to be relayed by the new FA and propagated by each intermediate RFA in the new path to reach the crossover FA, when the new FA receives the MH's request.

Referring to Fig. 7 and TABLE 1, we highlight the factors affecting each of the introduced delay measures. $D_{new\text{-}old}$ is dependent on link, queuing, and routing delays on the routing path from the new FA to the old FA. $D_{old\text{-}crossover}$ ($D_{new\text{-}crossover}$) is dependent on link, and queuing delays on each intermediate link between the old (new) FA and the crossover FA along the old (new) path in the FA hierarchy, and on processing delays at each intermediate RFA on this path. The identified race condition with possibility of subsequent hierarchy tunneling inconsistencies will occur if the inequality in (1) holds.

$$D_{new-old} + D_{old-crossover} \leq D_{new-crossover} \quad (1)$$

3.2 Foreign Agent Hierarchy Model

In this section, we present our adopted foreign agent hierarchy model, in terms of terminology and mobility agents advertised addresses. We reuse the terminology from MIP_RR [30], and assume that a FA hierarchy (Fig. 8) is rooted by a *Gateway Foreign Agent* (GFA), which the MH can use its publicly routable IP address as its home-registered care-of address. Any intermediate level FA is termed a *Regional Foreign Agent* (RFA).

We place no restriction on the number of levels in the hierarchy, nor on the number of FAs within a hierarchy level. When the MH is handing off from one FA to another, the *crossover FA* is the first common FA between the old and the new path to the GFA. In Fig. 8, when the MH hands off from FA_7 to FA_6, the old path is {FA_7, FA_3, GFA} and the new path is {FA_6, FA_3, GFA} with the crossover FA being FA_3. An individual FA advertises the IP address of the GFA, and might advertise his own IP address if such address is not private. If the MH is not equipped to handle regional registrations, and an FA advertises its own IP address, the MH can register this FA's address as its home-registered care-of address, according to the base Mobile IP protocol [43]. Hence, we offer a backward compatible mode of operation for legacy MHs, which are unable to handle local-area mobility extensions.

Individual FAs advertise their *Network Address Identifier* (NAI) [3] through a FA-NAI extension [10] appended to the agent advertisement. Inspection of the FA's NAI

allows a MH to decide whether it has changed foreign domains, or whether it has returned home. Furthermore, if individual FA addresses are private addresses, the MH uses the previous foreign agent's NAI to identify such FA when sending a registration request to a new foreign agent as part of a smooth handoff mechanism [47]. For instance, in Fig. 8 FA_7 advertises the IP addresses {FA_7, GFA}, as well as his NAI {FA_7@RealmX}. For authentication purposes between foreign agents, security associations [35] are required between each parent FA and its children FAs beneath it in the FA hierarchy.

We believe that an FA should not advertise the FA hierarchy addresses leading to its GFA. In such manner, the size of the FA advertisement is not dependent on the number of hierarchy levels above it in the FA hierarchy, less bandwidth is required if the FA advertisement is to be transmitted over a wireless link, the structure of the FA hierarchy is hidden from the MH, and the structure of the FA hierarchy can change dynamically without having to alter the FA advertisement. The fact that the FA hierarchy structure is not advertised to the MH implies that the MH cannot decide which RFA is the crossover FA to be used as the target of its next regional registration. Hence, the MH always uses the GFA as the target of any regional registrations within the FA hierarchy.

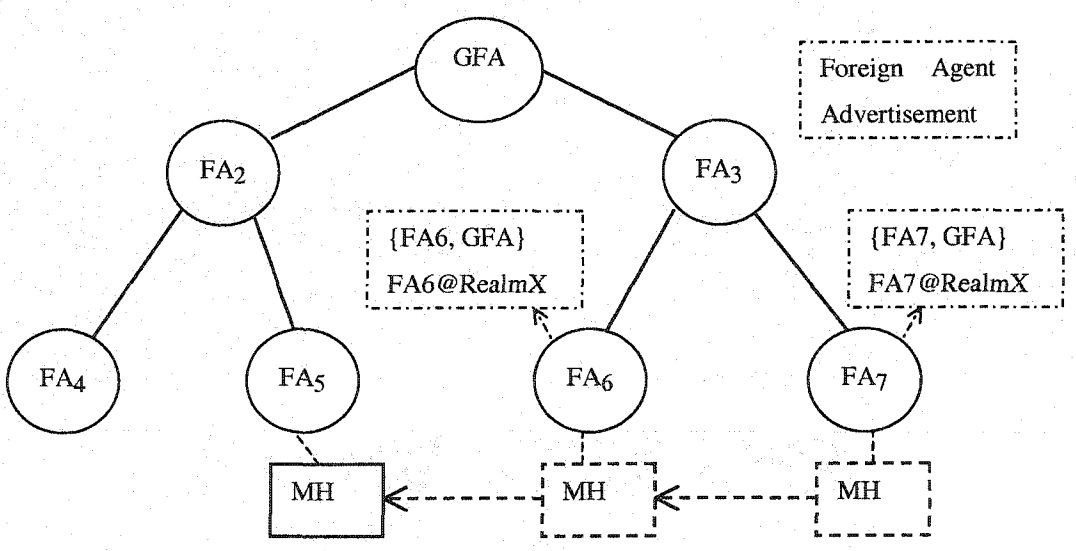


Fig. 8. A foreign agent hierarchy with a sample of agents' advertisements.

3.3 A Regional Registration Processing Framework

While performing an intra-hierarchy handoff, the MH's resulting registration can be classified as: a regional registration implying a local handoff to be replied to by the crossover FA, or a home registration associated with a local handoff (HR-LH), if the timing to send a home registration coincides with a local handoff. In this section, we present a regional registration processing framework and associated tunneling consistency mechanisms. We opt for a signaling design methodology that does not require the smooth handoff mechanism to maintain tunneling consistency, leaving such functionality to be optionally used by the MH to further reduce potential packet loss. In our signaling design, the crossover FA triggers the tunneling consistency mechanism upon receiving a regional registration request. Thus, race conditions, stemming from the crossover FA's inability to properly identify its status as a "crossover FA" for a certain registration request as previously identified within MIP_RR (see section 3.1), are prevented. In the following sections, we present our proposed framework for regional registrations processing, explaining how the smooth handoff mechanism, if used by the MH, can coexist with our proposed approach. In addition, we discuss how authentication and replay protection can be performed for protocol messages. Furthermore, we suggest a new replay protection update mechanism, which propagates the new identification value assigned by the crossover FA to the MH, to ensure future successful processing of MH's registrations by upper levels of the FA hierarchy. Furthermore, we compare and contrast our proposed approach versus MIP_RR using qualitative and analytical comparison.

3.3.1 Operational Overview

When the MH sends a regional registration request, it is propagated upwards in the hierarchy until it reaches the crossover FA. The crossover FA generates a regional registration reply switching the tunneling path for the MH from the old path to the new path. The regional registration reply is propagated down the new path until it reaches the MH. Any future data packets received at the crossover FA for the MH are tunneled through the new path, alleviating the need for the old path.

We suggest using a *Deregistration mechanism*, as a tunneling consistency mechanism, triggered by the crossover FA, by which a *binding update message with*

lifetime equal to 0 is propagated through the MH's old path originating from the crossover FA. A similar approach, albeit relying on home registrations only, was previously proposed and implemented in the Dynamics Hut Hierarchical mobile IP implementation [27]. Furthermore, we require binding update delivery acknowledgement by the receiving FA in response to the deregistration message. Consequently, each RFA, beneath the crossover FA in the old path, receiving the binding update message from its parent FA performs the following steps.

- Note the current tunnel endpoint for the MH,
- Delete the MH's visitor entry,
- Generate a deregistration message to the noted tunnel endpoint (one of its children FA),
- Generate a binding acknowledgment message back to the sender FA (its parent FA).

This process repeats at each intermediate RFA until the BU message reaches the leaf FA that was previously serving the MH (old FA), hence does not have a tunnel endpoint for the MH. If a RFA does not receive a binding acknowledgment message from the tunnel endpoint (one of its children FA) after a specific time interval, it is responsible for resending the BU message, until an acknowledgment is received.

The proposed tunneling consistency mechanism ensures that the old path entries for the MH are cleared in a timely fashion. Fig. 9 illustrates the proposed signaling message flow for regional registrations along with the associated tunneling consistency mechanism invoked by the crossover FA. If the MH is requesting simultaneous binding [43], i.e., the desire to maintain the old and new path simultaneously, in its regional request, such deregistration mechanism is not triggered by the crossover FA.

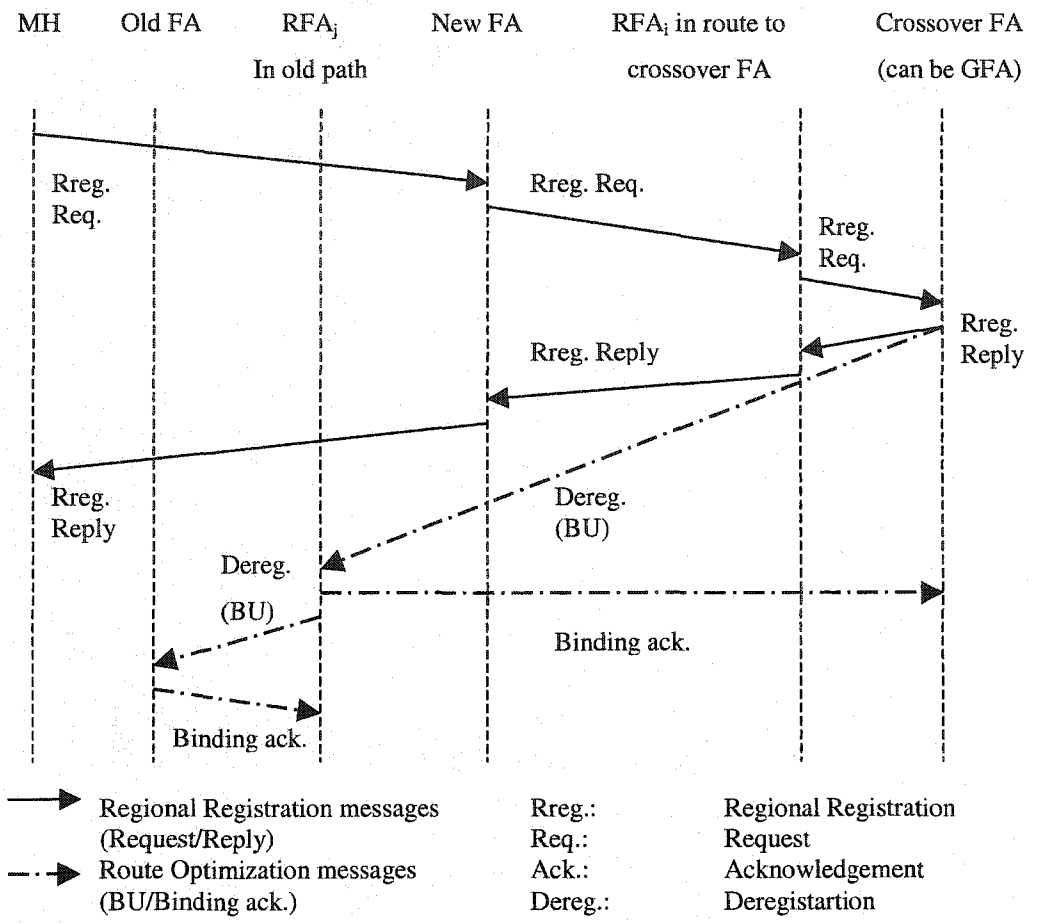


Fig. 9. Proposed signaling message flow for regional registration.

3.3.2 Authentication and Replay Protection

Any binding update or acknowledgement message exchange between foreign agents is authenticated by a route optimization authentication extension [47] based on the pre-established security associations between each parent and child FA. Replay protection is provided for binding updates by usage of the identification field within the binding update and acknowledgment messages [47] according to the replay protection style between each 2 pairs of foreign agents.

The MH authenticates its regional registration request using its registration key shared with the foreign agent hierarchy by appending a MH-GFA authentication extension. The crossover FA checks the authenticity of the request by confirming the authenticator value supplied by the MH, and makes sure that this indeed is the latest message from the MH (message freshness). Replay protection for regional registration requests is provided through the usage of the identification field within the request and reply messages [43] according to the replay protection style selected by the MH for regional registrations.

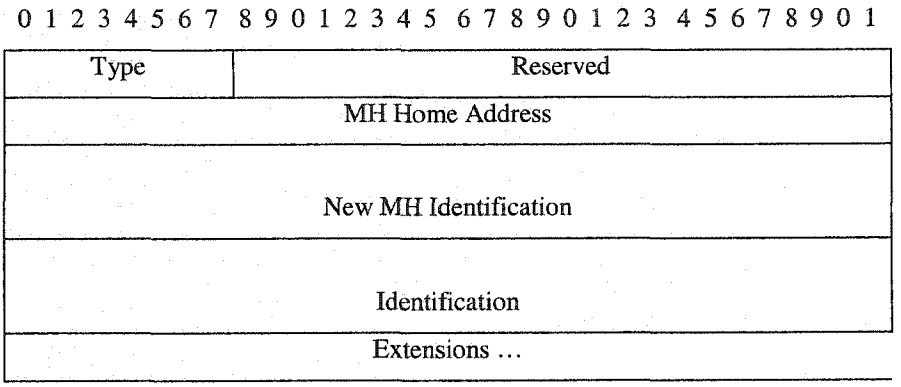
Timestamp replay protection is processed according to [43]. Nevertheless, the FAs individual clocks along with the MH's clock, used to generate the timestamps, need to be synchronized. In such case, any newly generated timestamps by intermediate FAs need not be distributed in the FA hierarchy.

The FA hierarchy only advertises the GFA, but not the hierarchy itself. Thus, in the case of nonce replay protection, the MH associates the identification value supplied within the registration reply with the current GFA. Any intermediate RFAs record the current nonce value for future use, if such intermediate RFA is in a position to reply to a future regional registration request. If an RFA generates a new nonce value, a mechanism is needed to disseminate this new nonce value to higher FAs in the hierarchy, since any of these FAs might be next to authenticate future regional registration requests from this MH[5]. We propose that the RFA generating the new nonce value, the crossover FA, sends a *replay protection update message* upwards in its FA hierarchy. This new Mobile IP message (Fig. 10) propagates upwards all the way to the GFA and contains the MH home address, along with the new identification value and is authenticated by means of a FA-FA authentication extension. Intermediate RFAs in the path towards the GFA associate the new identification value with the MH. In such manner, such RFAs are capable of authenticating any future regional registration requests by the MH. Similarly, in case of timestamp replay protection, the same idea of propagating the generated timestamp using

[5] In nonce replay protection, the crossover FA generates the new nonce value that the MH supplies in its next regional registration request.

the replay protection update message can be equally applied, if general clock synchronization cannot be achieved.

Alternatively, another solution to provide replay protection is through the announcing FA by means of a challenge-response mechanism [46]. In such case, timestamps or nonces are not needed between the MH and the GFA.



Type	Message Type.
Reserved	Reserved. Sent as 0; ignored on reception.
MH Home Address	The home address of the MH to which this message refers.
New MH Identification	A 64-bit number, the new identification value for the MH stored by the receiver for future use.
Identification	A 64-bit number, assigned by the sender to assist in protecting against replay attacks.

Fig. 10. Replay protection update message format and fields.

3.3.3 Coexistence with Smooth Handoff Mechanism

If the MH is requesting the usage of the smooth handoff mechanism (section 2.3.2), the new FA transmits a BU on behalf of the MH to the old FA (BU_{NewFA}), which

authenticates the BU and sends a binding acknowledgment to the new FA, which delivers it to the MH. Meanwhile, a deregistration message is propagated through the old path by the crossover FA towards the old FA. The following two cases can arise.

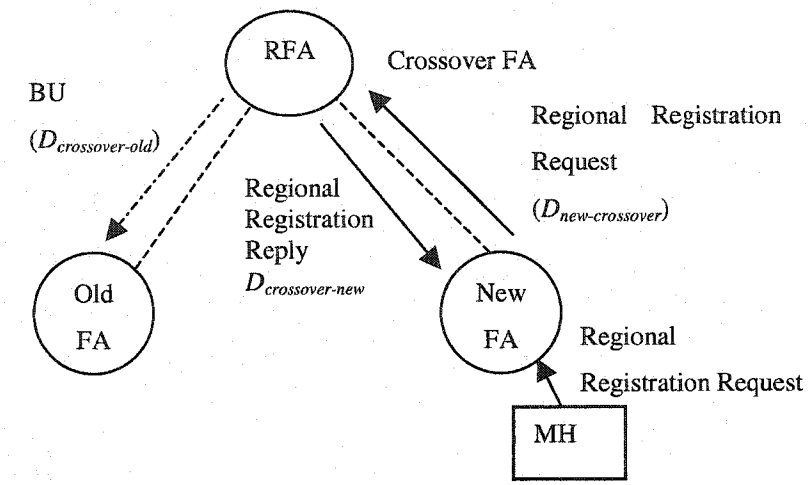
1. *BU_{NewFA} reaches the old FA before the deregistration message.* The MH's visitor entry is deleted, and a binding cache entry pointing to the new FA is created. Such binding cache entry is eventually removed when the old FA receives the deregistration message, since such message is simply a BU with zero lifetime.
2. *Deregistration message reaches the old FA before BU_{NewFA}.* The MH's visitor entry is deleted upon first receiving the deregistration message. In such case, the old FA receives BU_{NewFA} while no visitor entry exists for the MH. Thus, the old FA denies the BU generating a negative binding acknowledgment back to the new FA.

If buffering services are provided by foreign agents to the MH [49] and the deregistration message reaches the old FA before BU_{NewFA}, the old FA does not know how to forward the buffered packets to the new FA while it needs to delete the MH's visitor entry. Thus, a mechanism is needed which informs the old FA about the MH's new FA when the MH is not using the smooth handoff mechanism (i.e., the MH is not providing the old FA information to the new FA). Even if the MH is smooth handoff enabled, individual foreign agents may not be advertising their own IP address, or their NAI to allow the MH to use either for smooth handoff purposes. Such mechanism leverages the existence of a hierarchy, and relies on propagating the new FA IP address, provided by the new FA itself, with the regional registration request along the new path, then along the old path with the tunneling consistency binding update messages. We present the details of such mechanism as part of our home registration processing framework, which we detail in section 3.4.

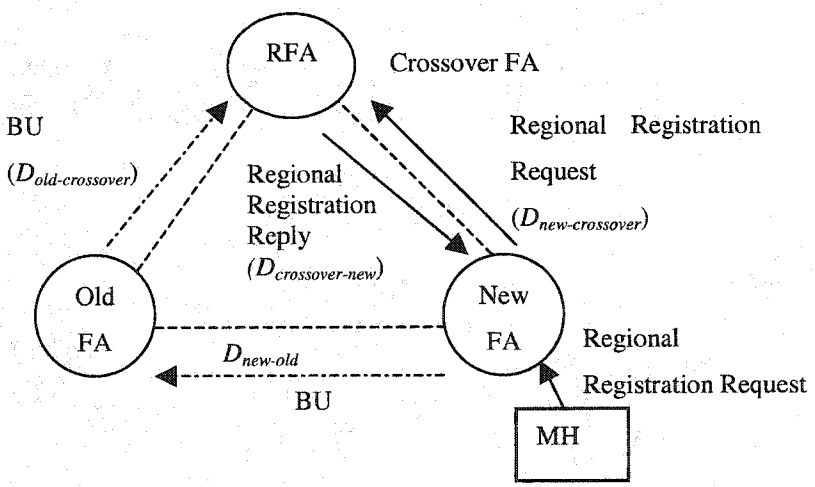
3.3.4 Analysis and Comparison

We analyze the proposed regional registration framework and associated tunneling consistency mechanism using the delay measures introduced earlier in TABLE 1. Fig. 11

abstracts the regional registration processing along with involved delays in our proposed approach as well as the MIP_RR's approach for comparison purposes.



(a)



(b)

Fig. 11. Abstracted view of regional registration processing along with involved delays. (a) Proposed (b) MIP_RR [30].

We define *regional registration latency* as viewed by the MH to be the time interval between submitting a regional registration request and receiving a regional registration reply. After receiving a regional registration reply, the MH is assured that its tunneling path within the foreign agent hierarchy has been switched to the new path. Furthermore, such regional registration reply indicates to the MH the completion of the current handoff event from the foreign agent hierarchy perspective (network-layer point of view). We introduce a new delay measure $D_{crossover\text{-}new}$ to be the delay to generate and transmit the regional registration reply by the crossover FA, to be relayed by all intermediate RFAs to reach the new FA, starting when the crossover FA receives the MH's registration request. Such delay is dependent on link, and queuing delays on each intermediate link between the crossover FA and the new FA along the new path in the FA hierarchy, and on registration reply processing delays at each intermediate RFA on this path. In general, the MH's perceived regional registration latency can be formulated as shown in (2).

$$\text{Registration latency} = D_{new\text{-}crossover} + D_{crossover\text{-}new} + \begin{array}{l}\text{Time for the new FA to receive the request from the MH +} \\ \text{Time for the MH receive the reply from the new FA}\end{array} \quad (2)$$

We define $D_{crossover\text{-}old}$ to be the delay to generate and transmit a BU down the old path by the crossover FA, to be relayed by all intermediate RFAs to reach the old FA, starting when the crossover FA receives the MH's registration request. In TABLE 2, we compare the proposed approach versus MIP_RR, in terms of the following aspects.

1. The MH's perceived registration latency.
2. The delay before the tunneling consistency mechanism is initiated ($D_{InitConsistent}$) starting when the new FA receives the MH's request.
3. The delay before the tunneling consistency mechanism runs to completion ($D_{Consistent}$).
4. The delay before the old FA can forward any packets (buffered or received afterwards) to the new FA ($D_{Forward}$), if any, starting when the new FA receives the MH's request.

For our proposed approach, we are assuming the availability of a mechanism that propagates the new FA IP address information to the old FA (see section 3.3.3), while the MH is not using the smooth handoff mechanism. If the MH is using the smooth handoff mechanism, only $D_{Forward}$ is affected, according to whether the BU from the new FA, or the tunneling consistency BU from the parent FA reaches the old FA first.

Analyzing the formulas presented in TABLE 2, we conclude that the initiation of the tunneling consistency mechanism in our proposed framework is dependent on the number of RFAs between the new FA and the crossover FA (number of intermediate levels), and the corresponding link and queuing delays on the new path. On the other hand, for MIP_RR such measure is dependent on link, queuing and routing delays between the new and old FA, respectively. The mathematical relationship ($\leq$, $\geq$) between $D_{new\text{-}old}$ and $D_{new\text{-}crossover}$ holds the key to which approach is faster in ensuring tunneling consistency along the old path[6].

TABLE 2

QUANTITATIVE COMPARISON OF THE PROPOSED APPROACH VERSUS MIP_RR

Measure	Proposed	Proposed with MH using smooth handoff	MIP_RR [30]
Registration latency	As given in (2)	As given in (2)	As given in (2)
$D_{InitConsistent}$	$D_{new\text{-}crossover}$	$D_{new\text{-}crossover}$	$D_{new\text{-}old}$
$D_{Consistent}$	$D_{InitConsistent} + D_{crossover\text{-}old}$	$D_{InitConsistent} + D_{crossover\text{-}old}$	$D_{InitConsistent}+D_{old\text{-}crossover}$
$D_{Forward}$	$D_{new\text{-}crossover} + D_{crossover\text{-}old}$	Min $\{D_{new\text{-}old}, D_{new\text{-}crossover} + D_{crossover\text{-}old}\}$	$D_{new\text{-}old}$

[6] Assuming $D_{crossover\text{-}old}$ is equivalent to $D_{old\text{-}crossover}$.

Utilizing the smooth handoff mechanism as the basis for the tunneling consistency mechanism, allows initiating such mechanism from the earliest possible point on the RFA tree, which is the new FA. However, as pointed earlier in section 3.1, the use of such approach coupled with the dependence on the time of receiving the corresponding regional registration request by the crossover FA open the door for potential race conditions. Even in the absence of such potential race condition, we argue that our framework presents a viable approach to ensure hierarchy tunneling consistency and inform the old FA about the new FA when the MH is not enabled to use the smooth handoff mechanism, or the foreign agents are not advertising any means for personal identification such as their IP addresses, or their network access identifiers.

3.4 A Home Registration Processing Framework

The MH performs a home registration when it first enters the foreign domain, and periodically to maintain its home mobility binding. For intra-hierarchy handoffs, we classified home registrations (section 2.3.1) according to whether a local handoff from one FA to the other is involved. If no handoff is involved, the home registration is propagated in a path that has MH's visitor entries established using a previous regional registration. In such case, no tunneling consistency mechanism is required since there is no crossover FA, and no old path with visitor entries to be deleted. On the other hand, if a local handoff is involved (HR-LH, see section 2.3.1), the registration request is propagated in an un-established path until it reaches a crossover FA that forwards the request over an already established path until it reaches the GFA. In the latter case, simply applying a tunneling consistency mechanism that deletes the MH's visitor entries in the old path maintains the tunneling consistency, but degrades handoff performance since the MH can not be reached until the registration reply from the HA establishes the new path. HR-LH rate of occurrence depends on the MH mobility pattern, home registration lifetime, and when the MH receives foreign agents' advertisements. For instance, a MH can initiate a home registration that does not involve a local handoff, move to another FA due to a received FA's advertisement, and subsequently generate another home registration, which at this time involves a local handoff.

In this section, we present 2 novel approaches for processing of home registrations involving local handoffs within the same foreign agent hierarchy. We analyze and contrast both approaches, and later present a simulation study comparing their usage (section 3.5). Both approaches attempt to exploit the hierarchy structure, in order to optimize the MH handoff while waiting for the home registration reply to be received from the HA. The KOPA (Keep Old Path Alive) approach (section 3.4.1) follows the same line of thought as MIP_RR [30] in attempting to keep the old path "alive" and tunnel packets to the MH's new FA until the handoff completes by receiving a home registration reply. Nevertheless, such task is performed without relying on the MH's usage of the smooth handoff mechanism as a required component, and designed to prevent previously identified race conditions (section 3.1). The SINP (Switch Immediately to New Path) approach (section 3.4.2) emphasizes the local handoff aspect and switches the MH's tunneling path within the hierarchy immediately to follow the new path without waiting for the home registration reply. We focus here on home registrations associated with intra-hierarchy handoffs. In section IV, we present a foreign agent hierarchy cooperation-based framework to handle home registrations associated with inter-hierarchy handoffs.

3.4.1 KOPA: Keep Old Path Alive Approach

The KOPA approach relies on the crossover FA to initiate a mechanism by which the MH's old path is kept alive until the home registration is received, and consequently creating visitor entries in the new path. "Keeping the old path alive" implies performing the following two steps.

1. Replace visitor entries in the old path with binding cache entries, with a specified lifetime, that point to the same visitor entry's tunnel endpoint for the MH (one of the children FAs),
2. Inform the old serving FA about the MH's new FA to tunnel to it any already buffered or future data packets that arrive at the old FA.

Such procedure ensures future tunneling consistency since a visitor entry is replaced with a binding cache avoiding future erroneous decisions by RFAs when receiving a future MH's regional registration request. Propagating the new FA information to the

crossover FA and then down the old path is necessary since we do not rely on the MH's usage of the smooth handoff mechanism. Nevertheless, if the MH is using the smooth handoff mechanism, we present how our approach coexists with such mechanism in section 3.4.1.4. Fig. 12 illustrates an abstracted view of the KOPA approach for processing home registrations involving local handoffs.

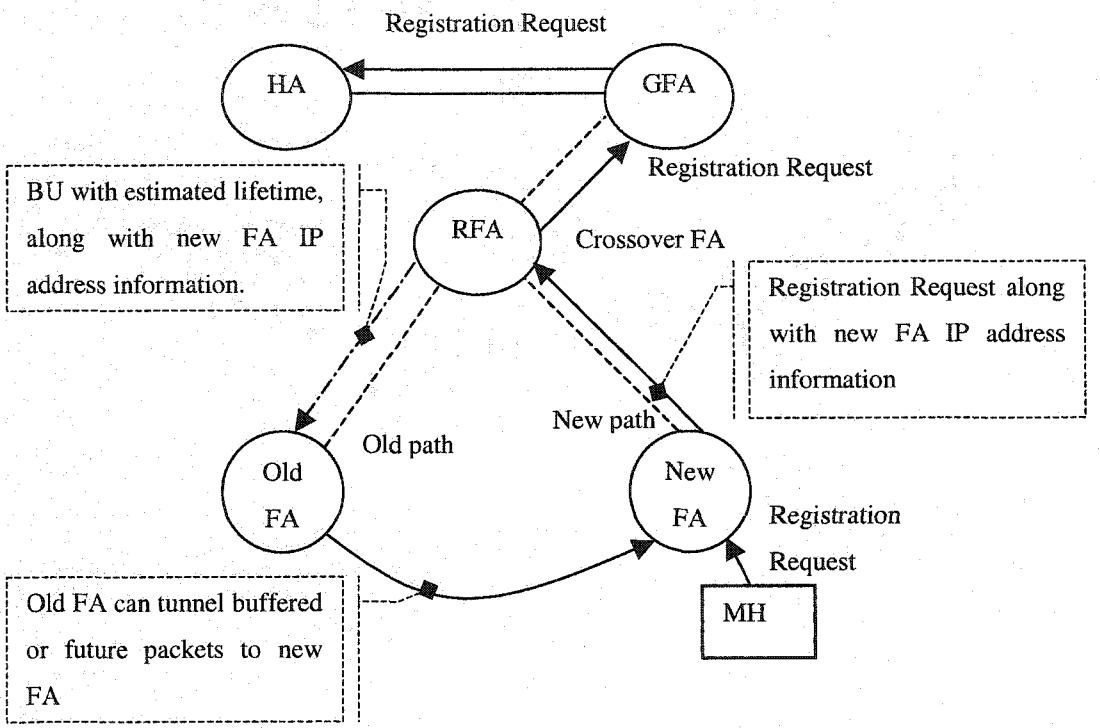


Fig. 12. The KOPA approach for processing home registrations involving local handoffs.

The new FA, upon receiving the MH's home registration request, propagates the registration request upwards in the new path, appending his own IP address information (see section 3.4.1.3). This registration request eventually reaches the crossover FA. The crossover FA generates a binding update message with an estimated lifetime (see section 3.4.1.1) down the old path to its visitor entry tunnel endpoint appending the new FA information it extracts from the received registration request. In addition, the crossover FA propagates the home registration request upwards towards the GFA[7] which sends it to the HA. The crossover FA only propagates the new FA information down the old path, but does not use such information for tunneling data packets to the MH even after a home registration reply is received. This is due to the fact that the next MH's regional registration might terminate at a crossover FA that is at a lower hierarchy level than the current crossover FA, hence the current crossover FA will not be informed to update its MH's tunnel endpoint. Thus, the crossover FA always establishes the MH's tunnel endpoint as its child FA that originally forwarded the registration request.

Each RFA, beneath the crossover FA in the old path, upon receiving the binding update message from its parent FA performs the following steps.

- Note the current tunnel endpoint for the MH,
- Delete the MH's visitor entry,
- Create a binding cache entry with the specified lifetime (pointing to the noted tunnel endpoint),
- Generate a binding update message to the noted tunnel endpoint (one of its children FA),
- Generate a binding acknowledgment message back to the sender FA (its parent FA).

This process repeats until the BU message along with the new FA information reaches the FA that was previously serving the MH, hence does not have a tunnel endpoint for the MH. The old FA, armed with the new FA information, can send any already buffered, or future data packets to the new FA that delivers them to the MH. The binding caches in the old path will eventually expire. Meanwhile, the home registration

---

[7] If the crossover FA is the GFA, then he simply forwards the request to the HA.

reply should be received from the HA switching the MH's tunneling path to the new path alleviating the need for the old path. Fig. 13 illustrates the signaling message flow for the KOPA approach.

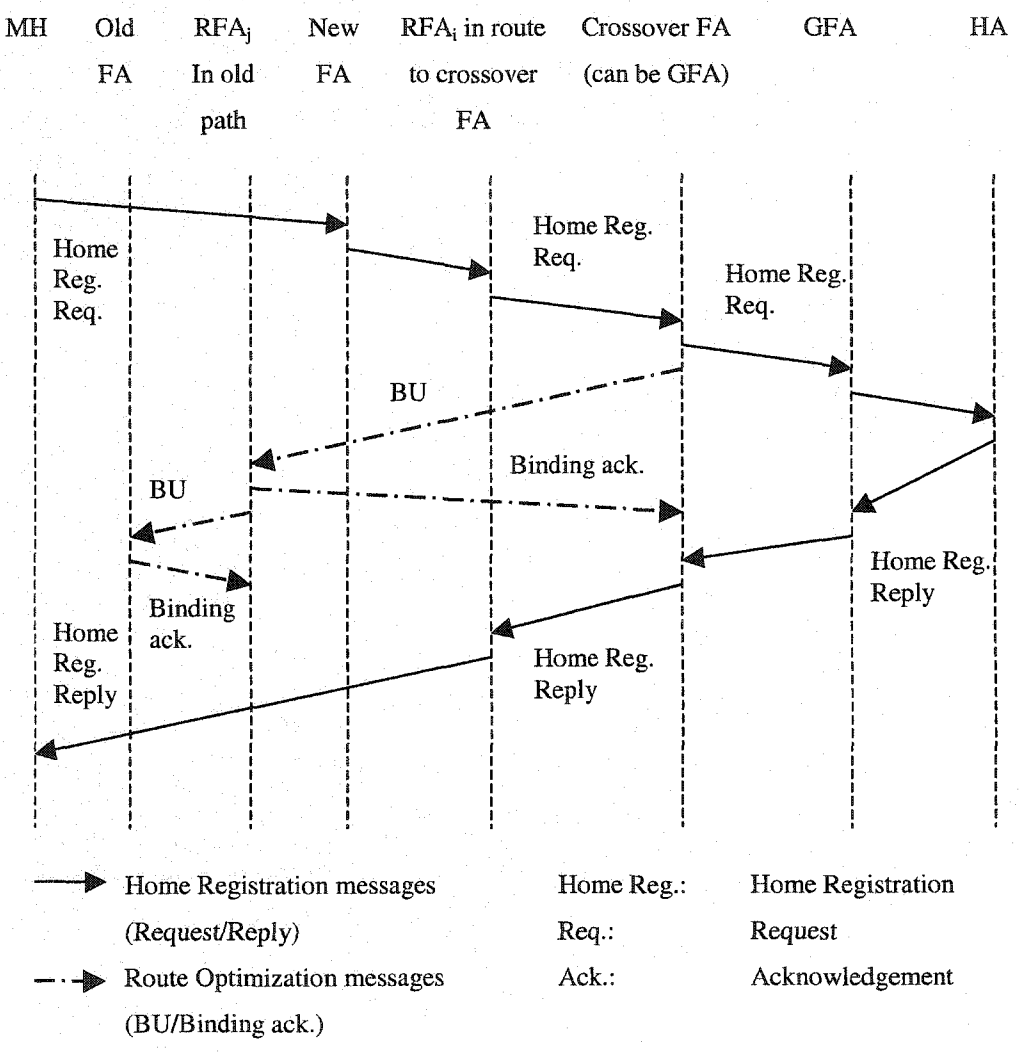


Fig. 13. Signaling message flow in the KOPA approach.

The following design issues need to be addressed for correct and efficient operation of the KOPA approach.

1. What lifetime the crossover FA uses for the binding update?
2. How the crossover FA ensures authentication and replay protection for the MH's home registration request, in order to initiate the tunneling consistency mechanism?
3. How the new FA information is propagated along the new path with the MH's home registration request, and later along the old path with the binding update message?
4. Coexistence with smooth handoff mechanism if used by the MH?

We next present our solution for each of these issues highlighting possible alternatives and design tradeoffs.

3.4.1.1 Binding update lifetime

The old path needs to be kept "alive" until the home registration reply establishes the new tunneling path for the MH in the foreign domain. Actually, the crossover FA starts tunneling data packets to the new path after it has received the home registration reply and established the sending child FA (in the pending registration record) as the tunnel endpoint for the MH. Hence, the crossover FA can switch to the new path after the registration request it propagates upwards, on behalf of the MH, eventually reaches the HA, is processed, and a home registration reply is received by the crossover FA. The time interval which is the difference between the time the crossover FA propagates the registration request upwards and the time it receives a corresponding home registration reply, represents the home registration latency as seen at each involved RFA. The duration of such time interval is affected by the following factors: the number of hierarchy levels and required request/reply messages processing at each level between the crossover FA and the GFA; the round trip time between the GFA and the HA; and the home registration processing time by the HA.

The binding update lifetime can be readily selected as the remaining MH's registration lifetime at the crossover FA when receiving the registration request. Nevertheless, such remaining lifetime is dependent on when the MH initiates a home

registration, might not be enough for the request to reach a distant HA, be processed by the HA, and for a registration reply to reach the crossover FA. Thus, we suggest incorporating measuring the home registration latency as part of the BU lifetime selection process, as explained next.

We propose to let RFAs measure the aforementioned time interval (Δt) for every approved MH's home registration. Such decision implies that a RFA notes the time it forwards a home registration request towards the GFA ($t_{forward\_request}$), storing it for future use in a pending registration record. When the RFA receives the approved home registration reply ($t_{receive\_reply}$), the stored forwarding time is extracted and the time interval is computed according to (3). This measure provides a crossover FA with an estimate to aid in selecting the BU lifetime, since it represents the most recent home registration latency as seen by this RFA

$$\Delta t = t_{receive\_reply} - t_{forward\_request} \tag{3}$$

The estimated measure is updated at each RFA in any hierarchy path towards the GFA that the MH's home registration request is propagated through. We only store the most recent measured value, and not compute the estimate as a weighted average of old and new values, since this process might not be that frequent depending on the time intervals between consecutive home registrations. However, a crossover FA cannot solely depend on this measure to compute the BU lifetime, since the crossover FA might not yet store this measure depending on the MH's mobility pattern. Fig. 14 depicts an example scenario where a crossover FA does not store a home registration latency measure for a MH. When the MH enters the foreign domain, it sends its first home registration to FA_6. Hence, home registration latency measures are stored by {FA_3, GFA} because the home registration reply flows in the path {GFA, FA_3, and FA_6}. Next, the MH hands off regionally from FA_6 to FA_5 with the GFA being the crossover FA. Later, the MH hands off from FA_5 to FA_4 and sends a home registration request. At such time, FA_2 is the crossover FA, which does not store a home registration latency measure for the MH.

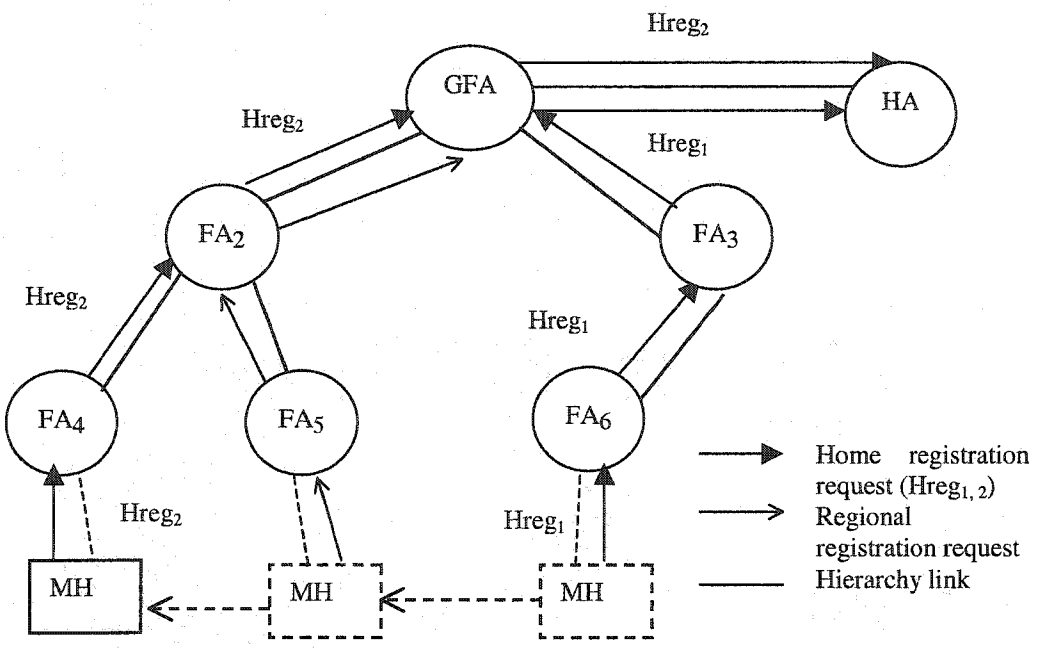


Fig. 14. A scenario where the crossover FA does not store home registration latency.

We suggest a BU lifetime selection mechanism that combines the knowledge of the remaining registration lifetime, and the measured most recent home registration latency at each crossover FA as shown in (4).

$$\textit{BU lifetime} = \text{Max}\ \{\textit{home reg. latency}, \alpha * \textit{remaining reg. lifetime}\} \qquad (4)$$

Where $0 < \alpha \leq 1$.

Analyzing equation (4), the fraction α represents the percentage of the remaining registration lifetime to be used in comparison to the home registration latency. We suggest using a value of 0.5 for α[8], since the initial value for the registration lifetime is set by the MH and later approved or modified by the HA, and the MH most probably will

[8] The value of α is implementation dependent, as what fraction of the remaining registration lifetime is desired to affect the selection of the BU lifetime.

issue a home registration request long enough before the registration expiration. Hence, the remaining registration lifetime might be too much larger than the home registration latency, depending on when the MH initiates a home registration request. If the home registration latency is not known, the BU lifetime is selected as a fraction of the remaining registration lifetime. Otherwise, the home registration latency guides the selection process by acting as a lower/upper bound for the BU lifetime.

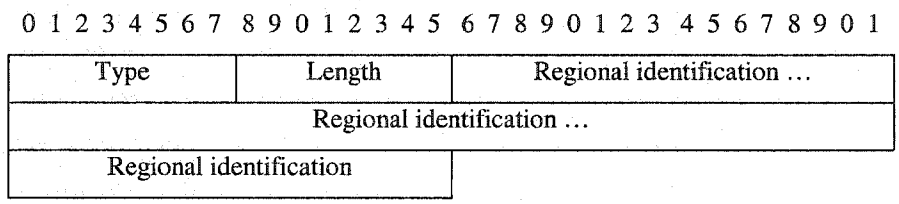
3.4.1.2 Authentication and replay protection

In the KOPA approach, the crossover FA acts upon the MH's home registration and generates binding updates down the MH's old path, hence altering the MH's tunneling state in affected RFAs. If the crossover FA cannot authenticate the received home registration, then such approach would not be feasible. However, this is a home registration with no regional authentication information. Hence, we exploit the presence of a hierarchy structure and the existence of an old and new path to the MH by extending such home registration, a HR-LH request (see section 2.3.1), as a special case of a *combined home and regional registration*. The home registration aspect ensures that the request is propagated all the way to the HA to renew the MH's home mobility binding. The regional registration aspect permits the crossover FA to act upon the received request and proceed in generating the keep "alive" binding updates. Consequently, we require that the MH authenticate any HR-LH requests by using a MH-GFA authentication extension. Moreover, the home registration request does not contain regional identification information[9], creating a regional replay protection problem. The crossover FA cannot guarantee the "freshness" of the request unless the MH includes its regional identification information along with its home registration. Thus, we define a *local replay protection extension* (Fig. 15) to be supplied by the MH with a HR-LH request to ensure the feasibility of the KOPA approach. The MH uses the local replay protection extension to supply its current regional identification value; to enable the crossover FA's processing mechanism of the home registration request.

The local replay protection extension is included with any other non-authenticated extensions pertaining to the MH and the hierarchy of foreign agents, after the Mobile-

[9] Recall that home and regional replay protection mechanisms are separate [30].

Home authentication extension [43], but before the MH-GFA authentication extension. In such manner, the home-targeted part of the registration request message along with its authentication is not affected when the foreign agents hierarchy removes the local replay protection extension along with its authenticating extension.



Type	Extension Type.
Length	8-bit unsigned integer, in this case is equal to 8 (Length of the extension in octets excluding extension type and length).
Regional identification	A 64-bit number, The current regional identification value for the MH.

Fig. 15. Local replay protection extension format and fields.

The crossover FA must authenticate the MH's request and ensure the validity of the supplied regional identification value. If the MH fails the authentication or the identification validity test, then the crossover FA generates a regional registration reply back to the MH with appropriate error code, but still forwards the home registration portion of the request upwards for normal home registration processing[10]. A regional registration reply with an authentication error code prompts the MH to attempt a new home registration without supplying its regional information while appending a replay protection extension. Such registration is processed according to [30]. A regional registration reply with an identification mismatch error code would provide the MH with

[10] The crossover FA forwarding of the home registration request, even through the MH failed regional authentication, is required to maintain the home mobility binding of the MH.

means to re-synchronize its identification information with the hierarchy depending on which style of replay protection is in use [43]. In such case, the MH resends its registration request using the new regional identification value supplied in the previous regional registration reply. Such request serves to ensure the execution of the tunneling consistency mechanism by the crossover FA.

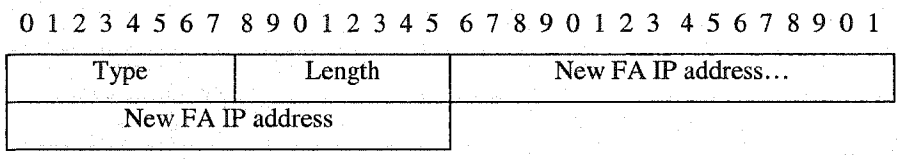
If the crossover FA is able to successfully authenticate the MH, it initiates the binding updates down the old path as previously explained, and forwards the MH's request unchanged upwards towards the GFA. Such processing allows RFAs in higher levels than the crossover FA to know the current regional identification used by the MH. Eventually, this request reaches the GFA, which removes any regional information extensions (e.g., the local replay protection extension along with the authenticating extension) and forwards the request to the HA. If the MH is using nonce replay protection, then a new nonce value needs to be generated and sent back to the MH. In this case, we suggest that the GFA perform this function by placing the new nonce value in a local replay extension to be appended at the end of the home registration reply received from the HA, and authenticated using a MH-GFA authentication extension. The registration reply flows down the hierarchy to reach the MH, informing all RFAs of the new nonce value generated for this MH. If timestamp replay protection is used, then such processing is not needed.

Exchange of binding updates and acknowledgments between foreign agents is authenticated using the route optimization authentication extension. Any FA is capable of ensuring the freshness of the BU message (replay protection mechanism) it receives by inspecting the BU's identification field [47].

3.4.1.3 New FA information

The new FA information needs to be propagated from the new FA along to the new path to the crossover FA, and then along the old path to the old FA (see Fig. 12). Along the new path, we use the home registration request to convey this information, while along the old path we use the binding updates designated to implement the KOPA approach.

We define a *local care-of address extension* to carry the new FA IP address information along the new path (Fig. 16). The new FA appends this extension to the HR-LH request it receives from the MH. In addition, it authenticates this information by appending a FA-FA authentication extension using the security association it shares with its parent FA. The home registration request and the added extensions are propagated in the new path up to the crossover FA. Each intermediate RFA in the new path removes the FA-FA authentication extension it receives and maintains the local care-of address extension inserted by the new FA address, while authenticating the request to its parent FA by using a FA-FA authentication extension. This process repeats until the registration request reaches the crossover FA. After authenticating the registration request (see section 3.4.1.2), the crossover FA extracts the local care-of address extension to use in formulating the binding updates down the old path, removes the local care-of address extension and the corresponding authentication extension from the registration request, and propagates the stripped down registration request upward in the hierarchy toward the GFA.



Type	Extension Type.
Length	8-bit unsigned integer, in this case is equal to 4 (length of the extension in octets excluding extension type and length).
New FA IP address	The IP address of the new FA.

Fig. 16. Local care-of address extension format and fields.

The crossover FA sends a binding update to its current tunnel endpoint for the MH (The first RFA in the old path) appending the local care-of address extension to the binding update before the route optimization authentication extension used to authenticate the BU message. The BU lifetime is set as explained in section 3.4.1.1. The BU care-of address is set to the MH home address. The presence of the local care-of address extension along with a non-zero lifetime and setting the BU care-of address to the MH home address dictates the following processing steps at each RFA in the old path receiving this BU.

- Delete current visitor entry for the MH retaining the current tunnel endpoint;
- Create a binding cache entry pointing to the tunnel endpoint;
- Send back a binding acknowledgment message back to the sender FA (father FA);
- Send a new binding update message to the tunnel endpoint using the provided values for the BU lifetime and MH care-of address while appending the local care-of address extension and authenticating the BU message using a FA-FA authentication extension.

This process repeats until the BU traverses the old path and reaches the old FA that does not have a tunnel endpoint to the MH. The old FA deletes its MH's visitor entry, creates a binding cache entry pointing to the new FA[11] using the provided BU lifetime; and acknowledges the BU message back to its father FA.

The same approach and data extensions providing the new FA IP address from within the hierarchy can be similarly used when processing regional registration requests (section 3.3) to allow for forwarding of any buffered data packets from the old FA to the new FA without relying on the MH's usage of the smooth handoff mechanism. The main difference would be that a zero lifetime is provided in the binding updates propagated down the old path resulting in no binding cache entries created along the old path. In this case, the old FA can forward any already-buffered packets to the new FA, and can assume a default lifetime for the time interval in which it performs tunneling of any future data packets to the new FA.

---

[11] The old FA extracts the new FA IP address from the local care-of address extension.

3.4.1.4 Coexistence with smooth handoff mechanism

If the MH is using the smooth handoff mechanism, the old FA will eventually receive 2 binding update messages: the BU message from the father FA which implements the KOPA approach ($BU_{FatherFA}$); and the BU message from the new FA forwarded on behalf of the MH as part of the smooth handoff mechanism (BU_{NewFA}). These 2 BU messages contain the same new care-of address information for the MH that is the new FA IP address. The BU that reaches the old FA first causes the old FA to delete its MH's visitor entry and create a binding cache entry using the supplied lifetime. When the second BU reaches the old FA, it provides the same care-of address as the one pointed to by the binding cache entry; hence no care-of address update is performed. Nevertheless, the MH specifies the lifetime in BU_{NewFA}, while the crossover FA specifies the lifetime in $BU_{FatherFA}$.

We suggest the following MH-biased processing by the old FA: *the second BU lifetime is applied only if it is larger than the remaining lifetime of the binding cache entry*. Hence, the MH is always granted the largest possible lifetime for the binding cache entry. In this special case, the granted lifetime might be different than the requested lifetime. Thus, the binding acknowledgement message can be modified to include an *approved lifetime* field. We acknowledge that such processing might create binding cache lifetime inconsistencies between the old FA and along the old path to the crossover FA, nevertheless the main purpose was achieved by preventing race conditions, and providing an alternative to inform the old FA about the new FA without relying on the smooth handoff mechanism. If absolute binding cache lifetime consistency along the old path is desired, and the $BU_{FatherFA}$ arrives later at the old FA with the current remaining lifetime being higher than requested (i.e., the remaining lifetime granted due to BU_{NewFA} is greater than requested in $BU_{FatherFA}$), then the binding acknowledgement message to the father FA will include the new lifetime, and can trigger a set of binding updates along the old path upwards towards the crossover FA.

3.4.2 SINP: Switch Immediately to New Path Approach

The SINP approach reinforces the local handoff aspect while the MH is sending a HR-LH request. We note that the HA does not know which local FA is currently serving

the MH, since the home registered care-of address for the MH is the GFA. We argue here that the establishment of the new path within the hierarchy should not be dependent on a home registration reply that only indicates that the GFA has been established as the MH's care-of address. Hence, we capitalize on such issue and view a home registration with an involved local handoff as truly a combined home and regional registration. The regional aspect of the registration is handled by the crossover FA in order to switch the MH's tunneling path from the old path to the new path immediately without waiting for a home registration reply, while the home registration aspect is handled by the HA to renew the MH's home mobility binding. Therefore, we suggest that *the crossover FA switches immediately the MH's tunneling path from the old path to the new path by issuing a regional registration reply in response to the MH's home registration reply.* To enable such functionality, similar to the KOPA approach, the MH formulates the home registration request by supplying any current regional protection information authenticating the request using a MH-GFA authentication extension (see section 3.4.1.2). In addition to generating a regional registration reply, the crossover FA forwards the home registration request upwards in the hierarchy towards the GFA for normal home registration processing by the HA. Furthermore, to ensure tunneling consistency, the old path to the MH is cleared by using a tunneling consistency mechanism triggered by the crossover FA similar to the deregistration mechanism introduced in section 3.3. Fig. 17 illustrates an abstracted view of the SINP approach for processing home registrations involving local handoffs, while Fig. 18 depicts a more detailed signaling message flow.

In essence, the MH generates a specially formulated home registration request, and expects receiving 2 registration replies: a regional registration reply, and a later home registration reply. Thus, involved foreign agents along the new path create two pending registration records: a pending home registration record, and a pending regional registration record. Matching registration replies against pending requests is performed based on the MH's home address and identification value stored in the pending record [43]. For the pending regional request, the MH's local identification value is stored in the pending regional request. Such value most probably will be different than the home identification value. Recall that the MH supplies its local identification value through a local replay protection extension to enable processing by the crossover FA.

The MH requests a home registration lifetime within its home registration request. Such information is targeted to the HA and represent the home binding registration lifetime. The regional registration reply is generated using the remaining registration lifetime of the existing MH's home registration that is available at the crossover FA. Hence, the crossover FA is establishing the new path as the MH's tunneling path for the remaining amount of time previously approved by the HA. No guarantees are provided here that the home registration reply would reach back the GFA and the MH before the new path registration lifetime expires. Nevertheless, we note that the MH, in practice, would periodically renew its home mobility binding long before its expiration. We opted for this approach instead of adopting an estimate approach similar to the one introduced in the KOPA framework (see section 3.4.1.1), since a regional registration reply is generated, hence we are bound by the previously approved lifetime amount by the HA.

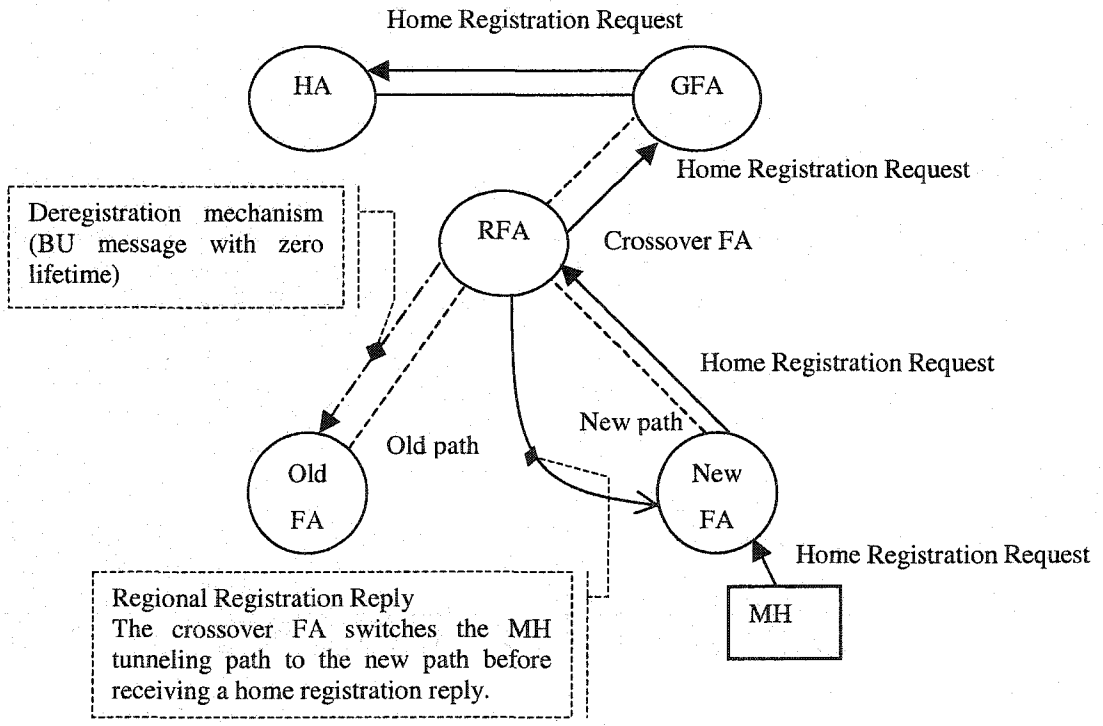


Fig. 17. The SINP approach for processing home registrations involving local handoffs.

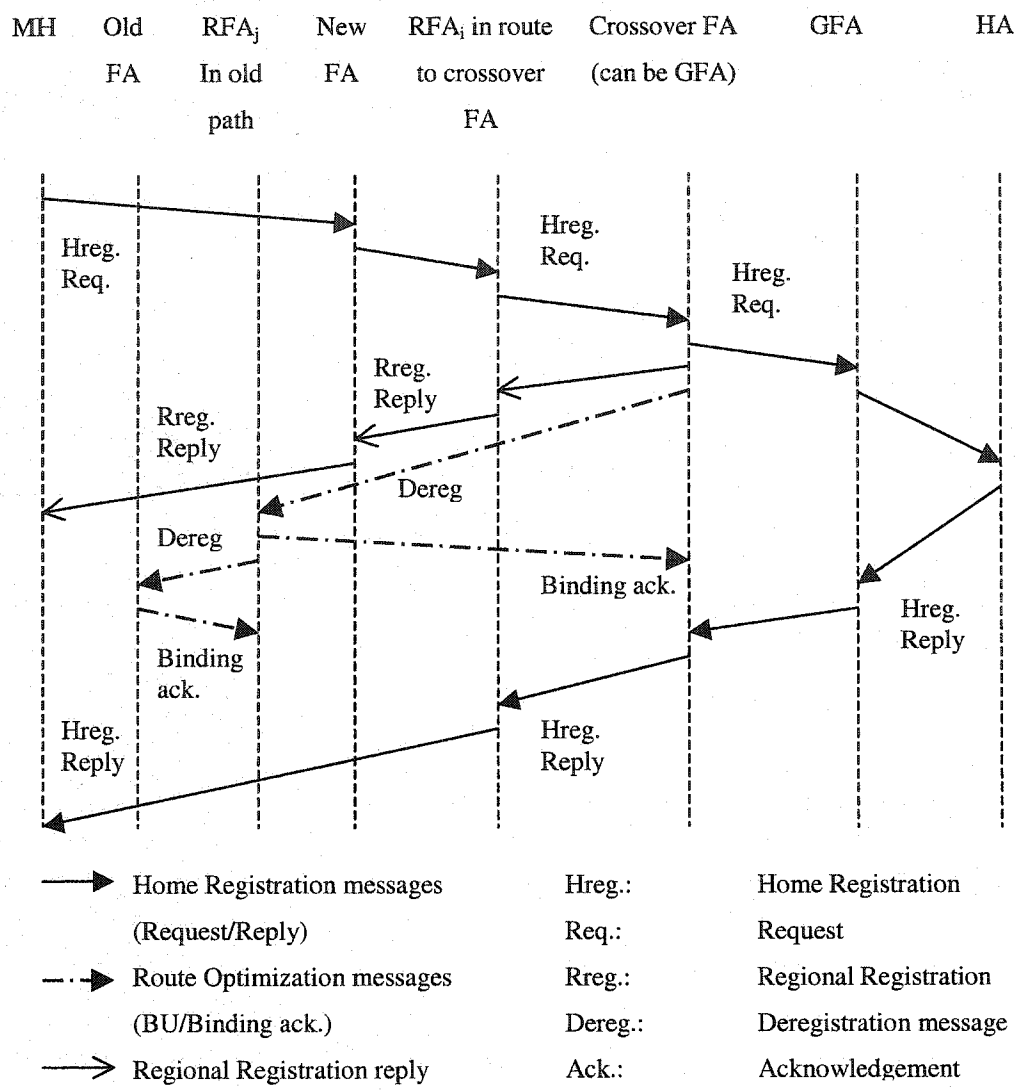


Fig. 18. Signaling message flow in the SINP approach.

Similar to the KOPA approach, the crossover FA is required to authenticate the MH request by inspecting the authenticator value in the MH-GFA authentication extension and checking the validity of the regional identification value provided in the local replay protection extension. If the MH fails either test; a regional registration reply is generated with appropriate error code. Nevertheless, the home registration portion of the request is propagated upwards towards the GFA for normal home registration processing. For

requests successfully processed by the crossover FA, the MH request is propagated upwards towards the GFA that removes any regional extensions and forwards the request to the HA. For timestamp replay protection, the crossover FA forwards the MH request unchanged leaving the local replay protection extension, hence allowing for upper-level RFAs to know the current identification value used by the MH. For nonce replay protection, the crossover FA puts a new nonce value in its regional registration to the MH. Two alternatives are available to inform upper-level RFAs about such new nonce value: propagate a separate replay protection update message upwards in the hierarchy towards the GFA as previously introduced as part of the regional registration framework (section 3.3.2), or append a local replay protection extension containing the new nonce value to the end of the MH request. In the latter case, the resulting message is authenticated using a FA-FA authentication extension.

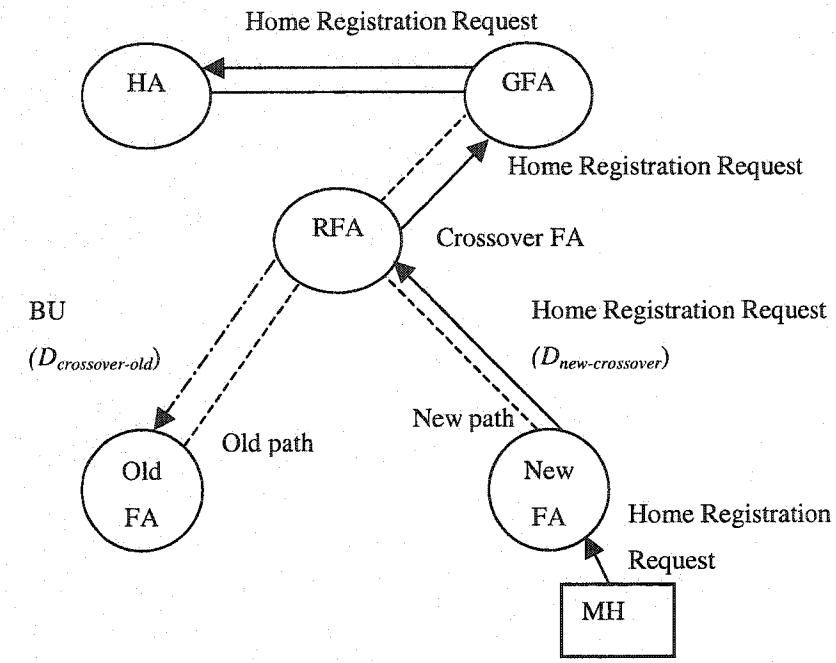
A successful regional registration reply that reaches the MH indicates that the local handoff was successfully processed and the tunneling path has been switched from the old path to the new path. In such case, if the MH does not receive a home registration reply, or receives a home registration reply indicating registration failure by the HA, the MH initiates another home registration request, which in this case would not involve a local handoff, since the local handoff has been already processed during the previous HR-LH. The lack of a regional registration reply, or the receipt of an authentication failure regional reply prompts the MH to re-synchronize with the hierarchy by sending a new home registration not supplying its regional information while appending a local replay protection extension. The receipt of an identification mismatch regional reply provides the MH with a new identification value that it uses in sending a new home registration request to ensure the execution of the hierarchy consistency mechanism by the crossover FA.

3.4.3 Analysis and Comparison

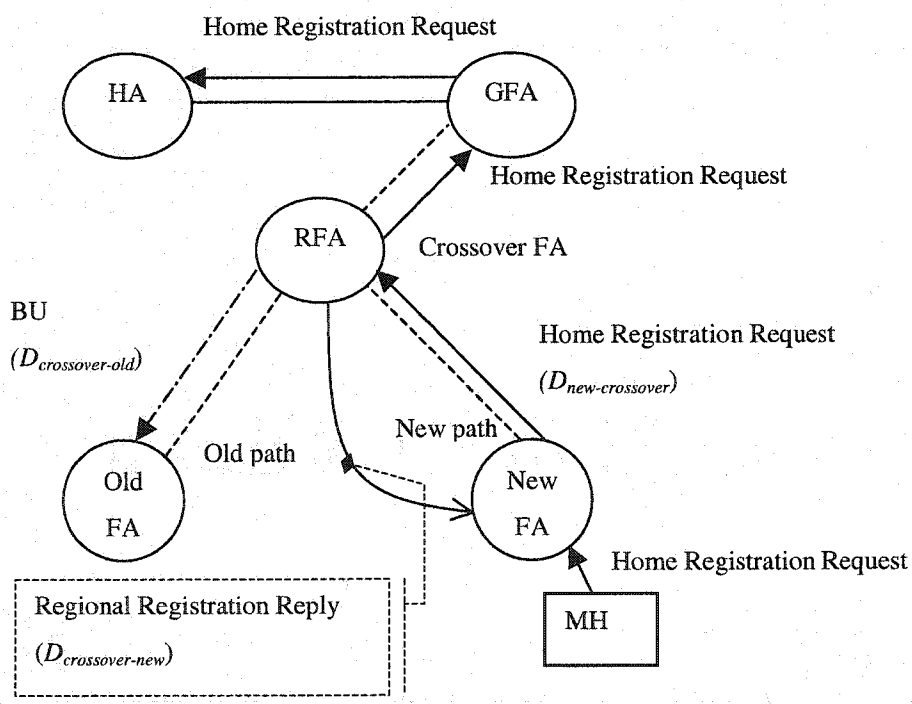
We analyze and compare the KOPA approach, the SINP approach, and MIP_RR [30] while processing home registrations involving local handoffs. We reuse the delay measures introduced in TABLE 1 in section 3.1, and in TABLE 2 in section 3.3.4. KOPA and SINP rely on a "crossover FA"-initiated approach, similar to the proposed regional

registration framework (section 3.3), to perform the tunneling consistency mechanism. Hence, the previous analysis and comparison presented in section 3.3.4 hold true for the following delay measures: $D_{InitConsist}$, $D_{Consistent}$, and $D_{Forward}$. In summary, in our proposed approaches (KOPA and SINP), the delay to initialize the tunneling consistency mechanism, for it to run to completion, and the delay before forwarding of data packets from the old FA to the new FA would be dependent on various processing and intermediate link delays between the new FA and the crossover FA on the new path, and between the crossover FA and the old FA on the old path. Whereas, with MIP_RR such delays are dependent on various processing and intermediate link delays between the new and old FA, and the old and crossover FA on the old path.

The MH's perceived home registration latency is unaffected by the different approaches, since it is defined to be the time interval between submitting a home registration request and receiving a home registration reply. However, the *network-layer handoff completion* within the foreign agent hierarchy varies among the investigated approaches. Recall that KOPA, and MIP_RR use a similar approach that does not switch the MH's tunneling path to the new path until a home registration reply is received, whereas, SINP emphasizes the local handoff aspect, switching the MH tunneling path to the new path as soon as possible without waiting for a home registration reply. Hence, the latter approach (SINP) achieves lower handoff latency than the former (KOPA and MIP_RR [30]). Specifically, in SINP the network-layer handoff latency (starting when the new FA receives the MH's request) is equivalent to the case of processing a regional registration by the crossover FA. The crossover FA is expected to be much nearer (in the network sense) to the new FA than a possibly distant HA. Fig. 19 illustrates an abstracted view of home registration processing along with involved delays in the KOPA and SINP approaches.



(a)



(b)

Fig. 19. Abstracted view of home registration (HR-LH) processing with involved delays. (a) KOPA approach. (b) SINP approach.

We define $D_{crossover\text{-}HA}$ to be the delay involved for a home registration request propagated by the crossover FA to actually reach the HA, starting when the crossover FA receives the MH's request (the registration request eventually reaches the GFA which sends it to the HA). Similarly, $D_{HA\text{-}crossover}$ is the delay involved for the HA to process the home registration request, generate a home registration reply sending it to the GFA, that propagates it within the hierarchy to eventually reach the crossover FA. In general, the MH's perceived home registration latency is given by (5), assuming the time for the crossover FA to process the home registration reply and propagate it within the hierarchy until it reaches the new FA is equivalent to $D_{crossover\text{-}new}$.

The network layer handoff latency for KOPA and MIP_RR is the same and is given by (6), while for SINP it can be calculated as in (7). The KOPA approach and MIP_RR are similar in completing the handoff from the old path to the new path only when receiving a home registration reply, whereas the SINP approach is expected to have a lower handoff latency measure since the handoff processing is localized within the hierarchy itself, without relying on a registration reply by the HA to set up the new path.

$$\begin{aligned}\text{Registration latency} = {} & \text{Time for the new FA to receive the request from the MH} + \\ & D_{new\text{-}crossover} + D_{crossover\text{-}HA} + \\ & D_{HA\text{-}crossover} + D_{crossover\text{-}new} + \\ & \text{Time for the MH to receive the reply from the new FA} \end{aligned} \quad (5)$$

$$\text{KOPA and MIP\_RR network layer handoff latency} = D_{new\text{-}crossover} + D_{crossover\text{-}HA} + D_{HA\text{-}crossover} + D_{crossover\text{-}new} \quad (6)$$

$$\text{SINP network layer handoff latency} = D_{new\text{-}crossover} + D_{crossover\text{-}new} \quad (7)$$

A consequence of waiting for the home registration reply to set up the new path in KOPA and MIP_RR is that the old path, and the old FA are relied upon to deliver any data packets that reach the crossover FA for a period of time. Hence, during the time interval it takes for the registration reply to reach the crossover FA ($D_{crossover\text{-}HA} + D_{HA\text{-}crossover}$), the crossover FA forwards any data packets through the old path. In general, after the old FA is informed about the new FA, data packets latency is augmented by the delay it takes for a packet to be processed, and sent by the old FA, and later received by the new FA. Using the SINP approach, the old path is used until the home registration request reaches the crossover FA, at which point any subsequent data packets follow the new path. Hence, the time period where the old path is still used by the crossover FA is reduced compared to KOPA and MIP_RR. Data packets latency is an important factor in interactive multimedia applications where a playout delay is maintained to overcome network jitter [36]. A data packet playout time is set to be the packet send time + the current playout delay. If a data packet is received after its playout time, it is dropped.

3.5 Performance Evaluation

We evaluate the performance of the proposed approaches through simulations. We implemented the proposed registration processing framework by extending the Columbia IP micro-mobility software (CIMS) [12], [15], which is an *ns*-2 network simulator [40] source code extension implementing a 1-level foreign agent hierarchy below the GFA. CIMS was extended to simulate *n*-level foreign agent hierarchies and model a true foreign domain. We defer the details of our network simulation framework to section V.

We investigate the performance of both TCP and UDP traffic for KOPA, SINP, and Base Mobile IP (MIP) approaches, and attempt to enforce an adequate number of home registrations involving local handoffs. In our approaches, the MH performs an initial home registration when it enters the foreign domain. Afterwards, it performs regional registrations until it renews its home mobility binding by issuing another home registration. Regional registrations are processed as presented in section 3.3, while home registrations involving local handoffs are processed according to KOPA (section 3.4.1) or SINP (section 3.4.2). In base MIP, the MH initiates a home registration request whenever it receives an advertisement from a new FA, hence has large registration signaling

overhead, and large handoff latency. For comparison, we include a "naïve" HR-LH processing approach where the old path is cleared by the crossover FA upon receiving the home registration propagated through the new path, hence treating HR-LH similar to regional registrations. We term this "naïve" approach the *Delete Old Path* (DOP) approach. Observing the performance of the DOP approach gives an insight of the benefits achieved by the KOPA and SINP approaches. We assume that the MH is not using the smooth handoff approach, and hence the old FA is only informed about the new FA in the case of the KOPA approach (see section 3.4.1.3).

Fig. 20 depicts the simulated network topology. The foreign domain is comprised of a single FA hierarchy. The FA hierarchy is a perfect 4-level binary tree. We added the capability of simulating foreign agent hierarchies modeled as perfect *N-ary trees* with an arbitrary number of levels. We use the notation $FA_{i,j}$ to denote the foreign agent number j in level i of the tree. Leaf foreign agents, $FA_{4,j}$ for $j = 1 \rightarrow 8$, provide wireless access to the MH by acting as base stations (BS), whereas other foreign agents in the hierarchy do not possess such capability. Neighboring base stations' coverage areas have an overlap region of 30 meters. We use *ns*-2 implementation of a wireless medium access layer (implementing IEEE 802.11 distributed coordination function which is based on CSMA/CA[12] [16]) for wireless connectivity between the MH and leaf foreign agents.

Each FA is only connected to its children foreign agents through individual 100 Mbps duplex links and link delay $LD_{FA\text{-}FA}$ in milliseconds (unless otherwise stated, default link delay is 0.5 ms). The GFA is connected to the MH's HA through a 1.5 Mbps duplex link with delay $LD_{GFA\text{-}HA}$ ms (unless otherwise stated, default link delay is 20 ms). We simulate a single MH within the hierarchy communicating with a fixed *Correspondent Host* (CH). The CH is connected to the HA using a 1.5 Mbps duplex link with 20 ms link delay.

[12] CSMA/CA stands for Carrier Sense Multiple Access with Collision Avoidance.

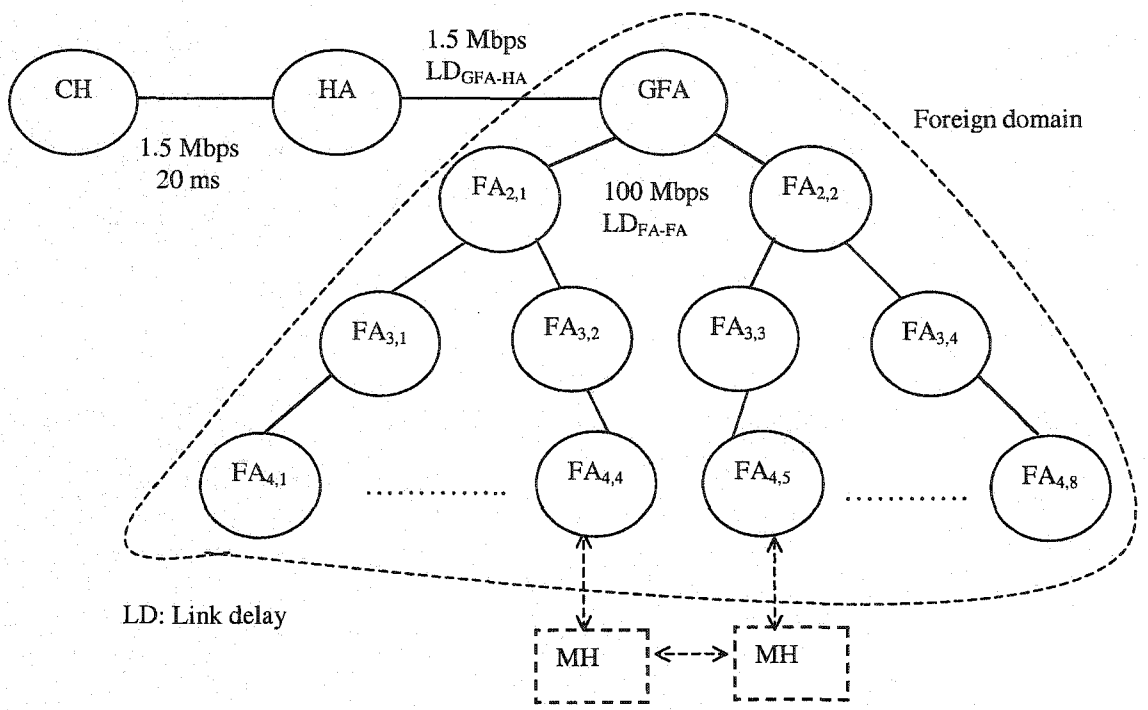


Fig. 20. Simulated network topology.

For the purpose of these simulations, the MH is periodically moving between $FA_{4,4}$ and $FA_{4,5}$, hence periodically performing 3 hop handoffs with the crossover FA being the GFA. We used this approach to have the same number of hops for all handoffs. The home registration lifetime requested by the MH was set to be 30 seconds. After the initial home registration, the MH performs regional registrations until it decides it is time to renew its home mobility binding by sending a home registration request. In general, the home mobility binding renewal procedure is scheduled, when issuing a home registration request, to be executed after a time interval equal to "*registration lifetime - ¼ registration lifetime,*" hence allowing enough time before home registration expiration for the request to make it to the HA. However, in order to obtain simulation results with adequate number of occurrences of HR-LH (home registrations involving local handoffs), the MH performs a home registration, instead of a regional registration; whenever it receives an

agent advertisement from a new FA and the remaining time to issue a home registration request is less than or equal a fraction of the initial registration lifetime. We chose the fraction to be 0.25, i.e., perform home registrations if the remaining time to issue a home registration is less than or equal ¼th the initial registration lifetime. Fig. 21 depicts the timing involved with the home mobility binding renewal process during these simulations.

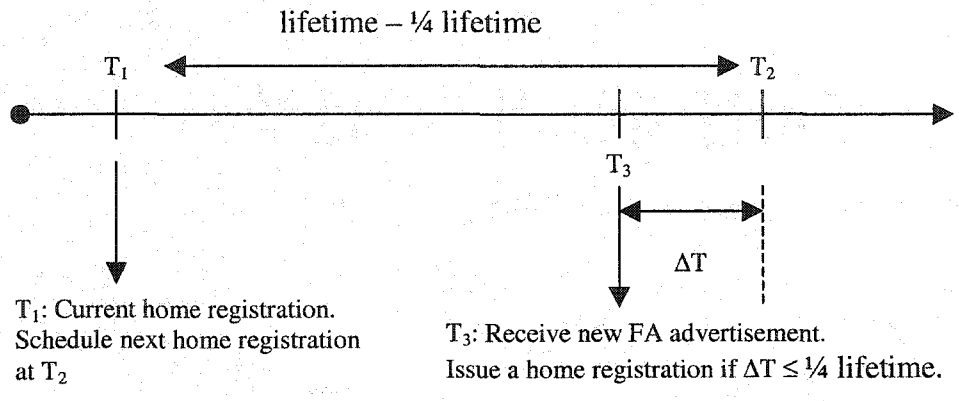


Fig. 21. Timing involved with the home mobility binding renewal process.

3.5.1 UDP Traffic

We simulate the behavior of an audio application by applying constant bit rate UDP traffic from the CH to the MH. A 160-byte data packet is transmitted every 20 ms to simulate a 64 kbps audio stream. The MH is moving periodically between $FA_{4,4}$ and $FA_{4,5}$ at a speed of 10 m/sec until an adequate number of handoffs is attained (more than 100 handoffs). The number of home registrations associated with local handoffs (HR-LH) is different from one simulation run to the other but was bounded between [30-36]% of all handoffs, i.e., [64-70]% of the handoffs involved regional registrations. We do not

impose any other traffic or overhead in the simulation to capture the ideal performance of the proposed approaches. Performance measures include the average number of lost packets per handoff, average number of lost packets per HR-LH, and in the KOPA case, the average number of encapsulated packets from the old FA to the new FA per HR-LH.

3.5.1.1 Effect of $LD_{GFA\text{-}HA}$

We investigate the effect of the delay to the HA ($LD_{GFA\text{-}HA}$) to show the effect of having a distant HA. Fig. 22 illustrates the average lost packets per handoff while varying $LD_{GFA\text{-}HA}$ from 5 to 50 ms and fixing $LD_{FA\text{-}FA}$ at 0.5 ms. The KOPA and SINP approaches outperform both base MIP and DOP, with DOP outperforming base MIP. In general, the number of lost packets per handoff increases linearly with the link delay increase in base MIP and DOP, while it is not affected in KOPA and SINP[13]. For instance, with $LD_{GFA\text{-}HA}$ equal to 50 ms (a distant HA) the average lost packets in base MIP equal 6.8 packets per handoff, whereas in KOPA and SINP it is almost the same equal to 0.26 packets per handoff, representing a reduction in packet loss of almost 96%. In addition, the average lost packets in DOP in this case is 2.4 packets per handoff, representing a reduction in packet loss of 89% for using KOPA or SINP. We note that KOPA and SINP perform similarly in terms of packets lost since the link delays on the old path are the same as on the new path, and KOPA is able to keep up the same performance as SINP, since the number of encapsulated packets from the old FA to the new FA increases linearly with the increase of $LD_{GFA\text{-}HA}$ as illustrated in Fig. 23. Such linear increase is attributed to the increase of $LD_{GFA\text{-}HA}$ while $LD_{FA\text{-}FA}$ is fixed, allowing for longer use of the old path by the crossover FA as the MH's tunneling path. However, we later show that the KOPA approach affects average packet latency due to relying on the old path and tunneling from the old FA to the new FA, for a longer time interval than the SINP approach. The DOP approach, even though "naïve" in terms of HR-LH processing, represents an improvement over base MIP (e.g., 65% packet loss reduction when $LD_{GFA\text{-}HA}$ is 50 ms) since it reduces registration latency through regional registrations, and consequently reduces packet loss.

---

[13] The longer it takes for the registration request to reach the HA, the greater the number of lost packets will be in base MIP and DOP in the absence of any smooth handoff mechanisms.

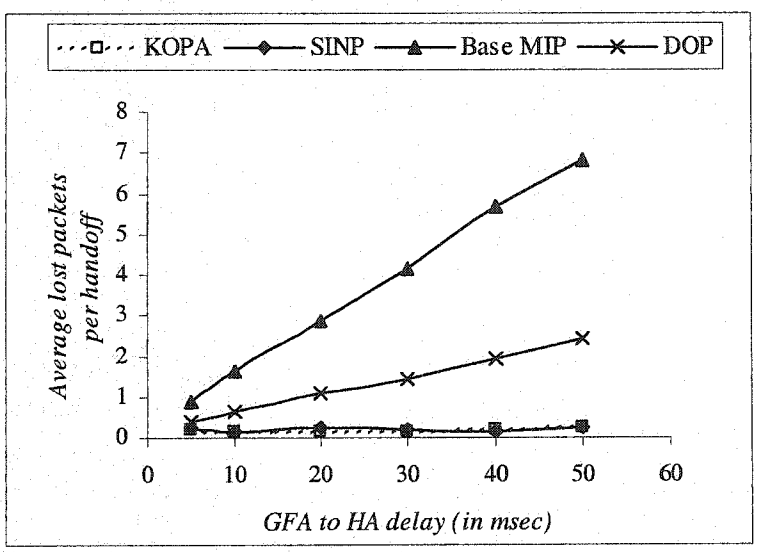


Fig. 22. Average lost packets per handoff versus $LD_{GFA\text{-}HA}$.

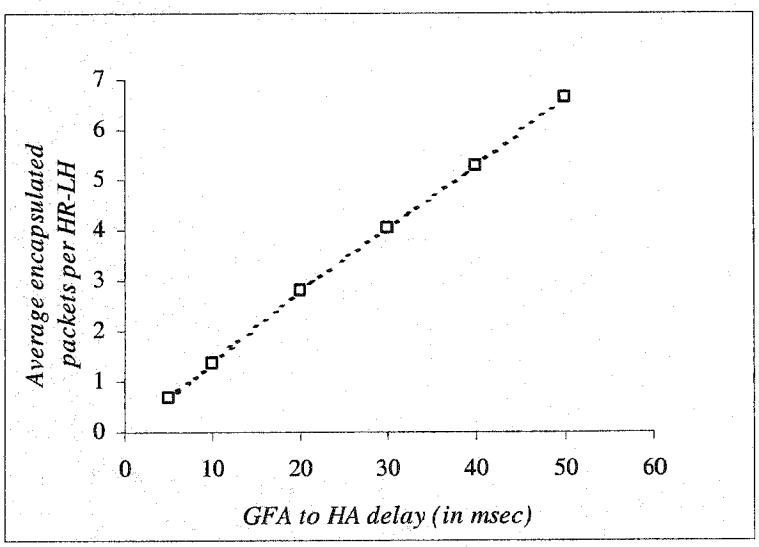


Fig. 23. Average encapsulated packets per HR-LH versus $LD_{GFA\text{-}HA}$ in the KOPA approach.

We demonstrate the reduction in packet loss achieved by introducing special processing for home registrations involving local handoffs. Fig. 24 depicts the average number of lost packets during HR-LH. In DOP, the number of lost packets per handoff increases linearly with the link delay increase, while it is not affected for KOPA and SINP. Comparing Fig. 22 and Fig. 24 in the case of DOP, we conclude that the total number of lost packets during HR-LH represents a very high percentage compared to the total number of lost packets over all handoffs. This ratio is bounded between [75-93]% for $LD_{GFA\text{-}HA}$ between [5-50] ms; hence the necessity of the KOPA and SINP approaches to handle home registration processing during HR-LH.

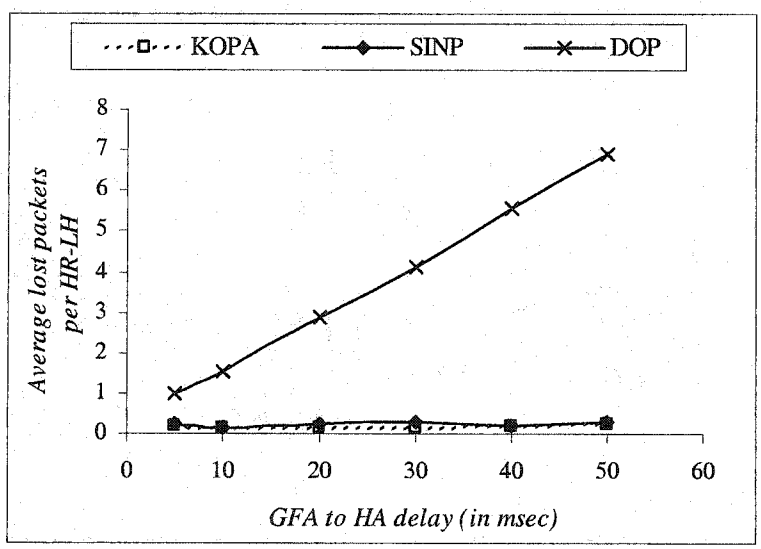


Fig. 24. Average lost packets per HR-LH versus $LD_{GFA\text{-}HA}$.

3.5.1.2 Effect of $LD_{FA\text{-}FA}$

We investigate the effect of hierarchy link delay $LD_{FA\text{-}FA}$ to show the effect of the delay between the new FA and the crossover FA. In this experiment, we exclude the DOP

approach since we previously showed that it is an improvement over base MIP, but does not match the performance achieved while using KOPA and SINP. In our simulated network topology (Fig. 20), with 3 hop handoffs an $LD_{FA\text{-}FA}$ value of d ms implies that the propagation delay for the path from the old FA to the new FA is $6d$ ms[14]. Fig. 25 illustrates the average lost packets per handoff when varying $LD_{FA\text{-}FA}$ between 0.5 and 5 ms while fixing $LD_{GFA\text{-}HA}$ at 20 ms. All approaches exhibit average lost packets per handoff linearly increasing with the increase of $LD_{FA\text{-}FA}$. However, a sizable packet loss reduction can be achieved with KOPA and SINP. For example, for an $LD_{FA\text{-}FA}$ value of 5 ms (the propagation delay between the crossover FA and the new FA is 15 ms), the average lost packets per handoff in base MIP is 4.7 packets per handoff, while in KOPA and SINP it is almost 2 packets per handoff representing a reduction in packet loss of 57%. The increase in propagation delay between the new FA and the crossover FA is the cause for the linear increase in packet loss in KOPA and SINP, since it takes longer for the MH's request to get to the crossover FA, allowing for increased packet loss over the old path in KOPA while the old FA is not informed about the new FA, and in SINP before the switch to the new path is performed by the crossover FA. The SINP approach switches the MH's tunneling path immediately, while KOPA uses the old path for a period of time until the home registration reply is received at the crossover FA. KOPA's use of the old path is evident in Fig. 26 that shows the average number of encapsulated packets per HR-LH in the KOPA approach. We note that the average number of encapsulated packets is almost the same for various link delay values since the propagation delay between the crossover FA (GFA) and the HA is the same throughout the experiment (equal to link delay $LD_{GFA\text{-}HA}$). This results in a similar number of packets that reach the GFA[15], during the time interval it takes the HA to generate the registration reply after receiving the registration request, and are forwarded to the MH using the old path, resulting in a similar average number of encapsulated packets from the old FA to the new FA.

---

[14] The path from the old FA to the new FA consists of 3 hops from the old FA to the GFA, and 3 more hops from the GFA to the new FA.

[15] Packets sent by the CH are intercepted by the HA and tunneled to the GFA.

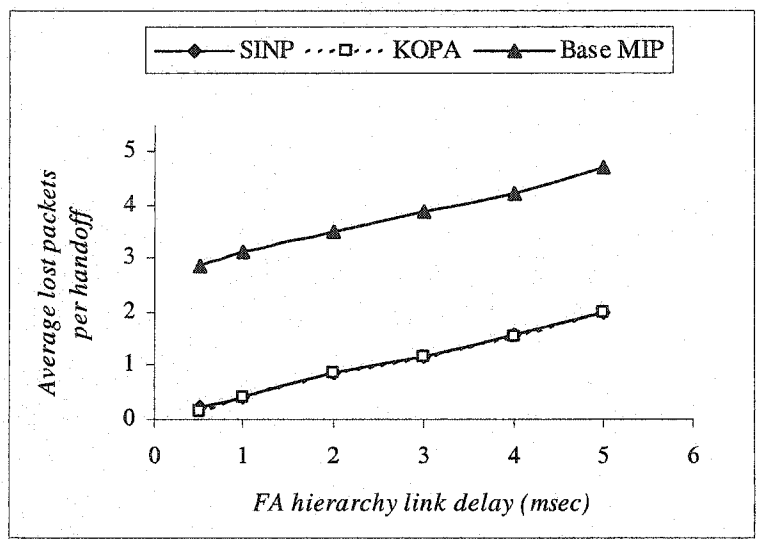


Fig. 25. Average lost packets per handoff versus $LD_{FA\text{-}FA}$.

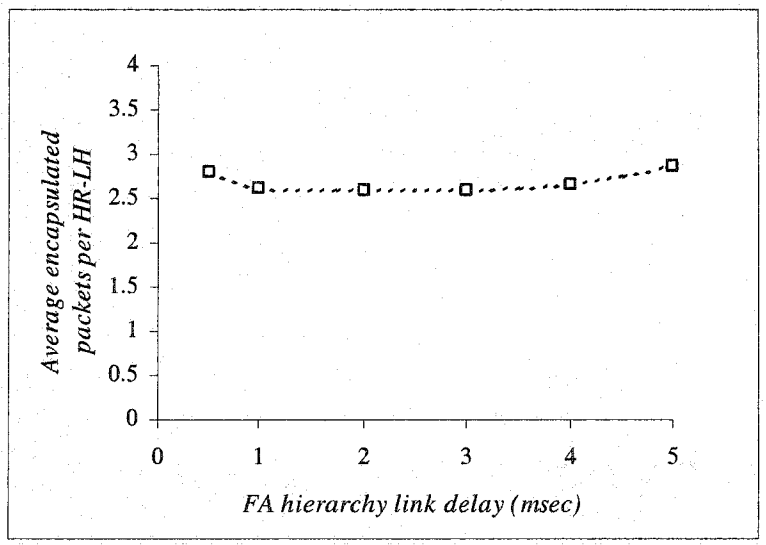


Fig. 26. Average encapsulated packets per HR-LH versus $LD_{FA\text{-}FA}$ in the KOPA approach.

Fig. 27 shows the average number of lost packets per HR-LH. The previously presented discussion applies here that the longer it takes for the registration request to reach the crossover FA, the higher the resulting packet loss. Comparing Fig. 25 and Fig. 27, when $LD_{FA\text{-}FA}$ is 5 ms, the percentage of packets lost during HR-LH compared to the total number of packets lost is 36% and 35% in the KOPA and SINP approaches, respectively. We conclude that the number of hops (number of intermediate RFAs) between the new FA and the crossover FA affect the performance of KOPA and SINP. In general, the usage of the smooth handoff mechanism by the MH and the serving FAs would reduce the observed packet loss, since the new FA (the earliest point of contact in the FA hierarchy) sends a BU message to the old FA on behalf of the MH. We note that our proposed approaches optimize registration processing within the FA hierarchy resulting in packet loss reduction compared to base MIP, and are not meant as a replacement for the smooth handoff mechanism. We later present, in section VI, a study of the effect of using the smooth handoff mechanism along with our approaches using various FA hierarchy network topologies. Another alternative to improve on the observed packet loss is to provide packet-buffering services by the old FA to the MH [49].

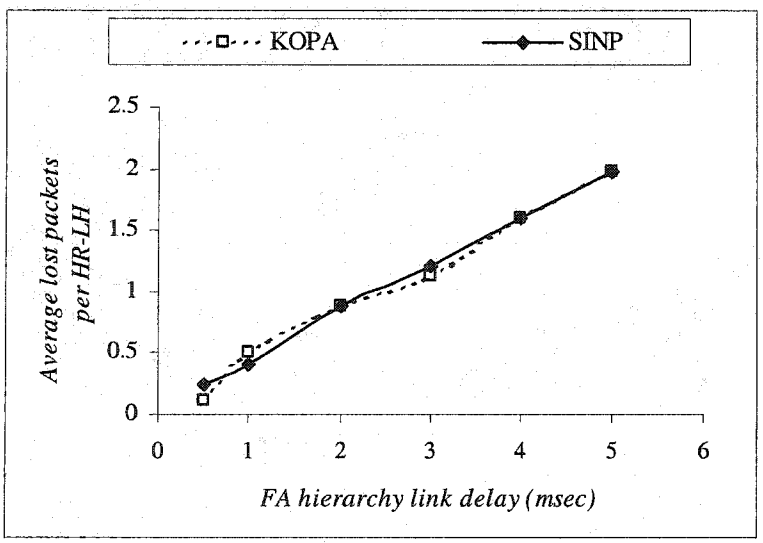


Fig. 27. Average lost packets per HR-LH versus $LD_{FA\text{-}FA}$.

3.5.1.3 Effect of tunneling in KOPA

We investigate the effect of the tunneling approach from the old FA to the new FA in KOPA compared to the immediate switch approach in SINP. The efficiency of such tunneling depends on the following factors.

- *How fast the old FA is informed about the new FA*. This affects the observed number of lost packets if the old FA is not providing packet-buffering services to the MH, or the MH takes too long to find a new FA.
- *The propagation delay between the old FA and new FA*. Tunneled packets experience such delay increasing packet latency, which affects multimedia applications attempting to deal with network jitter.

In addition, the existence of an old path with a tunnel and a new path for data packets can create out of order packets[16]. Similarly, such issues equally affect the efficiency of the smooth handoff mechanism.

We consider the case where $LD_{FA\text{-}FA}$ is 5 ms and observe the received packet order. On the average 1 packet was received out of order during HR-LH in KOPA, while no such behavior was observed with SINP. Moreover, a constant playout delay is maintained and varied while we measure the average number of application-dropped packets per handoff in KOPA and SINP. Note that this packet drop is different than packet loss experienced in the network due to mobility. A packet playout time is calculated to be the packet send time + playout delay. If a packet is received after its playout time it is dropped. Fig. 28 illustrates the average number of application-dropped packets per handoff versus the playout delay in ms. Recall that the propagation delay for a packet from the CH to a leaf FA is 55 ms[17]. We note that zero dropped packets can be achieved in SINP by maintaining a smaller playout delay (around 60 ms) than KOPA (around 90 ms). In our network topology, the old path is symmetric to the new path, and hence the added latency in KOPA is due to the tunneling process from the old FA to the new FA during HR-LH, hence the necessity for a higher playout delay.

---

[16] An older packet is received through the old path after a newer packet has already been received through the new path.

[17] 20 ms link delay from CH to HA + 20 ms $LD_{GFA\text{-}HA}$ + 15 ms over hierarchy links to a leaf FA.

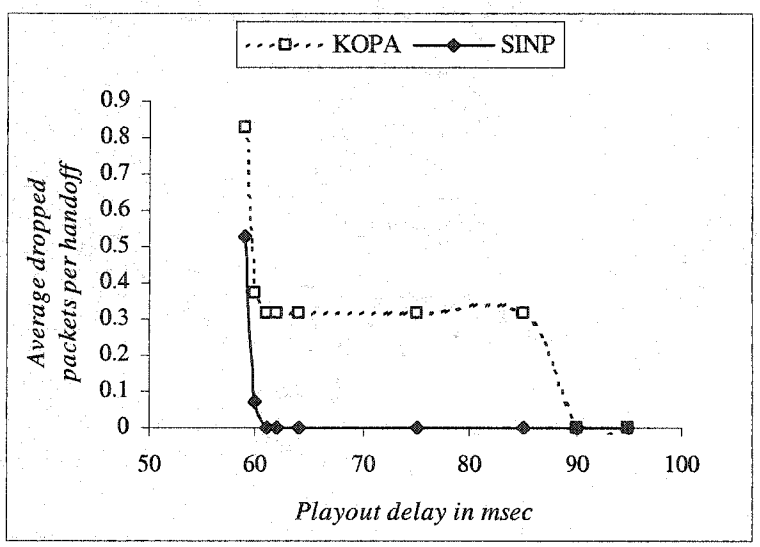


Fig. 28. Average number of dropped packets per handoff versus playout delay.

3.5.2 TCP Traffic

We simulate the behavior of a long-term FTP session between the MH and the CH where the MH is downloading a very large file from the CH. The MH is periodically moving between $FA_{4,4}$ and $FA_{4,5}$ at a speed of 20 m/s until an adequate number of handoffs is attained (> 100 handoffs). We vary $LD_{GFA\text{-}HA}$ between 5 and 50 ms and measure the observed application-level TCP throughput. No constraints are placed on buffering capabilities at individual links, hence no packet drops occur due to buffer overflows. TCP Tahoe [60] was used for the purpose of this simulation.

Fig. 29 illustrates TCP throughput in Mbps versus $LD_{GFA\text{-}HA}$. As expected, the application-perceived TCP throughput degrades with the link delay increase due to increased round trip times. We note that the throughput is bounded by 1.5 Mbps since the bottleneck bandwidth is 1.5 Mbps represented by the link between GFA and HA or between HA and CH (recall that hierarchy links have a bandwidth of 100 Mbps). Base MIP exhibits the worst throughput degradation since the throughput at an $LD_{GFA\text{-}HA}$ value

of 50 ms represents a drop of 30% compared to an $LD_{GFA\text{-}HA}$ value of 5ms. KOPA and SINP demonstrate similar behavior with the throughput degradation at link delay of 50 ms representing a drop of 9% and %10 respectively compared to a 5 ms link delay. At smaller link delays, e.g., 5 ms, all investigated approaches achieve a comparable throughput value. With the link delay increase, KOPA and SINP attain higher throughput than base MIP and DOP, e.g., at a link delay of 50 ms KOPA represents a throughput increase of 34% and 8% over base MIP and DOP, respectively. Such higher sustained throughput is exhibited due to the reduction in packet loss, consequently requiring fewer number of TCP retransmissions, which translates into higher TCP throughput.

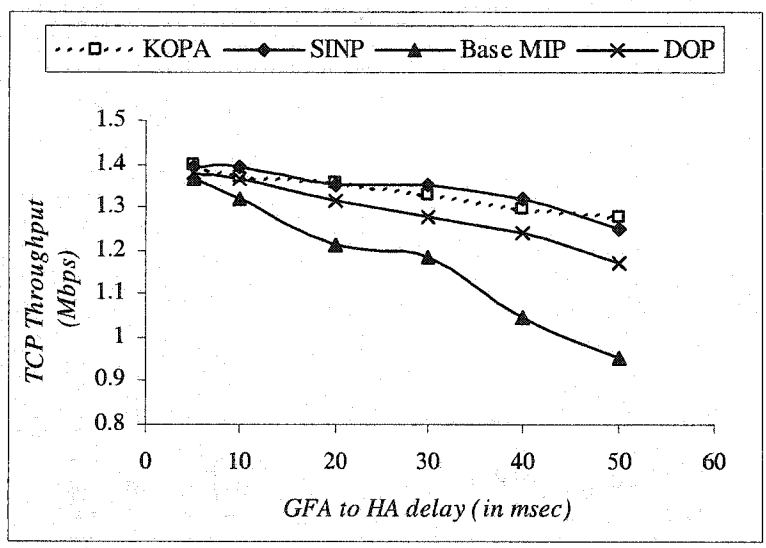


Fig. 29. TCP throughput versus $LD_{GFA\text{-}HA}$.

We investigate TCP's retransmission behavior during the simulation by measuring the retransmission ratio, calculated as the number of retransmitted packets to the total number of transmitted packets (Fig. 30). As expected, Base MIP is able to keep up with the increased number of lost packets by increasing the number of retransmitted packets,

hence leading to an overall lower throughput value. DOP achieves a lower retransmission ratio than base MIP, but is worse than KOPA and SINP for higher link delays. It is worth noting that, during the TCP traffic simulations, we sometimes observed batches of lost packets due to packet discards at the MAC layer in the serving base station. In the IEEE 802.11 MAC implementation, in order to send a data packet, the sending MAC agent transmits an RTS (Request to Send) packet, which should be replied to by a CTS (Clear To Send) packet. Upon receiving the CTS packet, the sending MAC agent is allowed to send a data packet. If a CTS packet is not received for a number of transmitted RTS packets, the designated data packet is discarded. Further probing revealed that the RTS packets are being dropped at the receiving node because of low packet energy level upon reception compared to a receive threshold, implying that the MH is moving out of range from its serving BS. We argue that this is due to the poor MH's "serving BS" selection process currently implemented in *ns*-2, which does not account for example for the distance between the MH and the serving BS [67]. This problem was not observed with UDP simulations, because of the low frequency of packets exchanged between the serving BS and the MH compared to the TCP case.

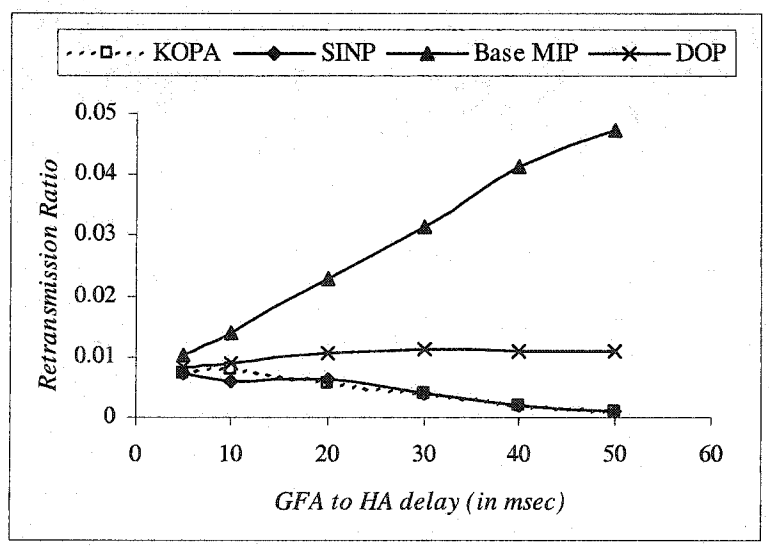


Fig. 30. TCP Retransmission ratio versus $LD_{GFA\text{-}HA}$.

We demonstrate the effect of increasing the handoff rate on TCP throughput. We consider 3 speeds for the MH: 5, 10, and 20 m/s corresponding to average handoff rates of 5, 10, and 15 handoffs per minute. In addition, we select 2 values for $LD_{GFA\text{-}HFA}$: 5 and 50 ms and compare base MIP, KOPA, and SINP. The objective is to investigate the effect of the handoff rate with either a nearby or a distant HA. Fig. 31 depicts the TCP throughput versus the MH speed (handoff rate). Solid lines represent a link delay of 5 ms (nearby HA), while the dashed lines represent a link delay of 50 ms (distant HA). For a nearby HA, the throughput drop with the handoff rate increase is not that significant for all approaches. For a distant HA, base MIP exhibits the worst throughput drop with the handoff rate increase attributed to an increase in the number of lost packets (at 20 m/s a throughput drop of 20% is reported compared to 5 m/s), whereas KOPA and SINP report a modest throughput drop with the handoff rate increase (at 20 m/s, throughput drops are 3% and 2% for KOPA and SINP, respectively, compared to 5 m/s).

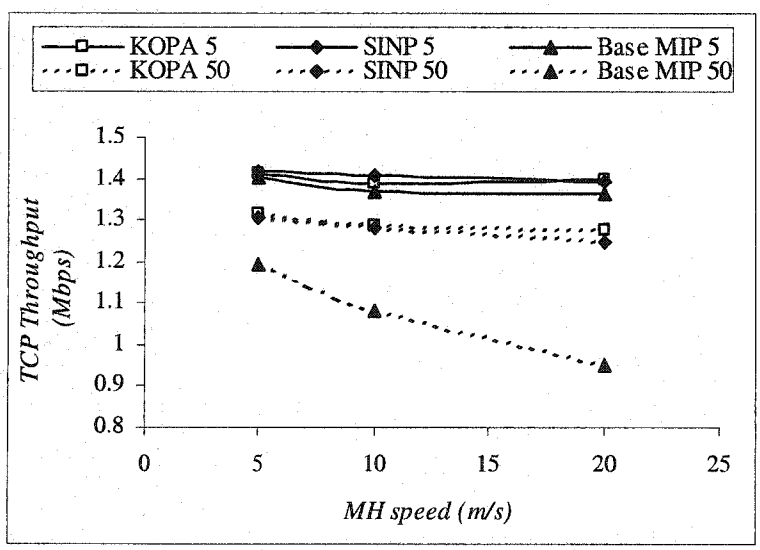


Fig. 31. Effect of handoff rate on TCP throughput.

3.6 Conclusion

In section III, we introduced a framework for home and regional registration processing for intra-hierarchy handoffs, when mobility is supported through the deployment of foreign agent hierarchies in the foreign domain. First, we identified race conditions and drawbacks within the regional registration processing framework for Mobile IPv4 [30], e.g., complete dependence on the smooth handoff mechanism being supported by the MH and the FA hierarchy, and the possibility of introducing inconsistencies in the registration lifetime and erroneous removal of visitor entries for a MH within the FA hierarchy. Such issues prompted us to introduce our registration processing mechanisms for home and regional registrations, which adopt a different signaling design methodology, and prevent the aforementioned race conditions. In general, our mechanisms rely on a "crossover FA"-initiated robust tunneling consistency mechanism instead of being MH-initiated, and based on the smooth handoff mechanism. We analytically contrasted and compared such different design methodologies highlighting the network and processing delays that affect each. We concluded that our mechanisms are not designed to be a substitute for the smooth handoff mechanism to reduce packet loss, but can be a successful alternative when either the MH or the FA hierarchy does not support such smooth handoff mechanism.

We suggested a foreign agent hierarchy model that provides local-area mobility extensions, albeit including a backward-compatible mode of operation for legacy MHs that are unable to take advantage of such extensions. In addition, we introduced a new identification value update mechanism for replay protection purposes, which ensures future successful registration processing within upper levels of the hierarchy. Furthermore, we introduced 2 novel home registration processing techniques for home registrations associated with local handoffs. The first technique termed "Keep Old Path Alive" (KOPA) relies on keeping the old path active tunneling packets to the MH from the old FA to the new FA, until a registration reply is received from the HA establishing the new path. The second technique termed "Switch Immediately to New Path" (SINP) adopts a proactive approach in switching tunneling to the new path, without waiting for the HA's registration reply, hence emphasizing the regional handoff aspect which should be processed within the hierarchy bounds, and reducing the network-layer handoff

latency within the hierarchy. Moreover, our mechanisms provide the same security measures as Mobile IP by providing authentication and replay protection for all registration messages through a set of suggested Mobile IP message extensions.

We simulated our proposed framework using the network simulator *ns*-2 [40] by extending the Columbia IP micro-mobility software [15]. Simulation results have shown that the proposed registration processing mechanisms achieve substantial improvements, compared to base MIP, in terms of reducing UDP packet loss during handoffs (up to 96%), and achieving higher TCP throughput (up to 34%) in the case of a distant HA, without relying on the availability of a smooth handoff mechanism [47]. Furthermore, results have demonstrated that the KOPA approach increases packet latency due to its reliance on tunneling from the old FA to the new FA. Such packet latency increase possibly affects delay-sensitive applications such as interactive multimedia applications. In addition, such latency increase requires maintaining a higher playout delay to ensure a reduced number of application-dropped packets.

In section IV, we introduce a local-area mobility solution based on cooperative foreign agent hierarchies in the foreign domain. The cooperation feature optimizes registration processing for inter-hierarchy handoffs in order to further reduce packet loss, while the registration processing mechanisms presented herewith are adopted for intra-hierarchy handoffs.

SECTION IV

LOCAL-AREA MOBILITY SUPPORT THROUGH COOPERATING HIERARCHIES OF MOBILE IP FOREIGN AGENTS

Section III introduced Mobile IP registration processing for intra-hierarchy handoffs when local-area mobility is supported through foreign agent hierarchies. In this section, we present a novel local-area mobility support framework based on deploying and operating a number of cooperating *FA Hierarchies* within the foreign domain. FA hierarchies cooperate in a scalable configurable manner to keep the MH, home-registered mobility binding, current. The proposed framework reduces inter-hierarchy handoff latency by isolating the effects of the MH's movement from the HA, hence reducing home registration signaling. If possible, the MH keeps the same its home registered care-of address, even if it moves across multiple FA hierarchies within the same foreign domain. In addition, the format and processing of Mobile IP protocol messages are modified to account for the failure of the MH's home registered care-of address while moving between FA hierarchies. The proposed framework along with introduced Mobile IP protocol modifications maintain the same level of security as the base Mobile IP, by providing message authentication and replay protection of protocol messages. We have evaluated the performance of the proposed framework using our extension of the network simulator *ns*-2 [8], and have demonstrated its effectiveness in reducing UDP packet loss and TCP retransmissions while a MH is moving between FA hierarchies within the foreign domain.

Section IV is organized as follows. Section 4.1 presents the motivation behind the cooperation framework. Section 4.2 introduces the suggested foreign domain architecture along with resulting modifications in the foreign agent advertisement messages. Section 4.3 presents a high-level operational overview of the cooperation framework between FA hierarchies. Section 4.4 presents the adopted registration messages and related processing

details. Section 4.5 details the conducted simulation experiments in order to evaluate the performance of the proposed framework. Finally, section IV is concluded in section 4.6.

4.1 Motivation and Overview

Deploying one FA hierarchy in the foreign domain places a burden on the GFA. The GFA has to maintain a visitor entry for every MH within the foreign domain. This becomes a drawback as mobility becomes the norm, rather than the exception. In addition, one GFA presents a single point of failure in such system. Although the regional registration approach [30] suggests that at least one GFA should be present in a domain, it does not allow cooperation between GFAs to maintain the current MH mobility binding within the domain to further reduce any unnecessary registrations with a possibly distant HA. Nevertheless, it allows the MH to request regional registration with its known GFA, other than the one advertised by the current FA. Such regional registration can fail since the current FA may know nothing about the current MH's GFA, forcing the MH to send a home registration request changing its home registered care-of address to the new GFA, and consequently incurring a possibly large home registration signaling overhead. In addition, this approach requires security associations between FAs and GFAs in different FA hierarchies, which might not be feasible if the FA hierarchies are controlled by different administrative entities within the same domain, or if even feasible increases substantially the required number of security associations. On the other hand, the Anchor FA approach [21] allows any FA to become an Anchor FA, requiring security associations between any two FAs. Management of such security associations can become cumbersome when the number of deployed FAs within a zone increases.

In order to further reduce any home registration signaling overhead while the MH is moving within the same foreign domain, and to reduce the required number of security associations between FAs, we suggest deploying *multiple cooperating FA hierarchies* in the foreign domain. Although, multiple FA hierarchies coexist in the foreign domain, they can cooperate in a configurable and scalable manner to maintain the MH home registered care-of address current, and the same as long as the MH is moving within the same foreign domain. The suggested cooperation is an attempt to deal with the inter-

hierarchy handoff regionally (within the foreign domain) instead of involving a possibly distant HA in processing such local movement.

Scalable cooperation implies requiring the minimum number of security associations to ensure such cooperation, and is achieved in our suggested architecture by allowing cooperation across FA hierarchies only between the roots of each hierarchy (section 4.3). In order to enable scalable inter-hierarchy cooperation, two security associations, one in each direction, are required between each 2 roots of such hierarchies. Recall that, within the same FA hierarchy additional security associations are required between each parent FA and its children FAs (section 3.2). For instance, if 2 hierarchies each with f leaf foreign agents are deployed within the foreign domain, the required number of security associations to enable our suggested inter-hierarchy cooperation is 2 (one in each direction between the 2 GFAs). Whereas, in a non-scalable setting where any FA can be required to forward a registration request to the GFA of the other hierarchy [30], $4*f$ security associations are required (2 security associations between any leaf foreign agent in one hierarchy and the GFA of the other hierarchy, hence $2*f$ associations between one hierarchy and the other GFA). In general, k deployed hierarchies within the foreign domain only require ${}_kP_2$ security associations for scalable inter-hierarchy cooperation between all hierarchies[18].

Configurable cooperation implies the capability to easily configure such cooperation by network administrators. Such task is achieved by extending the FA's advertisement message, to signal whether a GFA will send and/or receive cooperation requests on behalf of this MH.

4.2 Foreign Domain Architecture

The foreign domain is partitioned into *Routing Zones* (Fig. 32). Routing zones are non-overlapping in the sense that each routing zone constitutes an independent FA hierarchy. The ability to partition the foreign domain into FA hierarchies gives great

---

[18] The number of ways of obtaining an *ordered* subset of m elements from a set of n elements is ${}_nP_m$ and is equal to $n!/(n-m)!$

flexibility to network administrators. Each hierarchy can be managed independently from the other, while still not precluding any possible inter-hierarchy cooperation.

Special cases might arise when partitioning the foreign domain into routing zones such as a *domain-wide* routing zone, and *single-FA* routing zone. The domain-wide routing zone case implies deploying one FA hierarchy within the foreign domain. The single-FA routing zone case implies that the routing zone is constituted of only one FA, i.e. the FA hierarchy has been reduced to a single FA. If all the routing zones in the foreign domain are single-FA routing zones, then this foreign domain partitioning maps to the anchor FA strategy of independent FA deployment within the foreign domain [21].

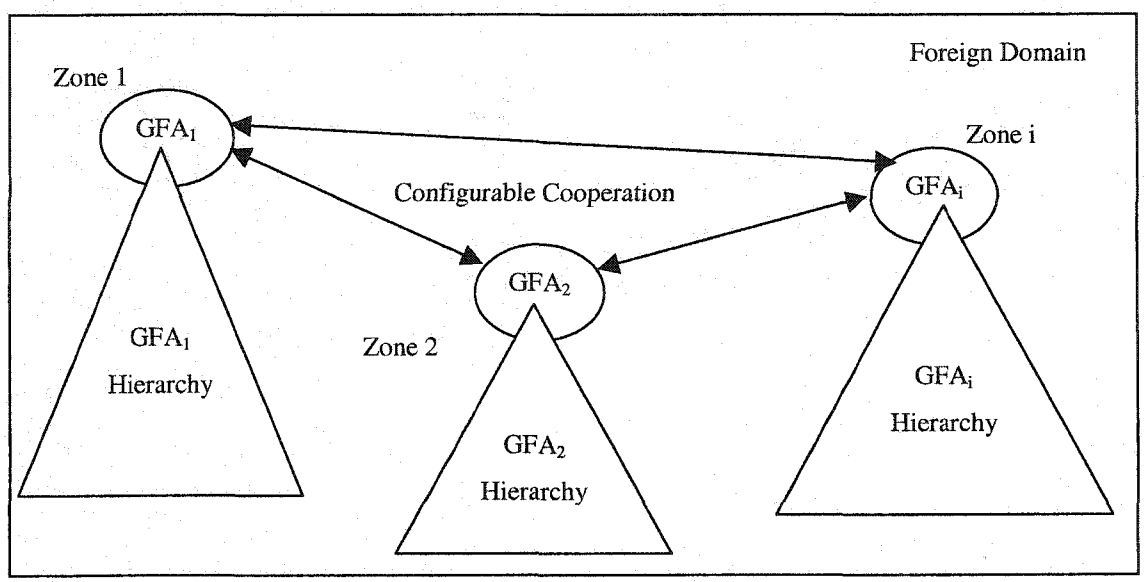


Fig. 32. Routing Zones (FA hierarchies) within the foreign domain.

For an individual FA hierarchy, we adopt the foreign agent hierarchy model introduced in section 3.2. The root FA in a zone acts as a GFA for this foreign domain, and is required to have a publicly routable IP address. Each FA advertises its own IP

address, if not private, as well as its GFA IP address. The GFA IP address can be used by a MH as care-of address when registering with its HA. In this manner, at any point of time, different MHs may register different GFA addresses as their care-of addresses. Hence, mobile hosts routing entries are distributed between GFAs, and consequently, a single GFA does not have to act as the HA tunnel endpoint for all MHs within the foreign domain. In addition, any FA advertises its *Network Access Identifier* (NAI) [3] in its agent advertisement message. This enables the MH to determine if it is in its home domain, or it is now in a visited foreign domain, and whether it has changed domains since its last registration. All FAs in all routing zones have the same realm (domain name) in their NAI.

We introduce the required foreign agent advertisement extensions to enable configurable cooperation between deployed foreign agent hierarchies. The MH can detect that it has changed routing zones, within the same foreign domain, by examining the advertised GFA IP address. GFAs cooperate to maintain the mobility binding of an MH current without having to register with its HA, unless deemed necessary by the MH, or by the current GFA. To make such cooperation configurable and controllable by network administrators, any FA advertises two new flags in its mobility agent advertisement extensions [43] as depicted in TABLE 3. These flags define whether this GFA, the root of the current FA hierarchy, will permit the following.

1. Accept cooperation requests from other GFAs within this foreign domain? (The *P* flag)
2. Send cooperation requests on behalf of the MH to other GFAs within this foreign domain? (The *C* flag)

For instance, if the MH has home registered GFA_1 as its care-of address, and is moving into the GFA_2 hierarchy, cooperation can occur if GFA_2 advertises the possibility of sending cooperation requests on behalf of the MH (set the *C* flag), and if GFA_1 advertises the possibility of accepting cooperation requests from other GFAs (set the *P* flag).

TABLE 3

COOPERATION FLAGS IN THE FOREIGN AGENT ADVERTISEMENT MESSAGE

Flag	Explanation
C	If set, the GFA of this FA hierarchy will send cooperation requests, on behalf of the MH to another GFA within the same domain.
P	If set, the GFA of this FA hierarchy will receive, and acknowledge cooperation requests from other GFAs within the same domain.

4.3 Operational Overview

We present a high-level overview of a cooperation scenario between foreign agent hierarchies for local-area mobility support. Throughout this section, unless otherwise stated, the processing of home and regional registrations from within a foreign agent hierarchy is performed according to the regional and home registration processing frameworks previously introduced in section III (see sections 3.3 and 3.4).

When a MH first enters the foreign domain, it is required to perform a home registration with its HA to establish a home-registered care-of address. The MH is responsible for periodically renewing such home mobility binding. Assume that a MH first enters the foreign domain, and is located within the GFA_i hierarchy. We shall focus hereafter on the case where the MH chooses to home register GFA_i as its care-of address. According to [48], the HA generates a *registration key*, and distributes it to both the MH, and GFA_i. This registration key will be used to authenticate the MH within this foreign domain until it performs another home registration, and another registration key is generated and distributed by the HA. GFA_i in turn distributes this registration key down its own hierarchy to the regional FA that forwarded the home registration request. The MH needs to remember that GFA_i is its home registered care-of address as long as it is within this foreign domain. GFA_i is termed the *Home Registered GFA* (HRGFA) for this MH, and represents the root of the forwarding tree for this MH inside this foreign domain. Such terminology was introduced to differentiate between GFAs, and to distinguish the HRGFA hierarchy from other FA hierarchies in the foreign domain. Note

that, different MHs might have different HRGFAs, according to which GFA is their current home registered care-of address. This remains in effect until the MH decides to perform another home registration while within the same foreign domain or another GFA decides that the MH must perform a home registration. This can be due to, for example, the failure of the current HRGFA (GFA_i). The home registration process from within the GFA_i hierarchy is illustrated in Fig. 33.

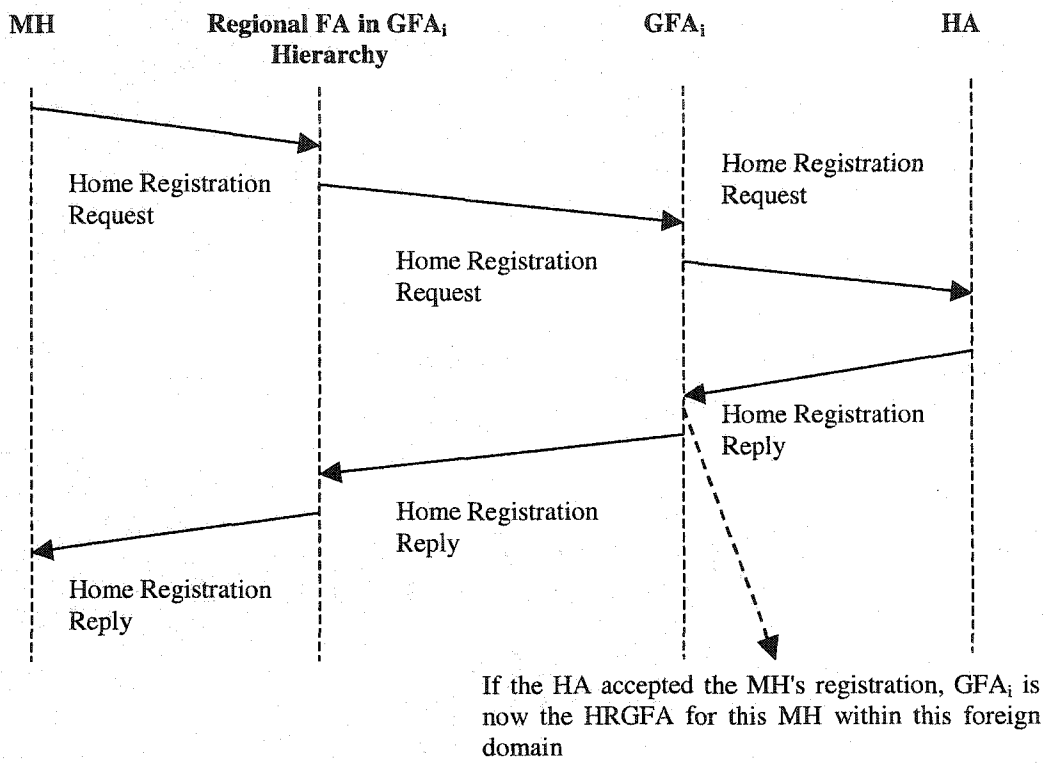


Fig. 33. Home registration process.

When the MH moves to another FA hierarchy within the same foreign domain, e.g. changes location from within the GFA_i hierarchy to within the GFA_j hierarchy, it has two choices available. The first choice is to perform a new home registration changing its home registered care-of address to GFA_j. Alternatively, it can inform GFA_j to cooperate

with GFA_i to maintain its home mobility-binding current if both GFA_i and GFA_j allow such cooperation. This can be envisioned as if GFA_i is dynamically acquiring a new child FA, GFA_j. The MH can base its decision for example on the fact that it is active, sending or receiving datagrams, or currently idle, or based on the cooperation advertisements by both GFAs. If the MH is active, then the obvious choice, to minimize the handoff latency, is to keep his home registered care-of address to be GFA_i, meanwhile GFA_i tunnels any newly received datagrams to GFA_j, which in turn tunnels them down its own FA hierarchy. Later on, if the MH changes location to within the GFA_k hierarchy, the same cooperation process is repeated to establish a tunnel from GFA_i to GFA_k, and the old tunnel from GFA_i to GFA_j is eventually removed (Fig. 34). If the MH is idle, it can choose to inform GFA_j that it needs to perform home registration, to minimize tunneling overhead within the foreign domain.

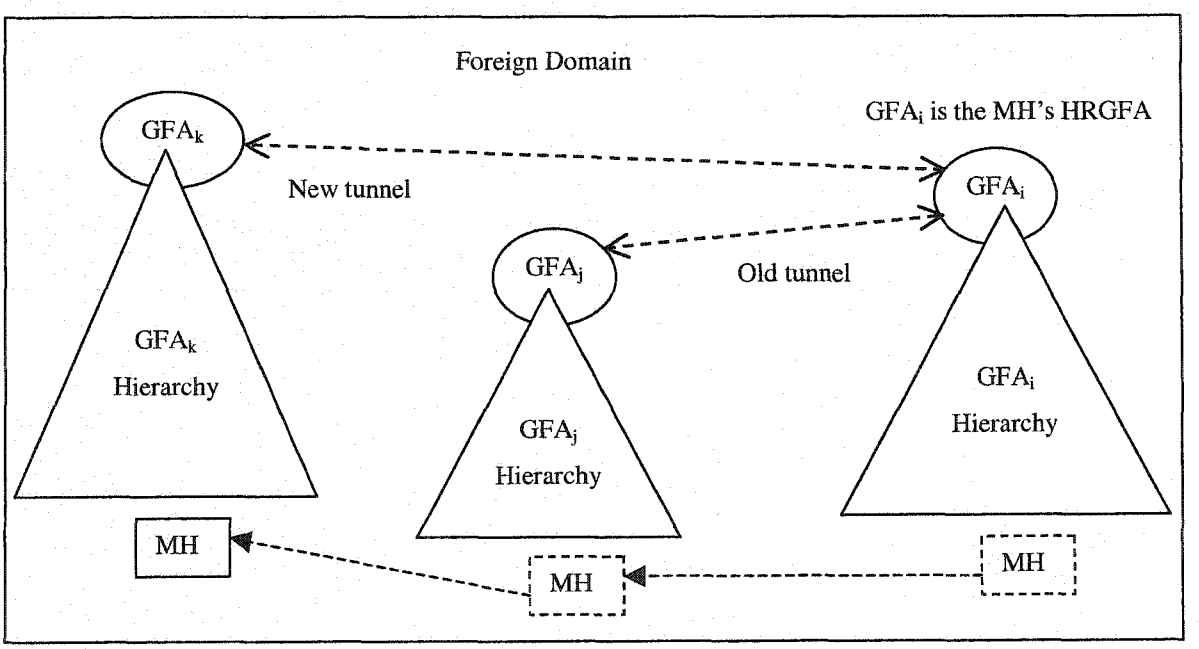


Fig. 34. The MH movement between FA hierarchies within the foreign domain. GFA_i is the MH's HRGFA. The MH moves to within the GFA_j, and GFA_k hierarchies, respectively.

In cooperation mode between GFA_j and GFA_i, GFA_j relays the MH registration to GFA_i. If GFA_i accepts the registration relayed by GFA_j, it sends to GFA_j the registration key acquired from the HA, when the MH first entered the foreign domain. GFA_j in turn distributes this registration key down its own hierarchy. If for some reason, GFA_i failed, GFA_j receives an ICMP [51] error while trying to contact GFA_i, it may go ahead and perform a home registration on behalf of MH. In this case, the MH needs to have included its *Home Credentials* in the registration request to GFA_j. The MH's home credentials are any registration information pertaining to its HA as defined in [43]. If the MH did not include its home credentials, GFA_j returns a registration reply to the MH containing an appropriate error code. The MH upon receiving this registration reply sends another home registration request choosing its care-of address as GFA_j. In this case, further delay and potential packet loss is introduced by the fact that GFA_j sends a registration reply to the MH with an error code, and consequently the MH sending another home registration with either messages having to flow through the current FA hierarchy. Therefore, in this case we suggest formulating the home registration message in a new manner by adding a new Mobile IP extension [43] that carries regional registration information (see section 4.4.3). The differently formulated home registration request represents a combined *home-regional registration* request. The home portion of the registration request serves to establish GFA_j as the new care-of address within the foreign domain, in case the current HRGFA is not reachable. Meanwhile, the regional portion of the request provides the MH's regional contact information (the current HRGFA) for the current GFA. The current GFA, upon receiving the home-regional registration request, attempts to contact the MH's current HRGFA by using the regional registration information. If ICMP errors persist after a number of retries, the current GFA uses the MH's home registration information to perform a home registration on behalf of the MH. In such manner, an attempt is made to account for the failure of the HRGFA, and the MH's home mobility binding is maintained current, while minimizing the incurred delay. The home-regional registration process is illustrated in Fig. 35 and Fig. 36 according to whether the current GFA is successful or not in contacting the HRGA.

As long as the MH is moving within its HRGFA hierarchy, or within another GFA hierarchy for which it had already sent a home-regional registration, the MH can initiate a

regional registration to change its local care-of address within the same FA hierarchy. The regional FA generating the regional registration reply, sends a deregistration message to the old care-of address registered for that MH, unless the MH is requesting simultaneous binding within its registration request. Such deregistration message is forwarded hop by hop down the hierarchy path leading to the MH as part of the hierarchy consistency mechanism. The HRGFA is the FA responsible for generating the deregistration message while responding to a home-regional registration message (see section 3.3).

When the MH returns to its HRGFA hierarchy (GFA_i) after handing off to another hierarchy, it initiates a regional registration targeted to its HRGFA. As an added precaution, the MH's return to the HRGFA hierarchy requires special treatment since previously stored tunneling state in RFAs might not have been properly cleared. Therefore, to avoid any erroneous regional registration replies by intermediate RFAs, the MH signals the need that such registration must only be replied to by the HRGFA to properly establish its new tunneling path within the hierarchy (see section 4.4.2).

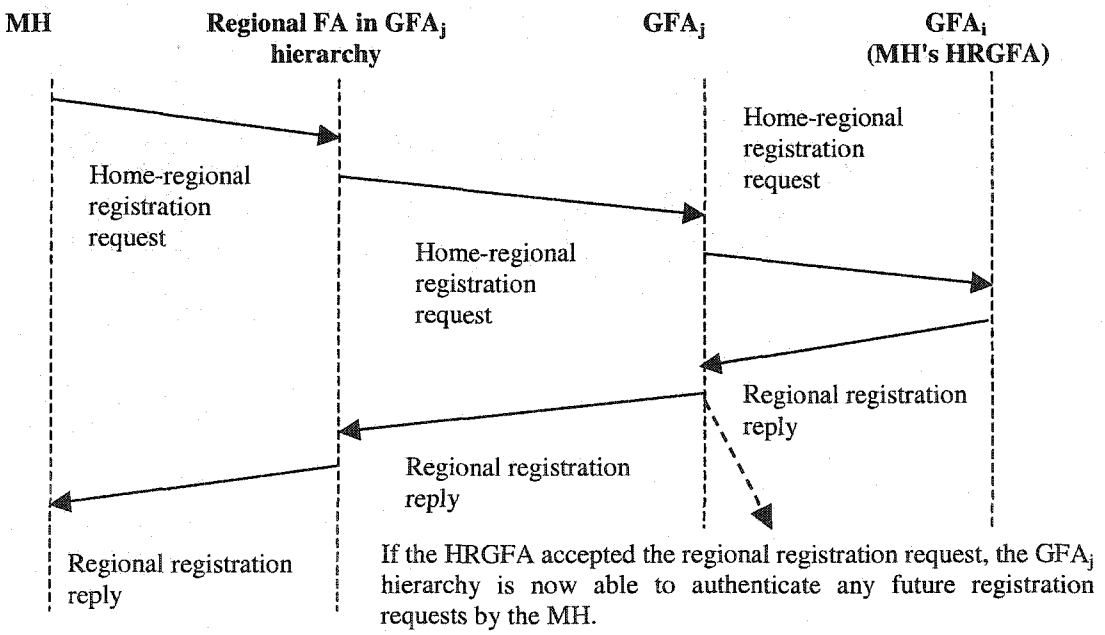


Fig. 35. The home-regional registration process in case the MH's HRGA is reachable.

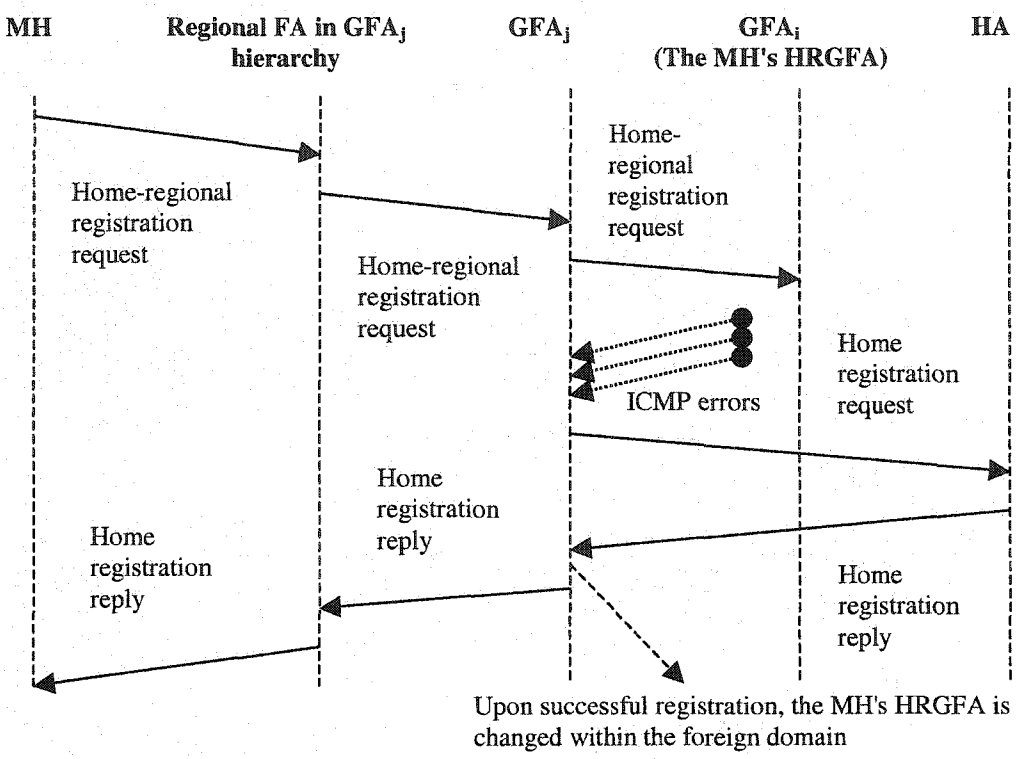


Fig. 36. The home-regional registration process in case the MH's HRGFA is not reachable.

4.4 Registration Messages and Processing

Fig. 37 illustrates a simplified MH's registration state diagram while moving within a foreign domain deploying multiple cooperating foreign agent hierarchies. We assume that the MH chooses to take advantage of the FA hierarchies' cooperation by performing a home-regional registration when detecting a handoff to another hierarchy. The MH can perform either of the following registrations within the foreign domain.

- *Home registration.* Performed when first entering the foreign domain, periodically when renewing the home mobility binding, and anytime the MH

wishes to change its home registered care-of address. After a successful home registration, the MH transitions into the regional registration state.

- *Regional registration.* Performed to change the local care-of address of the MH within the foreign domain. A received FA advertisement from another FA hierarchy prompts the MH to perform a home-regional registration, unless it is moving back to its HRGFA hierarchy.
- *Home-regional registration.* Performed to take advantage of the advertised FA hierarchies' cooperation. After receiving a successful reply for a home-regional registration, the MH switches back to performing regional registrations while moving within the bounds of its current FA hierarchy.

The following sections highlight the processing involved and message extensions, if any, for each of the aforementioned registration types, especially in the presence of multiple FA hierarchies.

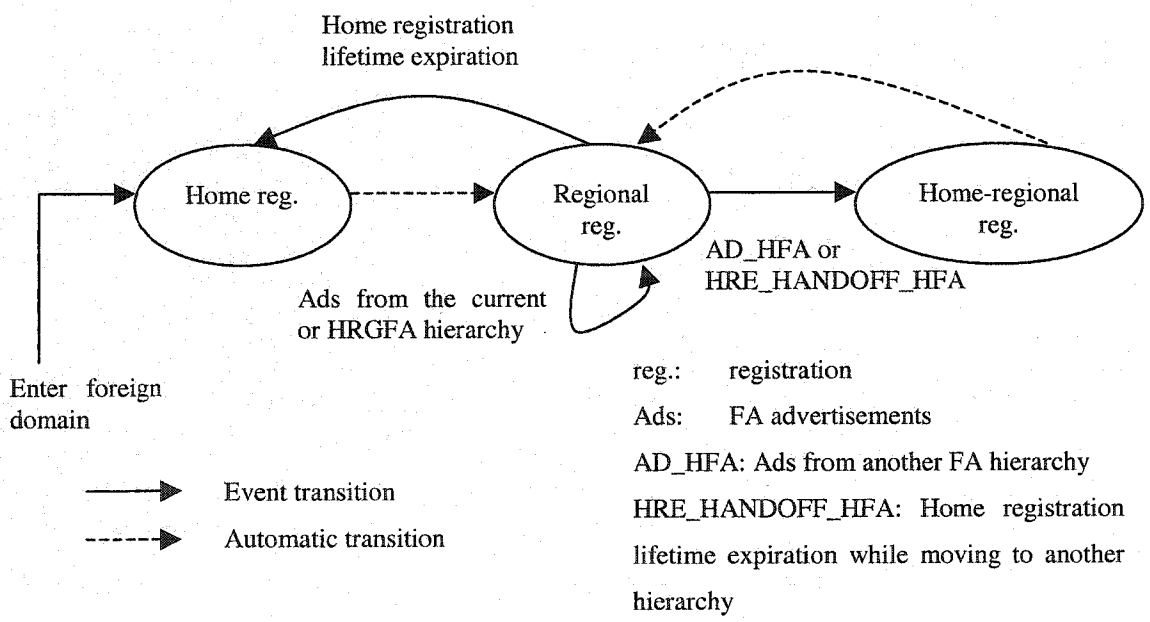


Fig. 37. The MH registration state diagram.

4.4.1 Home Registration

The initiation and processing of home registrations in a FA hierarchy setup was previously introduced in sections II and III. Section 3.4 presented a home registration processing framework for home registrations involving local handoffs. In this section, we highlight any additional processing required to take into account the presence of multiple cooperating FA hierarchies.

In summary, the MH is required to perform a home registration when it first enters the foreign domain, and periodically to renew its home registered care-of address before the home registration lifetime expiration. If the MH is able to acquire a co-located care-of address, it registers this address as its care-of address directly with its HA according to [43]. Alternatively, if registration with the foreign agent is required, the MH home registers the co-located care-of address through the current GFA [30]. Otherwise, The MH formulates a home registration request using as care-of address the GFA IP address, or the advertising FA IP address. The advertising FA upon receiving this registration request inspects the supplied care-of address. If the care-of address is its IP address, then this FA forwards the registration request to the MH's HA and acts according to [43]. Alternatively, if the MH is using the GFA IP address as care-of address, this FA proceeds as previously explained in section 3.1. The MH's home registration is eventually processed by the current GFA and the MH's HA; establishing this GFA as the MH's care-of address.

We note that due to the presence of multiple hierarchies, the MH might be moving within a hierarchy other than its HRGFA's hierarchy. When the MH's home registration lifetime is about to expire, the following cases can be identified.

- *The MH is within its HRGFA hierarchy.* The MH simply sends a home registration to its HA.
- *The MH started moving into another FA hierarchy other than its HRGFA hierarchy.* The MH formulates a home-regional registration request by forming a home registration request addressed to his HA; with care-of address its HRGFA. Next, the MH appends a regional data extension (the HRGFA extension, see section 4.4.3) to this home registration request to provide information about its current HRGFA and records the current GFA as its care-

of address. Intermediate FAs, recognizing that the care-of address in the regional extension is their GFA IP address, forward the registration request upward in the FA hierarchy while adding the Hierarchical FA extension, and the FA-FA Authentication extension. The current GFA inspects and records the information provided by the MH in the regional extension, notes the intermediate FA that relayed this request, and relays the registration request to the HRGFA. Since the care-of address in the original registration information is the HRGFA, the current GFA simply relays the registration request. In such case, the regional extension is provided to enable cooperation between the two FA hierarchies. The HRGFA removes any unnecessary extensions and the regional extension, and records the sending GFA before relaying the request to the MH's HA. If the HRGFA is not reachable, a registration reply with appropriate error code is returned to the MH. In such case, the MH might formulate another home registration request with care-of address the current GFA, changing its HRGFA within the domain to be the current GFA. In such manner, this home registration along with the regional extension serve to renew the home-registered care-of address, and to notify the current HRGFA about the new care-of address for the MH inside the foreign domain.

- *The MH is within a FA hierarchy other than its HRGFA hierarchy.* The MH has already established a mobility binding within the current GFA hierarchy, i.e., it has previously initiated a home-regional registration request. The MH formulates a home registration request addressed to his HA; with care-of address its HRGFA. In such case, the current GFA simply relays the registration request to the HRGFA appending its hierarchical FA extension, authenticating this request with the FA-FA Authentication extension. The regional extension need not be supplied in this case, since cooperation information had been previously exchanged between these two FA hierarchies.

4.4.2 Regional Registration

Regional registrations are initiated and processed as previously presented in section 3.3 where a regional registration processing framework for intra-hierarchy handoffs was introduced. We identify various scenarios when the MH is sending a regional registration.

- *The MH is moving within its HRGFA hierarchy.* The MH generates a regional registration requested targeted to the HRGFA, with the care-of address set to the current FA.
- *The MH is moving within another FA hierarchy.* The MH had already registered its current hierarchy with the HRGFA by sending a home-regional registration request (see section 4.4.3). Afterwards, if the MH needs to handoff to a new FA within this GFA hierarchy, it generates a regional registration request targeted to its HRGFA, with the care-of address set to the advertising FA.
- *The MH is moving back to its HRGFA hierarchy.* When first returning to its HRGFA hierarchy, the MH sends a regional registration request signaling the fact that this registration must only be replied to by the HRGFA. Intermediate RFAs forward such registration upwards in the hierarchy for normal regional registration processing, nevertheless without generating a registration reply even in the presence of a visitor entry for this MH. If by any chance, a visitor entry does exist, it is cleared. In addition, the first RFA that has a visitor entry for this MH (the crossover FA) initiates a tunneling consistency mechanism by sending a deregistration message to the old care-of address. Higher RFAs only clear their visitor entry since the registration message is received from the currently stored tunnel endpoint for the MH[19]. Eventually, the registration message reaches the HRGFA, which in turn generates the registration reply establishing the tunneling path for this MH. The MH indicates the need for such special handling by setting a new flag in the reserved bits as part of the regional registration request.

---

[19] If a visitor entry exists at a crossover FA, when a MH is returning to its HRGFA hierarchy, higher RFAs in the path to the HRGFA most probably would also have visitor entries for this MH.

In normal regional registration processing, the crossover FA generates the regional registration reply that is forwarded down the FA hierarchy until it reaches the MH. In the worst case, the crossover FA is the HRGFA if the MH is within its HRGFA hierarchy, or it is the current GFA if the MH is moving within another FA hierarchy. In all cases, the lifetime field in the regional registration reply is set to the remaining lifetime of the MH home registration. The initial value for this remaining lifetime had been recorded from the home registration reply previously sent by the HA to the HRGFA.

4.4.3 Home-regional Registration

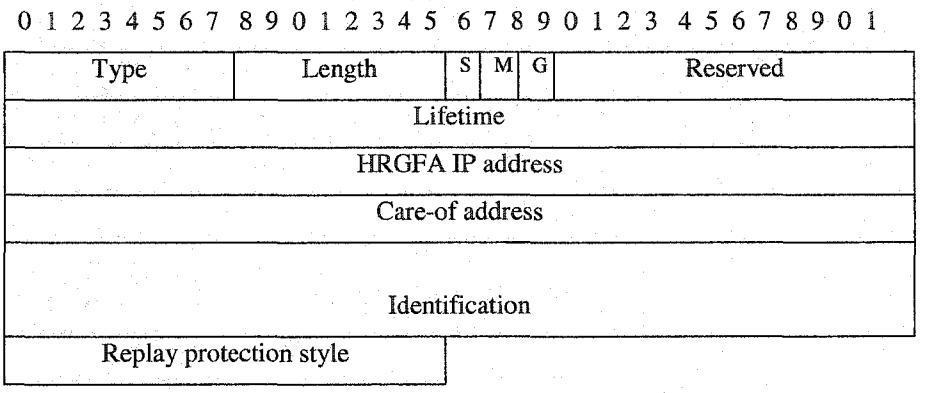
Home-regional registration is performed when the MH discovers that it is changing FA hierarchies within the same foreign domain, i.e. the current FA advertisement contains a GFA IP address different than the MH's HRGFA. Home-regional registration attempts to combine the home and regional registration in one message to minimize any unnecessary delays faced while moving to a new FA hierarchy, in case the HRGFA has failed. The home-regional registration request is basically a home registration request with a mandatory regional data extension.

The current FA is advertising its NAI, with the realm part of the NAI the same for all FA hierarchies within the foreign domain. Consequently, the MH can deduce that it is still within the same foreign domain, and needs to formulate a home-regional registration request, instead of a home registration request.

In order to be able to carry home registration information, along with regional registration information in one message, a *HRGFA extension* is defined to carry the regional registration information (Fig. 38). The HRGFA extension must exist in the home-regional registration message. Information in the HRGFA extension is authenticated by the MH-GFA Authentication extension. The HRGFA extension serves a dual purpose in the home-regional registration request as follows.

- *For the current GFA.* It provides information about the current mobility binding between the MH and the HRGFA such as the HRGFA IP address, the style of replay protection currently in use between the MH and the HRGFA along with the current identification value [30].

– *For the HRGFA*. It provides the current GFA IP address, whether the MH is requesting simultaneous binding, the current identification value to validate this registration request, and the type of encapsulation to be used between the HRGFA and the current GFA if this registration request is accepted.



Type	Extension Type.
Length	Length in bytes of the extension, not including the Type and Length bytes.
S	If set, the MH is requesting simultaneous binding, i.e., the HRGFA retains any prior mobility bindings for this MH to enable fast handoffs [24].
M	If set, the MH requests that HRGFA uses *Minimal encapsulation* [45] for any datagrams tunneled to this GFA. This bit informs the current GFA, that if this registration is accepted, this type of encapsulation will be used by the HRGFA.
G	If set, the MH requests that HRGFA uses *GRE encapsulation* [31] for any datagrams tunneled to this GFA. This bit informs the current GFA, that if this registration is accepted, this type of encapsulation will be used by the HRGFA.
Reserved	Reserved for future use.
Lifetime	The requested lifetime for this registration by the MH.
HRGFA IP address	The IP address of HRGFA.
Care-of address	The current GFA IP address.
Identification	A 64-bit number, used for matching registration requests with registration replies, and for protecting against replay attacks of registration messages. This is the current identification value used between the MH and the HRGFA.
Replay Protection Style	This field identifies how to interpret the Identification field. In addition, it informs the GFA of the current replay protection style used between the HRGFA and the MH. A value of 0 means timestamp protection. A value of 1 means nonce protection.

Fig. 38. The format and data fields of the HRGFA extension.

The Home-Regional registration request formulated by the MH is structured in the following order:

- Home registration information as defined in [43];
- Any non-authenticated extensions relevant to HA appended by the MH;
- The Mobile-Home Authentication extension to authenticate the previous information;
- The HRGFA extension carrying regional registration information;
- Any non-authenticated extensions relevant to HRGFA appended by the MH;
- The MH-GFA authentication extension to authenticate the previous information.

Upon receiving the MH's registration request, the current FA behaves the same as if it is receiving a home registration, and appends the hierarchical FA extension and authenticates the request by the FA-FA Authentication extension. Each intermediate FA performs the same processing, and forwards the registration request upward in the FA hierarchy until it reaches the current GFA. The current GFA identifies this registration request as a home-regional registration request, and begins by processing the regional registration information supplied in the HRGFA extension. The current GFA might append any extensions relevant to HRGFA. Such extensions are authenticated using an FA-FA authentication extension with an authenticator value computed based on the established security association between the current GFA and the HRGFA. Finally, the current GFA forwards the registration request to the HRGFA. The HRGFA validates the registration request, and if successful records that the current regional care-of address for this MH is the forwarding GFA, and returns a regional registration reply back to the sending GFA. The regional registration reply includes the MH's registration key encrypted using the shared security association between the current GFA and the HRGFA. The GFA, in turn, distributes this key down its own hierarchy, until the registration reply reaches the MH. In such manner, this new FA hierarchy is able to authenticate any future regional registration requests received from this MH. Moreover, the HRGFA initiates a hierarchy consistency mechanism by sending a deregistration message to the old care-of address of the MH (see section 3.3).

If the current GFA discovers the failure of the HRGFA by receiving ICMP errors, it may attempt to forward the registration request for a predetermined number of times; afterwards it gives up and switches to performing home registration on behalf of the MH. The current GFA strips the HRGFA extension, and the MH-GFA Authentication extension from the home-regional registration request, and might append any necessary authentication extensions to establish a security association between itself and the HA, and forwards the request to the HA. The HA identifies this request as a home registration request and acts according to [43]. If the HA accepts the registration request, the MH's HRGFA is changed within the foreign domain to be the current GFA. In such case, the current GFA (the new HRGFA) uses the identification and replay protection style fields in the HRGFA extension as the MH's current choice for replay protection style and initial identification value. Alternatively, if the MH supplies a replay protection extension along with the home-regional registration request, the current GFA is required to use the replay protection style and initial identification value from such extension[20].

4.5 Performance Evaluation

We evaluated the performance of our proposed foreign agent hierarchy cooperation framework through network simulations. We extended the Columbia IP Micro-Mobility Software (CIMS) from Columbia University [15], which is a network simulator *ns*-2 [40] source code extension, to model a foreign domain where local-area mobility is supported through a set of cooperating FA hierarchies. The details of our local-area mobility network simulation framework are presented in section V.

Briefly, in cooperation mode, the MH responds to an eminent inter-hierarchy handoff, within the same foreign domain, by issuing a home-regional registration as previously introduced in section 4.3, unless it is returning to its HRGFA hierarchy. In the latter case, it issues a regional registration as detailed in section 4.4.2. In addition, a non-cooperation mode between FA hierarchies was added for comparison purposes, where the MH issues a new home registration upon changing FA hierarchies within the foreign domain. We

[20] The MH might wish to enforce a new identification value (or a new replay protection style) if a change of HRGFA will occur (due to the failure to contact the current HRGFA).

use the notation (C-GFAs) for cooperating gateway foreign agents, while (NC-GFAs) is used for non-cooperating GFAs.

We experiment with UDP and TCP traffic while highlighting the effect of inter-hierarchy handoffs. We do not adopt the agent advertisement handling approach by the MH, previously presented in section 3.5, that produces a significant number of home registrations involving local handoffs, since it is not the issue in focus in our current simulation. Consequently, a MH does not renew its home mobility binding unless its home registration timer is about to expire, which may or may not coincide with a local handoff. Regional registrations are processed according to section 3.3, and 4.4.2, while home-regional registrations are processed according to section 4.4.3. Furthermore, the KOPA approach is used to handle HR-LH registrations, if any (see section 3.4.1).

Fig. 39 illustrates the simulated network topology. The MH is moving within a foreign domain comprised of 2 cooperating FA hierarchies. The 2 FA hierarchies are modeled similarly as perfect 4-level binary trees. In each FA hierarchy, each father FA is connected to its 2 children FAs through a 10 Mbps local-area network (subnet) with 1 ms delay. GFA_1 and GFA_2 are connected through an individual 10 Mbps duplex link with $LD_{GFA\text{-}GFA}$ link delay (default value is 1 ms). Each GFA is connected to the MH's HA through an individual 1.5 Mbps duplex link with $LD_{HA\text{-}GFA}$ link delay (default value is 20 ms). We simulate a single MH within the hierarchy communicating with a fixed correspondent host (CH). The HA is connected to the CH through an individual 1.5 Mbps duplex link with 20 ms delay.

We use the notation $FA_{i,j}$ to denote the foreign agent number j in level i of the tree, in each hierarchy. Leaf foreign agents, $FA_{4,j}$ for j: 1→8, provide wireless access to the MH by acting as base stations (BS), whereas other foreign agents in the hierarchy (regional foreign agents) do not possess such capability. Neighboring base stations' coverage areas have an overlap region of 25 meters. We use *ns*-2 implementation of a wireless medium access layer for wireless connectivity between the MH and leaf foreign agents [26].

For the purpose of these simulations, the MH is periodically moving between $FA_{4,8}$ in hierarchy 1 and $FA_{4,1}$ in hierarchy 2 at a speed of 20 m/s to investigate the effect of inter-hierarchy handoffs. In cooperation mode, such movement pattern results in periodical handoffs where the current HRGFA (GFA_1) acts as the crossover FA due to the resulting

home-regional and regional registrations. Whereas, in non-cooperation mode, a periodical change of the MH's home-registered care-of address is observed due to the resulting home registrations.

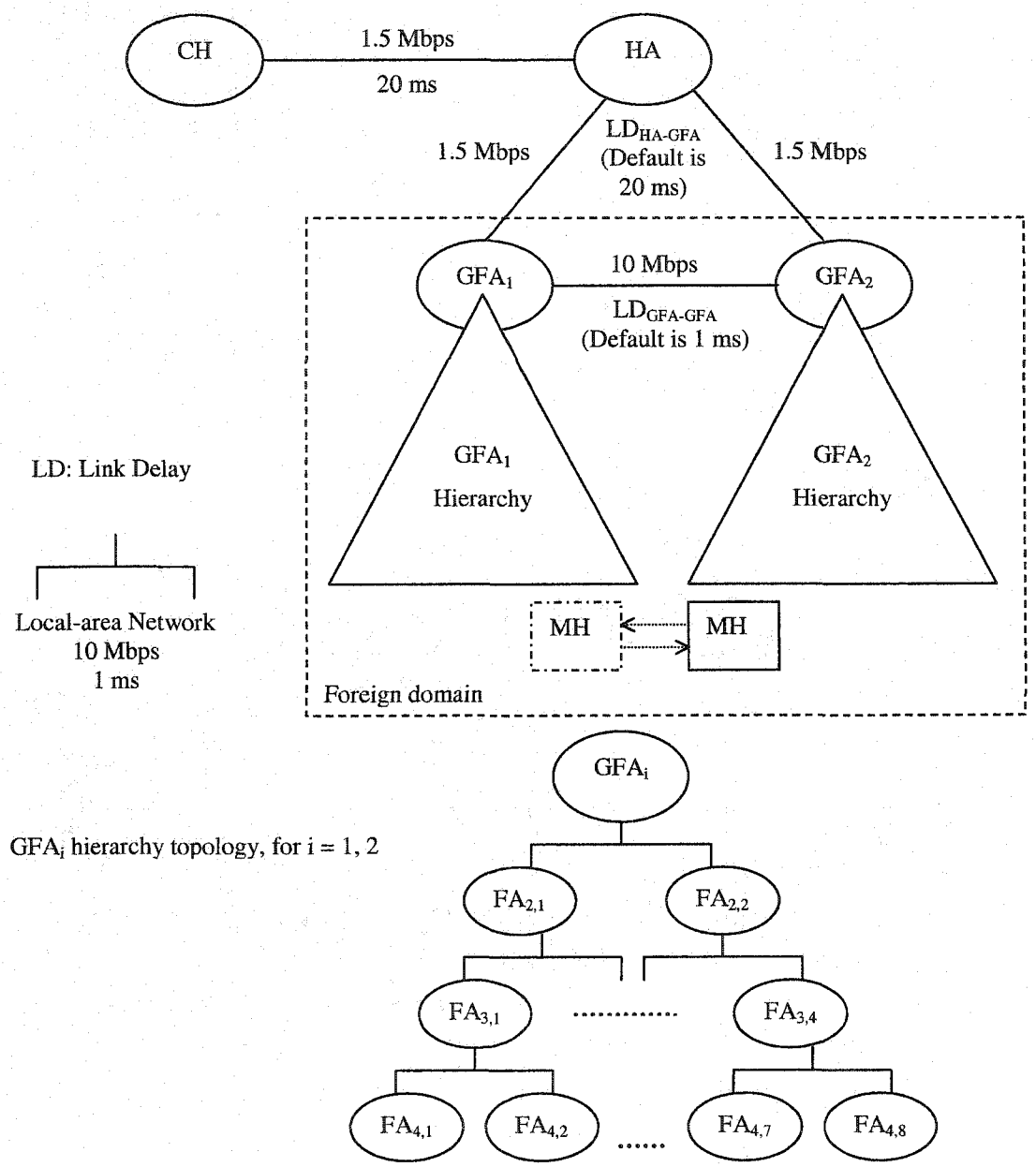


Fig. 39. Simulated network topology.

4.5.1 UDP Traffic

We simulate the behavior of sample audio and video applications by applying constant bit rate UDP traffic from the CH to the MH. For the audio application, a 160-byte data packet is transmitted every 20 ms to simulate a 64 kbps audio stream. For the video application, a 1000-byte data packet is transmitted every 25 ms to simulate a 320 kbps video stream. The MH's periodical movement is performed until an adequate number of handoffs is attained (more than 100 handoffs). We do not impose any other traffic or network overhead to capture the ideal performance of the cooperation framework.

4.5.1.1 Effect of $LD_{HA\text{-}GFA}$

We investigate the effect of increasing the link delay between the HA and either GFAs ($LD_{HA\text{-}GFA}$) from 10 to 50 ms to model a distant HA scenario. We measure the total number of lost packets during the simulation and compute the average number of lost packets per handoff for both audio and video applications. Fig. 40 depicts the average number of lost packets per handoff for cooperating and non-cooperating hierarchies when audio and video applications are separately used. We identify each investigated scenario by appending "/A" or "/V," e.g., "NC-GFAs/A" denotes non-cooperating GFAs results while audio traffic is in effect, and "C-GFA/V" denotes cooperating-GFAs results while video traffic is in effect.

We note that the frequency of data packets in the audio application is higher than the video application, which explains the general trend that the number of lost packets in the audio case is higher than the video case for either investigated approaches. Focusing on the audio application, the average lost packets in the C-GFAs approach are independent of $LD_{HA\text{-}GFA}$, while the NC-GFAs lost packets increase linearly with the delay increase. For instance, at 10 ms link delay, C-GFAs outperforms NC-GFAs by recording a lower number of lost packets (58% reduction in packet loss). Similarly, at 50 ms link delay between the HA and either GFAs, C-GFAs achieves 0.95 lost packets/handoff while NC-GFAs records 7.4 lost packets/handoff, constituting an 87% reduction in packet loss. Such increasing packet loss reduction is the byproduct of converting home registration signaling overhead into a regional registration and benefiting from the network proximity

of the other FA hierarchy. By inspecting video results, we observe a similar pattern of substantial packet loss reduction (a 47% and 90% packet loss reduction is achieved at link delays 10 and 50 ms, respectively)

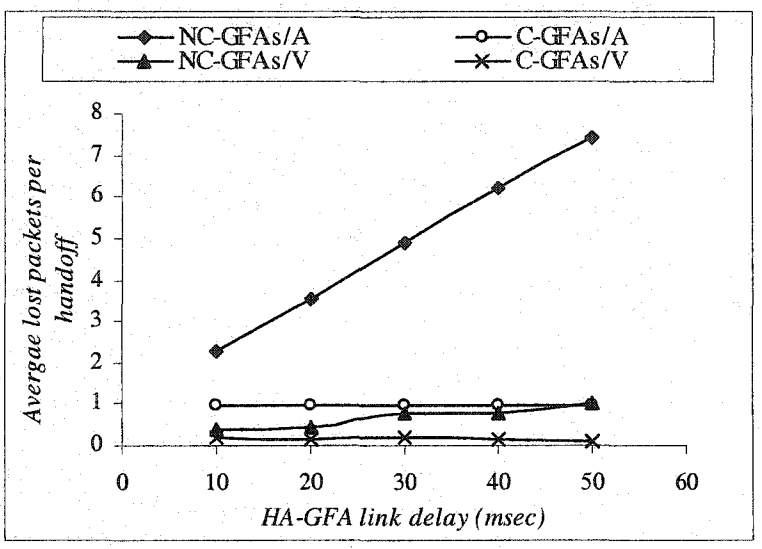


Fig. 40. Average lost packets per handoff versus $LD_{HA\text{-}GFA}$.

4.5.1.2 Effect of $LD_{GFA\text{-}GFA}$

We investigate the effect of increasing the link delay between GFA_1 and GFA_2 ($LD_{GFA\text{-}GFA}$) from 1 to 5 ms, to highlight the effect of the link delay between the HRGFA and the current GFA. We measure the average number of lost packets and average packet latency for an audio application. $LD_{HA\text{-}GFA}$ is fixed at 20 ms.

Fig. 41 illustrates the average number of lost packets while varying the delay between the HRGFA and the current GFA (GFA_1 and GFA_2) for both cooperating and non-cooperating approaches. As expected, NC-GFAs lost packets are not affected with $LD_{GFA\text{-}GFA}$ increase, because of the resulting home registrations that eventually traverse the links

$HA\text{-}GFA_1$ or $HA\text{-}GFA_2$ with similar link delay of 20 ms[21]. On the other hand, as expected, C-GFAs produce lower lost packets per handoff than the NC-GFAs case, that slightly increases linearly with the GFA-GFA link delay increase. For instance, at an $LD_{GFA\text{-}GFA}$ value of 5 ms, NC-GFAs achieves a 66% reduction in packet loss, compared to 72% reduction at 1 ms link delay. Such experiment demonstrates the basic property that the cooperation framework is attempting to exploit: *another GFA hierarchy within the same foreign domain will most probably be nearer (in the network sense) than a distant HA*. Exploiting such property by cooperation between GFAs results in substantial reduction in packet loss for inter-hierarchy handoffs.

Fig. 42 illustrates the effect of increasing $LD_{GFA\text{-}GFA}$ on the average packet latency. For every data packet that eventually reaches the MH, i.e., it is not lost in the network, the packet latency is computed as the difference between the simulation packet receive and send times[22], and the average packet latency is reported at the end of the simulation. We note that NC-GFAs records lower average packet latency than C-GFAs, independent of $LD_{GFA\text{-}GFA}$, whereas with C-GFAs, the average packet latency slightly increases with the link delay increase. Such result is expected since with NC-GFAs, a packet traverses the network path ($CH\text{-}HA\text{-}GFA_i$) and then is tunneled within GFA_i hierarchy until it reaches the MH. Whereas with C-GFAs, on the average half the packets traverse the network path (CH-HA-HRGFA) and are tunneled within the HRGFA hierarchy towards the MH, while the other half of the packets, after reaching the HRGFA, are tunneled to the other GFA, incurring the link delay $LD_{GFA\text{-}GFA}$. Hence, depending on the MH's movement pattern and network topology, higher packet latencies might be expected with the cooperation framework.

---

[21] Recall that both hierarchies have the same topology and subnet delays. Hence, a home registration from within either hierarchy faces the same delays in the absence of any other competing traffic.

[22] The packet send time is stored by the sender in the packet header, while the receive time is noted by the receiver, that in turn computes the packet latency as "receive time – send time".

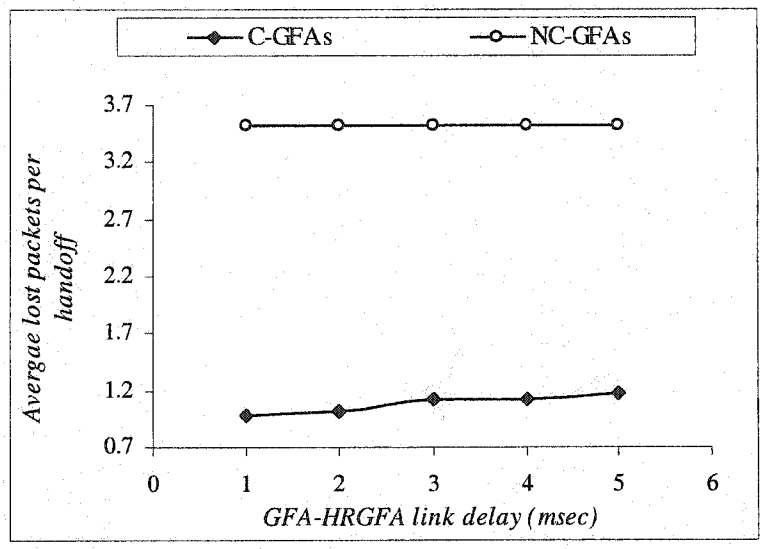


Fig. 41. Average number of lost packets versus $LD_{GFA\text{-}GFA}$.

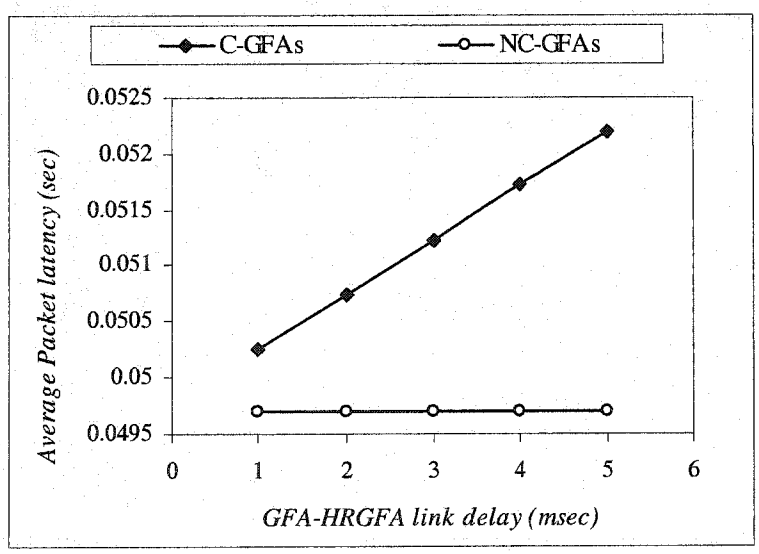


Fig. 42. Average packet latency versus $LD_{GFA\text{-}GFA}$.

4.5.1.3 Effect of the HRGFA being unreachable

We investigate the effect of the HRGFA being unreachable (due to its failure, or link(s) failure between the current GFA and the HRGFA) through simulation as follows. A desired probability of not being able to contact the HRGFA by the current GFA (P_F) is chosen before simulation, and is stored as part of any GFA configuration. A P_F value of 0 implies that the current GFA will always succeed in contacting the HRGFA, while a P_F value of 1 implies that the current GFA will always fail in contacting the HRGFA. When a GFA receives the MH's home-regional registration, it generates a uniformly distributed random number between 0 and 1. If the generated number is greater than or equal to the chosen P_F, the attempt to contact the HRGFA is carried on by the current GFA, otherwise, the current GFA switches to perform a home registration by contacting the HA on behalf of the MH, using the home information supplied as part of the home-regional request as introduced in section 4.4.3. We designed such approach to compensate for the fact that *ns*-2 does not provide an implementation of ICMP [51] to detect the inability to reach the HRGFA. We do not enforce any elapsed period to simulate the time it takes to actually receive back an ICMP error message after the current GFA sends the request. Adding an elapsed period of time to compensate for such delay might increase the observed UDP packet loss, proportional to how long this elapsed time interval would be and the packet generation rate (constant bit-rate traffic is used in this experiment).

Fig. 43 illustrates the average number of lost packets as P_F is increased from 0 to 1, while $LD_{GFA\text{-}GFA}$ is fixed at 1 ms, and $LD_{HA\text{-}GFA}$ is fixed at 20 ms. As expected, the number of lost packets increases with P_F increase, due to the incurred home registration signaling overhead. For instance, average packet loss at P_F value of 1 constitutes more than a three-fold increase compared to a P_F value of 0. Hence, the cooperation framework can deal with the event of the HRGA being unreachable at the expense of unavoidable increased packet loss in the case of active UDP traffic.

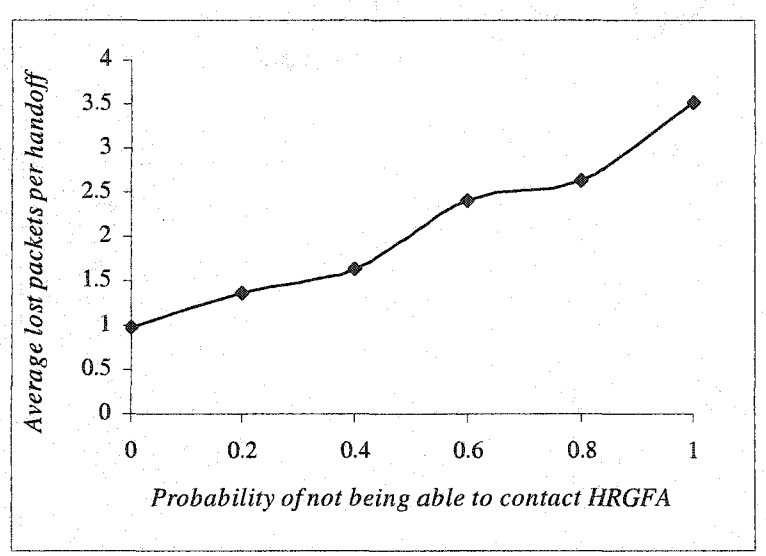


Fig. 43. Average lost packets versus the probability of being unable to contact HRGFA.

4.5.2 TCP Traffic

We simulate the behavior of a long-term FTP session between the MH and the CH where the MH is downloading a very large file from the CH. We vary $LD_{HA\text{-}GFA}$ between 10 and 50 ms and measure the observed application-level TCP throughput. No constraints are placed on buffering capabilities at individual links, hence no packet drops occur due to buffer overflows. TCP Tahoe [60] was used for the purpose of this simulation. $LD_{GFA\text{-}GFA}$ was fixed at 1 ms.

Fig. 44 depicts the application-level TCP throughput while increasing $LD_{HA\text{-}GFA}$ for both C-GFAs and NC-GFAs. As expected, the increase in link delay results in an increase in round trip times, and hence the drop in observed TCP throughput. Note that TCP throughput is constrained by 1.5 Mbps (the bandwidth of the slowest link which in this case is either CH-HA, or HA-$GFA_{1,2}$ link banwidth). The C-GFAs maintain a slight throughput edge over NC-GFAs for large link delays (higher than 20 ms, e.g., at link delay 50 ms, C-GFAs achieves a 5% higher throughput than NC-GFAs).

To investigate the reason why NC-GFAs is able to record comparable throughput values, we measure the retransmission ratio previously introduced in section 3.5.2. Recall that, the retransmission ratio is computed as the ratio of retransmitted packets to the total number of packets. Such ratio highlights what percentage of packets the transport layer had to retransmit to achieve the current throughput value.

Fig. 45 shows the TCP retransmission ratio for the previous experiment (while increasing $LD_{HA\text{-}GFA}$). NC-GFAs are able to record comparable throughput values to C-GFAs by increasing the retransmission ratio to combat the increased frequency of a TCP retransmission timer timeout (Such event occurs because of lack of acknowledgments due to mobility-induced packet loss). For instance, at 50 ms link delay, NC-GFAs records a retransmission ratio of 5% compared to a modest 0.7% for C-GFAs.

We should expect to observe a slight drop in TCP throughput for C-GFAs if we perform the experiment of increasing $LD_{GFA\text{-}GFA}$ while fixing $LD_{HA\text{-}GFA}$ and adopting the same mobility pattern (see section 4.5.1.2 for the corresponding UDP traffic experiment).

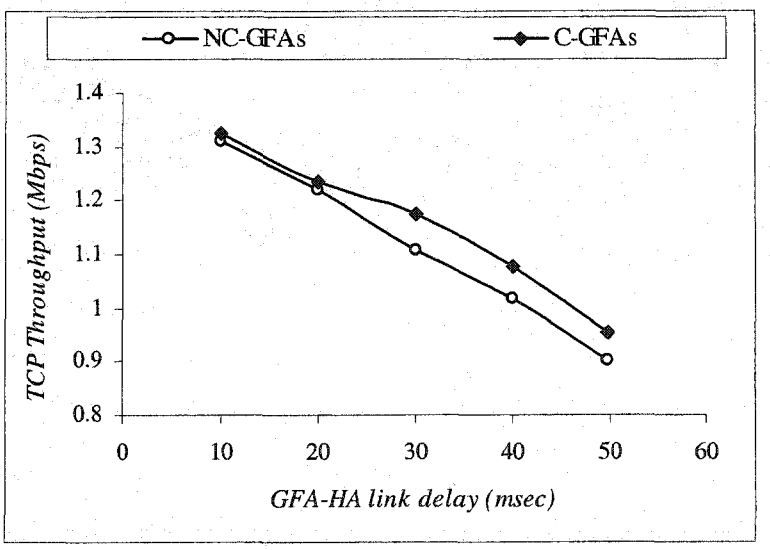


Fig. 44. TCP throughput versus $LD_{HA\text{-}GFA}$.

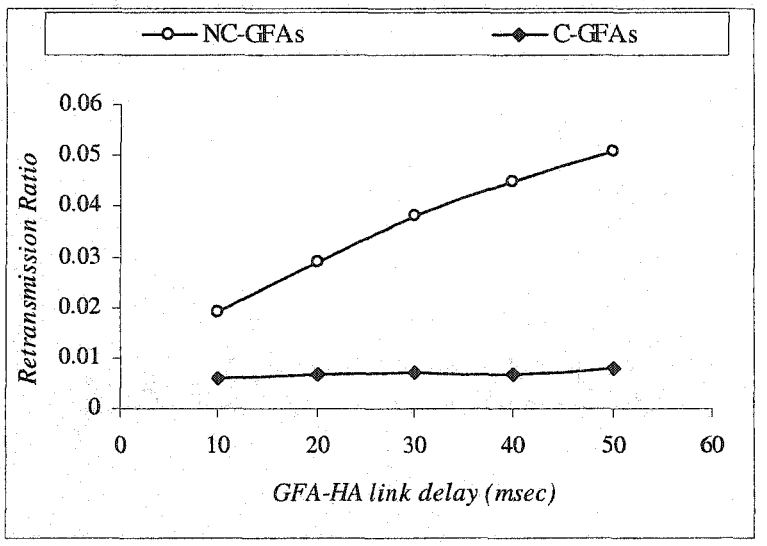


Fig. 45. TCP Retransmission ratio versus $LD_{HA\text{-}GFA}$.

4.6 Conclusion

In section IV, we presented a local-area mobility support framework based on Mobile IPv4 FA hierarchies, where a novel approach for scalable configurable cooperation between FA hierarchies in the foreign domain was introduced. FA hierarchies within the same foreign domain cooperate to reduce any unnecessary home registration with a possibly distant HA Hence, the HA is shielded from such local movement and home registration signaling overhead is reduced. The required extensions and modifications in processing of Mobile IP protocol messages were presented. When the MH is moving across FA hierarchies, the processing of protocol messages accounts for the MH's home-registered care-of address failure, such that the handoff delay due to such failure is reduced. The proposed solution maintains the same security measures as the base Mobile IP protocol in providing message authentication and replay protection. Moreover, the proposed cooperation framework requires a minimal number of security associations between deployed foreign agents belonging to different hierarchies. Performance

evaluation of the cooperation framework, using an extension we implemented for the network simulator *ns*-2 [8], have shown that up to 90% reduction in UDP traffic packet loss can be achieved compared to a non-cooperating set of FA hierarchies in the case of a distant HA. In addition, TCP traffic experiments have demonstrated that the proposed framework reduces the required number of data packets retransmissions compared to the non-cooperating approach, while achieving slightly higher throughput values. Thus, the effectiveness and feasibility of the cooperation framework have been successfully demonstrated.

In section V, we introduce the design and implementation details of our network simulation framework for local-area mobility. This framework has been used to conduct the simulation experiments described in sections III and IV.

SECTION V

NETWORK SIMULATION FRAMEWORK FOR LOCAL-AREA MOBILITY

Sections III and IV introduced a framework for local-area mobility support based on the deployment of multiple cooperating foreign agent hierarchies in the foreign domain. In this section, we present the design and implementation details of the network simulation used in evaluating the proposed mobility support framework. The Columbia IP Micro-mobility Software [15] (CIMS), which is a network simulator *ns*-2 [40] source code extension[23], was enhanced allowing the simulation of *n*-level foreign agent hierarchies, and providing foreign domain modeling capabilities. We automate the simulation of a configurable hierarchy model represented as a perfect *N*-ary tree with different types of network topologies. The introduced model allows specifying which hierarchy levels under the GFA belong to an overlay foreign agent tree superimposed on top of the hierarchy, such that other hierarchy levels are simple routers with no mobility support functionality. Furthermore, we introduce design changes to allow tunneling of data packets from any regional foreign agent node in the hierarchy to a lower hierarchy level towards the MH. Such design changes remove the restriction that the MH and the forwarding RFA need to be in the same network cluster as dictated by *ns*-2 addressing rules. Moreover, the GFA and HA can be selected as two different network entities, allowing true foreign domain modeling and the deployment of multiple foreign agent hierarchies in the foreign domain. In addition, a number of implementation enhancements were introduced such as support for regional registrations, periodical home registrations, and smooth handoff from the old to the new FA.

Section V is organized as follows. Section 5.1 introduces a brief overview of *ns*-2 highlighting relevant available mobile networking capabilities. Section 5.2 presents an overview of CIMS' *ns*-2 extension with emphasis on the provided FA hierarchy implementation. Section 5.3 presents CIMS-based design and implementation

[23] CIMS source code extension is based on *ns* version 2.1b6.

enhancements in order to simulate the proposed mobility framework. Section 5.4 presents a sample simulation scenario focusing on node and network topology configuration. Finally, section V is concluded in section 5.5.

5.1 Network Simulator Overview

The *ns*-2 network simulator is a publicly available object-oriented event driven simulator for computer networks. It is widely used in the research community because of source availability, modularity, and open architecture permitting components reuse and extension. The simulator implementation uses a split-language programming model, where C++ is used for implementing the simulator core when more processing is required to allow for fast simulations, while OTcl[24] is used as a command and configuration interface for simulation scenario description, dynamic network components configuration, and simulation events scheduling. *ns*-2 includes implementations for a number of network protocols and technologies such as network links (point-to-point, LANs, etc.), queuing models, IP and Mobile IP, unicast and multicast routing, transport protocols such as TCP and UDP, quality of service protocols, and network applications (FTP, Telnet, etc.) [26].

ns-2 supports the creation of wired network topologies consisting of stationary network nodes connected by links, which possess a queuing mechanism, delay, and throughput specification. Wireless topologies are comprised of wireless nodes connected by means of channels. Hybrid wired/wireless topologies can be constructed by combining wired and wireless topologies construction approaches. Packets manipulation (construction, consumption) is performed by agents, which are attached to nodes. Examples of agents are network-layer agents such as routing agents for packet forwarding, transport-layer agents such as TCP or UDP senders/sinks. Applications can act as traffic sources using agents to communicate with traffic sinks in other network nodes, e.g., an FTP application can use a sender TCP agent on one node to communicate with a sink TCP agent on another node.

---

[24] OTcl [66] is an object-oriented variant of Tcl (Tool Command Language).

We briefly introduce relevant node addressing information and mobile networking capabilities in *ns*-2, with emphasis on Mobile IP support.

Addressing and Packet Forwarding Support

A packet is addressed to a destination pair (*node address*, *port number*). The node address specifies which node is the final destination for this packet. The port number implies which agent within the destination node is responsible for processing this packet. An *address classifier* within the node processes incoming packets: packets addressed to this node are passed on to a *port classifier* to decide which agent is this packet handler, while packets addressed to other nodes are passed on to next hop destinations based on available routing information (e.g., network links in wired topologies).

ns-2 supports hierarchical node addressing for *n* levels to decrease routing table information. The default configuration supports 3 addressing levels, namely: *domains*, *clusters*, and *nodes*. For example, a node address of "1.2.3" implies this is domain 1, cluster 2 within domain, and node number 3 within cluster 2. A 3-level addressing structure dictates the presence of a node address classifier comprised of 3-level classifiers. A domain classifier contains one entry per domain. A cluster classifier contains one entry per cluster within the node's domain. A node classifier contains one entry per node within the node's cluster.

Wireless Communication and Mobile IP Support

ns-2 supports a pure wireless framework for simulating wireless ad-hoc networks. This framework implements wireless channels, radio propagation models, antennas, MAC protocols, link layer with address resolution protocol (ARP) support, and ad-hoc routing protocols. Random node movement is supported in a 2-dimensional grid. In addition, support was later added for a hybrid wired/wireless framework. A hybrid topology consists of wired nodes, base stations, and mobile nodes. Base stations act as gateways between wired and wireless nodes.

Mobile IP support was first developed for wired nodes, and later extended to permit wireless nodes mobility. Currently, only MIPv4 is supported, although separate researchers have developed *ns*-extensions for IPv6 and MIPv6 [25]. The supported MIPv4 components are HA, MH, and FA, with the MH using the FA as its care-of address. A MH is modeled as a mobile node, while HA and FA are modeled as MIPv4

base stations. Fig. 46 depicts the design of a wireless MIPv4 base station [26]. A *registration agent* sends out beacons (agent advertisements) periodically, and in response to a MH's solicitations. In addition, it forwards registration requests received from the MH to its HA, and relays registration replies received from the HA to the MH. A corresponding registration agent is attached to a mobile node, to send out solicitations, process agent advertisements, and handle Mobile IP registration signaling. A HA base station installs node classifier entries to point to a *packet encapsulator* when it receives registration requests from a FA, and stores the FA node address in a tunnel exit data structure. When a data packet is received at a HA, addressed to a MH, the encapsulator provides IP-in-IP encapsulation functionality, where the destination node (the FA) is determined by inspecting the tunnel exit data structure for the existence of current MH care-of address. By closely inspecting Fig. 46, we observe that the MH and HA need to be co-located within the same cluster, assuming 3 address levels, for later packet redirection to the packet encapsulator. An FA base station pre-installs a *packet decapsulator* agent. When a registration reply is received from the HA, classifier entries are installed within the decapsulator to point to the path to the MH (the network link leading to the MH in wired topologies). In wireless topologies, the decapsulator eventually forwards data packets to an ad-hoc routing agent.

The MH's handles agent advertisements in a simple manner, leading to a non-optimized handoff mechanism between base stations. Agent advertisements are stored in a linked list data structure that is periodically inspected to delete any expired entries based on the advertisement lifetime. The head of the linked list contains the FA list entry from which the MH received the most recent advertisement. If the current care-of address list entry has expired, the MH is prompted to issue a new registration request to the current head of the linked list. When a MH receives an advertisement from a FA that does not exist in the linked list, it immediately registers with its HA through this FA making it its serving base station. We can observe that if a MH is in an overlapping region between 2 base stations, it might keep constantly switching between base stations. When an advertisement is received from an already stored FA, this FA's list entry is moved to the head of the linked list, and if this FA is the current care-of address, a registration is initiated to refresh the HA's tunneling state. This approach is adopted

instead of a periodic home registration approach, in which the MH would not issue a new registration to its current FA before a periodic registration timer expires, unless it senses that the FA has rebooted for example by inspecting the current agent advertisement sequence number. Recently, Widmer implemented an optimized handoff mechanism as part of his *ns* extension to simulate a mobile network architecture for vehicles, where for example inspecting the distance between the MH and the BS is an integral part of the handoff decision [67].

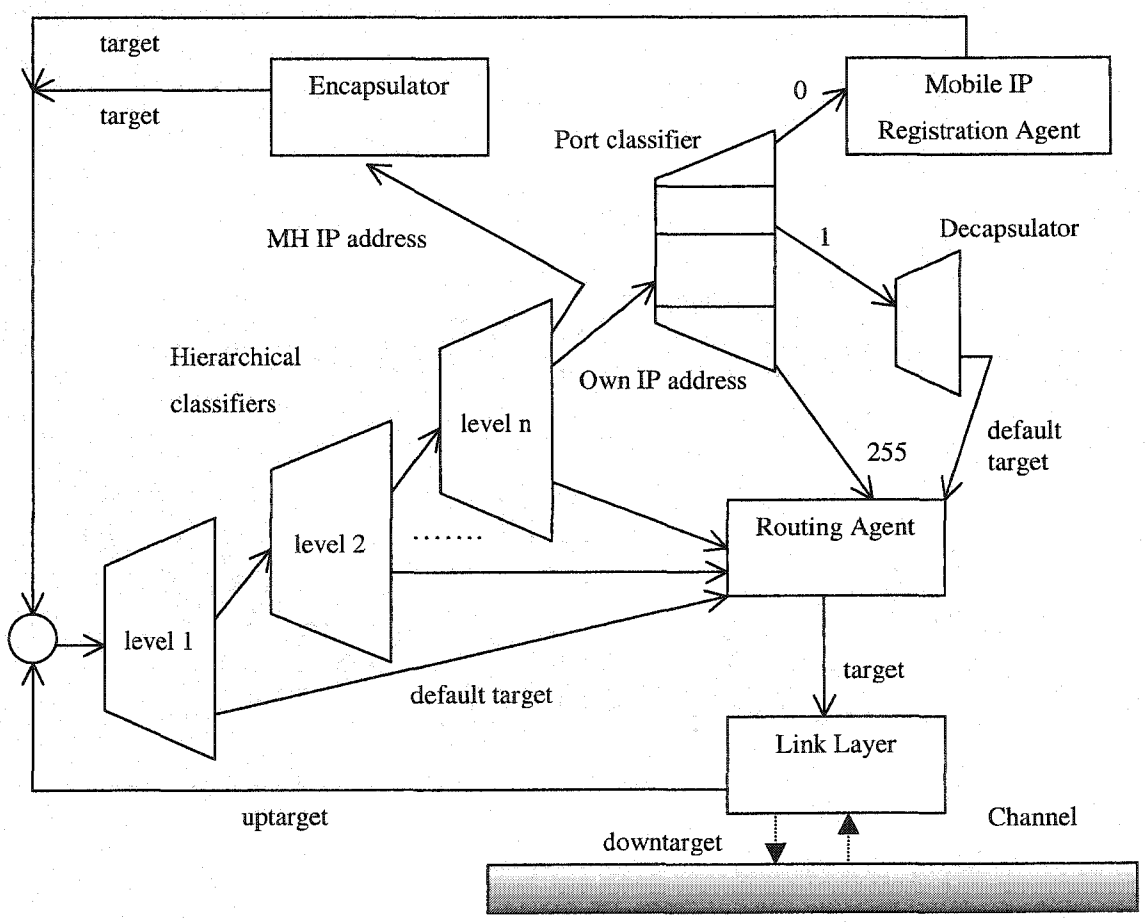


Fig. 46. The design of a wireless MIPv4 base station node in *ns*-2.

5.2 Columbia IP Micro-Mobility Software Overview

The Columbia IP Micro-mobility software (CIMS) [15] extends *ns*-2 by providing implementations for a number of micro-mobility protocols including Cellular IP [13], Hawaii [54], and Hierarchical Mobile IP [30]. CIMS has been used to compare the performance of these micro-mobility protocols [12]. We limit our overview presentation to the provided hierarchical Mobile IP implementation.

CIMS implements a simple foreign agent hierarchy where 1-level of foreign agents exists below a *Gateway Foreign Agent* (GFA) at the root of the hierarchy. A new node class is introduced modeling a GFA, which very much resembles the design of base stations (see Fig. 46) with the restriction that it does not send out agent advertisements. In addition, the GFA is selected to be the MH's HA in order not to change the existing tunneling infrastructure through the packet encapsulator. Such selection restricts the type of possible network topologies involving both a HA and a GFA, or multiple GFAs and a HA. Leaf foreign agents are modeled as *ns*-2 bases stations and only advertise their individual addresses but not the GFA's node address. A MH designates the GFA as its HA in its registration request, which prompts a receiving base station to forward the request to the HA which is actually the GFA. Data packets arriving at the GFA (HA) are encapsulated to the current care-of address as originally implemented in *ns*-2.

The implementation supports a single type of registration (home registration) from the MH to the HA (GFA) as opposed to home and regional registrations. This is attributed to the existence of only 1-level of FAs below the GFA, making the GFA the crossover FA for any MH's registration. Furthermore, periodic home registrations are not supported due to the previously outlined approach by which the MH handles received agent advertisements (see section 5.1).

ns-2 does not support any link layer notification that the MH is out of range with it current BS. CIMS added a notification to the old BS, through the new BS, when the new BS receives the MH's registration request with the old BS information. The old BS simulation object is directly accessed, and its state altered to signal the switch to being an old BS with regard to this MH. This BS state is used to immediately drop a data packet when this is an old BS, instead of attempting to forward it to the MH through the routing

agent. No smooth handoff mechanism implementation is provided to the foreign agent hierarchy.

CIMS uses a *Non-Ad-Hoc Routing Agent* (NOAH) [67] as its routing agent instead of the *ns*-provided full-fledged ad-hoc routing agents. With a full-fledged ad-hoc routing agent, a MH can use other MHs as intermediate routers instead of the serving BS, which creates a problem for the Mobile IP case, in which direct communication between a MH and its BS is assumed. With NOAH, a MH can directly communicate with it serving base station. Please refer to [67] for more details about NOAH and how direct communication between wireless nodes is enabled.

5.3 Network Simulator Design And Implementation Enhancements

In order to simulate the local-area mobility framework introduced in sections III and IV, a number of *ns*-2 and CIMS design and implementation enhancements were introduced. Such enhancements added the following features to the current CIMS implementation.

1. The capability of simulating multiple *n*-level foreign agent hierarchies within a foreign domain.
2. The capability of modeling a true foreign domain where the HA and GFA can be 2 different entities.
3. Support for home, regional, and home-regional registrations with periodical home registrations.
4. Support for the smooth handoff mechanism.

Section 5.3.1 introduces design and implementation enhancements to simulate a foreign domain with foreign agent hierarchies comprised of arbitrary number of levels. Section 5.3.2 explores the implemented support for home, regional, and home-regional registrations. Section 5.3.3 discusses the implemented support for the smooth handoff mechanism. Throughout the following sections, we introduce, where relevant, the OTcl API to be used from within network simulation scripts in order to properly configure various simulation aspects. Note that appendix A lists internal OTcl API, used from within the C++ implementation, for various simulation objects.

5.3.1 A Foreign Domain with Multiple Foreign Agent Hierarchies

A foreign domain consists of a number of foreign agent hierarchies. Our current implementation supports one foreign domain, i.e., all deployed hierarchies belong to the same foreign domain. Section 5.3.1.1 introduces the simulated foreign domain and foreign agent hierarchy model. Section 5.3.1.2 presents node design changes to model a true foreign domain where the GFA and HA can be 2 separate entities.

5.3.1.1 Foreign domain and foreign agent hierarchy model

An *n*-level foreign agent hierarchy is comprised of a GFA at the root of the hierarchy, RFAs at intermediate levels, and base stations providing wireless connectivity to MHs at the leaf level. We extended the GFA node class introduced by CIMS to function as a GFA and RFA, while MIP *ns*-2 base stations are used as leaf foreign agents. Each leaf and intermediate node in the hierarchy is informed about the list of its parent nodes' addresses leading to the GFA. GFAs and RFAs do not send agent advertisements, while base stations do, advertising the node's address. In addition, base stations can either advertise the GFA address, or the addresses of all parent nodes in the hierarchy leading to the GFA. The MH compares the advertised GFA address versus its current GFA address to decide whether it has changed foreign agent hierarchies. The following is the registration agent list of OTcl methods (used from within a BS or a RFA) introduced to set and get the type of base station advertisement, and to inform a node about the hierarchical path up to the GFA (set of RFAs between the node and the GFA).

set-adv-method **<adv-method>**

The implementation defines two advertising methods: HIERARCHICAL_ADV to advertise the full RFA path to the GFA, and GFA_ONLY_ADV to only advertise the GFA (in addition to the BS address). If the simulation script does not set the advertisement method, the default value is HIERARCHICAL_ADV. The possible set of alternatives for the advertisement method is stored in the global OTcl list VALID_ADV_METHODS.

get-adv-method

Get the advertisement method. Returns the current BS's agent advertisement method.

add-adds-coa-list **<coa-list>**

Add node addresses to the BS's list of care-of addresses. Add the node addresses in the list <coa-list> to the list of care-of addresses maintained by the registration agent. Element 0 in <coa-list> is the father

RFA of the current node, while element *n* is the GFA. If the selected advertisement method is HIERARCHICAL_ADV, the current node's address, along with all addresses supplied in <coa-list> are advertised. This method is equally used for an RFA/GFA node to inform it about its father RFA.

Any number of hierarchy levels and individual node degrees can be simulated. However, we automated the simulation of hierarchies configured as *perfect N-ary trees* with an arbitrary number of levels [52]. A perfect tree allows storing hierarchy nodes in a 1-dimensional array, starting with the root node as element 1, and the remaining hierarchy nodes consecutively stored in level-order from left to right. In addition, mathematical relationships can be derived to calculate some measures that permit automating the manipulation of such data structure. For a perfect *N*-ary tree with *n* levels, assuming the root node is at level 1, the total number of hierarchy nodes is given by the sum of the geometric series $1+ N +N^2+ \ldots+N^{n-1}$ which is $(N^n-1)/(N-1)$, the number of nodes in level *i* is given by N^{i-1}, the array index of the parent node of a node *M* is given by $\lceil (M-1)/N \rceil$, and the *N* array indices of the children of a node *j* are given by $N*(j-1)+2$, $N*(j-1)+3,.., N*j+1$.

Hierarchy levels can belong to a wired FA overlay tree superimposed on the hierarchy as follows. By default, a hierarchy is rooted by the GFA at level 1, and level *n* (the leaf level) consists of *ns*-2 base stations. Levels 1 and *n* are always a part of the FA overlay tree. The number of consecutive hierarchy levels below the GFA (excluding the leaf level), that are simple routers and have no RFA functionality, i.e., are not part of the FA overlay tree can be specified. For instance, a hierarchy configuration of "5/2/3" implies that the hierarchy is a 5-level hierarchy with the GFA at level 1, each node has a degree of 2 (2 children per node), and the 3 lower hierarchy levels are part of the FA overlay tree (level 5 is a base stations leaf level, levels 4 and 3 are RFAs levels, while level 2 consists of simple *ns*-2 nodes acting as wired routers). In general, with a hierarchy of *n* levels and a node degree of *N*, a hierarchy configuration can be expressed as "*n*/*N*/*n*-*i*" (*i*: $1 \rightarrow n$-1). The default hierarchy configuration is "*n*/*N*/1" and implies that no RFA levels exist below the GFA, while "*n*/*N*/*n*-1" is a hierarchy configuration where all hierarchy levels are part of the FA overlay tree with the upper *n*-1 levels being GFA/RFA

nodes, and the n^{th} level a base stations leaf level. Fig. 47 depicts the simulated model for foreign agent hierarchies along with the superimposed FA overlay tree.

Hierarchy nodes can be connected using any of the following alternatives.

1. Individual duplex links between each father FA and immediate children foreign agents;
2. A single local-area network (the entire hierarchy is located in one subnet);
3. A number of local-area networks (subnets) where each father and immediate children foreign agents are located in one subnet (the required number of subnets for a perfect n-level N-ary tree is given by $\lfloor$Total number of hierarchy Nodes / $N\rfloor$.

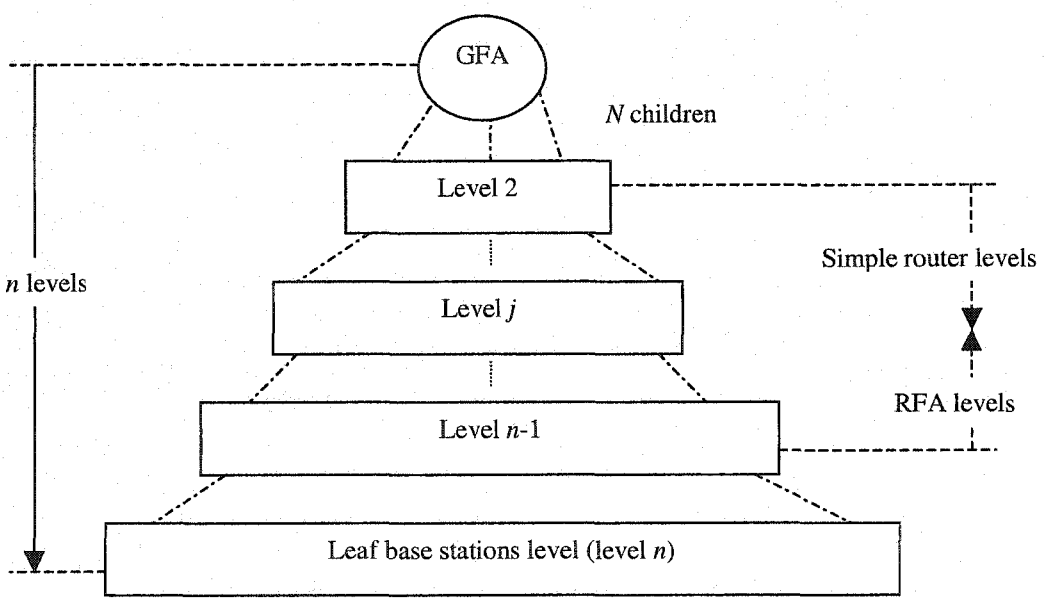


Fig. 47. The simulated model of foreign agents hierarchy.

A foreign domain may contain any number of FA hierarchies with different configurations and topologies. However, we automated the simulation of foreign domains where all FA hierarchies have the same hierarchy configuration using a multiple subnets topology, and each pair of GFAs are connected using individual duplex links. For example, a foreign domain configuration of "2/4/2/3" implies that this domain contains 2 hierarchies, each with the same hierarchy configuration of "4/2/3".

In order to automate the configuration of the base stations' wireless coverage, and their placement, we automated the BS placement strategy adopted by CIMS [15], where base stations are located on a 45 degrees inclined straight line, *L* meters apart. The value of *L* and a desired overlap area between BSs determine the cell coverage area. Consequently, increasing the hierarchy height increases the number of base stations at the leaf level, and the total wireless coverage area. The cell coverage area, receive threshold of the wireless physical medium, and antenna parameters are used to calculate the BSs' transmission power according to a two-ray ground reflection model [26].

5.3.1.2 A true foreign domain model

In order to simulate a true foreign domain where the GFA and HA are 2 separate entities, and to allow simulating multiple FA hierarchies (multiple GFAs) within the foreign domain, the GFA/RFA packet encapsulation infrastructure was modified to allow the possibility that the GFA/RFA and MH might not be in the same network cluster. In addition, a GFA/RFA might need to maintain a binding cache entry for a MH, e.g., when it receives a BU message with a specified lifetime as part of the KOPA approach (see section 3.4.1). In order to maintain the binding cache information, we reuse the tunnel exit data structure with expiration, while adding a tunnel type entry to distinguish between binding cache entries and visitor entries. The presence of a visitor entry allows an RFA to decide whether or not it is the crossover FA for a MH's registration request.

Fig. 48 illustrates the design of a GFA/RFA node. The encapsulator has been moved to the entry of the node to be consulted before the hierarchical address classifiers in order to inspect the tunnel exit (binding cache) before the routing tables (hierarchical address classifiers), when needed, and remove the restriction that the GFA/RFA and the MH need to be co-located within the same cluster.

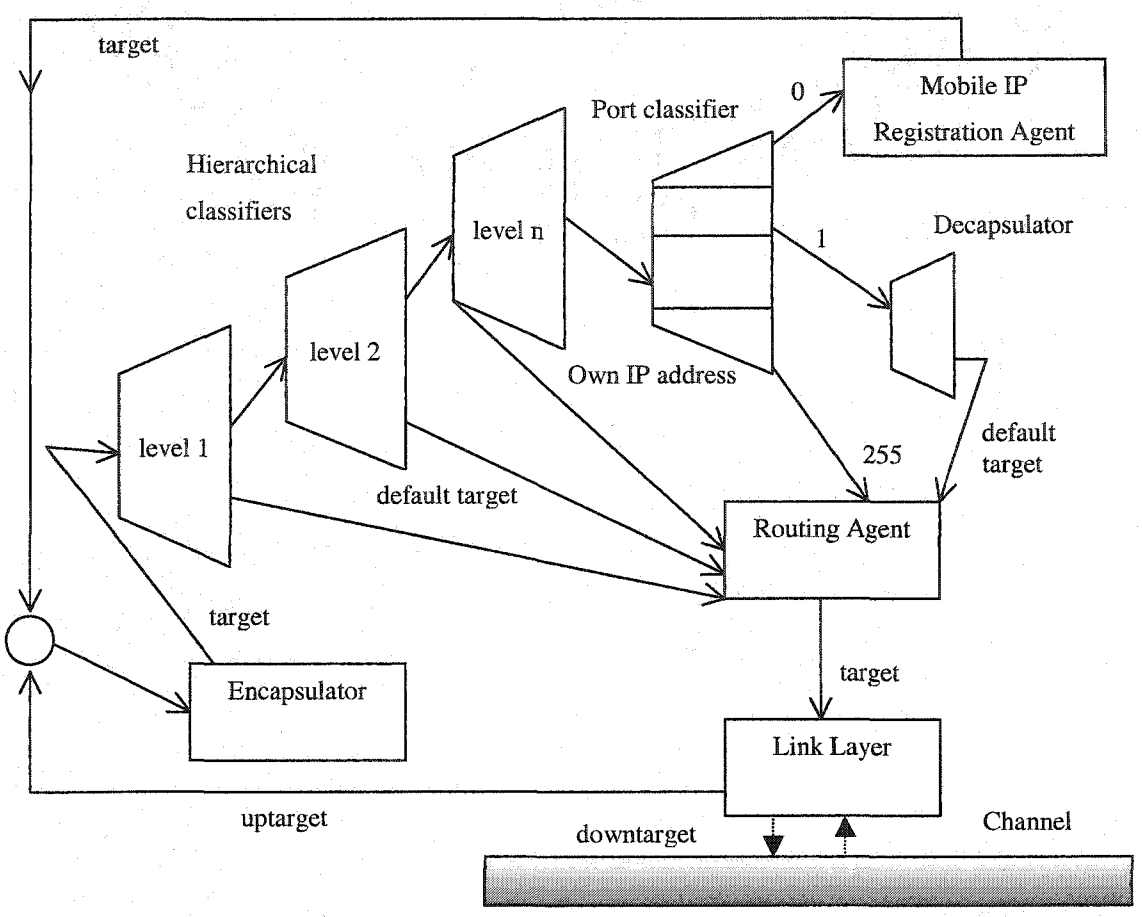


Fig. 48. Design of a GFA/RFA node.

Base stations and GFA/RFA nodes use the same encapsulator object; hence its functionality has been extended to allow its usage within either node types (Fig. 49). A HA provides IP-in-IP encapsulation for data packets destined to the MH. At a GFA/RFA node, if a packet is already encapsulated and destined to this node, the tunnel exit data structure is consulted to check for an entry specifying the current care-of address for the MH address. If a packet is not encapsulated, the MH address is assumed to be the destination address in the packet's IP header; otherwise it is the destination address in the inner IP header. GFA/RFA nodes perform re-tunneling to a child RFA by overwriting the outer IP header to reflect the new source (this node) and destination (the child FA's in the

hierarchy) information. Regardless of whether re-tunneling is performed or not, the packet is forwarded to hierarchical address classifiers to decide on the packet's next processing step (forward to a next hop, or forward to a port classifier if destined to this node).

```
/* On receive a packet p at encapsulator, process p to perform tunneling to new care-of address.
* p might be already encapsulated by a HA.
* if p is encapsulated, an inner and outer IP header exist.
*/
dst ← p's IP header destination address
if ((p already encapsulated) and (not addressed to this node))
        forward p to address classifiers
        return
endif
if (p already encapsulated)
        dst ← p's inner IP header destination address /* the MH address */
endif
lookup a care-of address in the tunnel exit data structure using dst
if (care-of address exists)
        if (p not encapsulated)  /* HA performs IP-in-IP encapsulation*/
                allocate new IP header
                fill allocated IP header with original IP header information
                designate the allocated IP header as p's inner IP header
        endif
        /* Overwrite the outer IP header with new source and destination information */
        replace outer IP header destination address with new care-of address
        replace outer IP header source address with this node's address
endif
forward p to address classifiers  /* whether p was actually encapsulated or not */
```

Fig. 49. Encapsulator packet processing pseudocode.

5.3.2 Support for Home and Regional Registrations

In order to simulate more realistic registration processing, support for periodic MIP home registrations has been added to *ns*-2. The existing MIP implementation continuously refreshed the home-registered binding to a FA by issuing a home registration to such FA regardless of the remaining home registration lifetime. This

approach was left unchanged for the basic MIP approach. Using the foreign agent hierarchy approach with regional registrations, the support for periodic home registrations alleviates the need for such unnecessary home registration overhead. In addition, such support was necessary to implement in order to compare the performance of the KOPA and SINP approaches (see section 3.4). The desired type of processing for home registrations involving local handoffs can be selected by using the following global OTcl command. The selected type of processing implies the tunneling consistency mechanism initiated by the crossover FA, and whether a regional registration reply is generated before a home registration reply is received from the HA.

set-home-renewal-handling **<method>**

Set the desired processing method for home registrations involving local handoffs. The implementation provides the following alternatives: KEEP_OLD_ALIVE for the KOPA approach, SWITCH_NEW for the SINP approach, DELETE_OLD for the DOP approach, and NOTHING. If the simulation script does not set the desired processing, a default value of KEEP_OLD_ALIVE is assumed. The possible set of alternatives for the cooperation mode is stored in the global OTcl list VALID_HOMER_HANDLING_METHODS. The processing mode is consulted from within the C++ implementation of the MIP registration agent to determine the appropriate course of action while processing the MH's home registrations.

Home Registration Scheduling

While sending a current home registration, the next home registration is scheduled after a time interval allowing for enough time for the future registration request to reach the HA, and for a registration reply to make it back to the GFA before the GFA's visitor entry for the MH expires. We selected this time interval to be "home registration lifetime - ¼ home registration lifetime". In section III, we presented how this approach could be modified to enforce a significant number of home registrations involving local handoffs (HR-LH) requests (see section 3.5). Alternatively, the MH could schedule the next home registration aided by the experienced home registration latency, leading to an adaptive home-registration scheduling process based on the current experienced delay. The home registration lifetime need not be a small time interval (order of milliseconds), or else it would create a lot of unnecessary home registration overhead. A typical value for home registration lifetime used during our simulation experiments is 30 seconds. The following

is the MH's registration agent list of OTcl commands introduced to set the home registration lifetime used by the MH, and to force a considerable number of HR-LH requests.

set homereg_lifetime_<value in seconds>

Set the home registration lifetime used by the MH. If the simulation script does not explicitly set the lifetime value, a default value of 50 seconds is assumed.

set forceHome_LocalHandoff_ <value>

Set how the MH handles the agent advertisements. A value of 0 implies not enforcing a considerable number of such registrations. A value of 1 implies to initiate home registrations while coinciding with a local handoff (see section 3.5)

MH's Processing of Agents Advertisements

Fig. 50 shows how a MH processes a received mobility agent advertisement in order to decide which registration, if any, should be initiated. Currently, the supported registration types are as follows.

1. *Home registrations*: To change the home-registered care-of address
2. *Regional registrations*: To change the local care-of address within a local-area while maintaining unchanged the home-registered care-of address.
3. *Home-regional registrations*: Initiated when handing off to a new hierarchy in the foreign domain, while in hierarchy cooperation mode (see section IV).

While the home registration is still valid and the advertisement received from the serving FA did not expire, the MH does not issue any further registrations (home or regional) to this FA. While the MH is moving within the same foreign agent hierarchy, a regional registration is initiated upon receiving an advertisement from a new FA (a new FA implies that the MH does not currently store this FA's advertisement in a current agents' advertisements data structure). Receiving an advertisement from a known FA (advertisement still valid, i.e., stored in the advertisements linked list) while having a current care-of address prompts the MH to move this FA advertisement entry to the head of the advertisements linked list, but no registration is initiated.

```
/* On receive a mobility agent advertisement (ad), process the ad to decide which registration is initiated.
* An ad contains at least the agent's IP address. If FA hierarchy mode is supported, ad will contain a set
of IP addresses (addr1, ..., addrn) where addr1 is the node's IP address and addrn the GFA address
* current_coa is the MH's home-registered care-of address.
* a current (not expired) agent advertisements linked list is maintained. */
regDest ← advertised node IP address  /* addr1 */
target_coa ← regDest
target_coa ← Advertised GFA IP address if applicable /* addrn */
lookup regDest in current advertisements linked list
if (regDest entry found)  /* a known mobility agent */
        /*Entry at head of advertisements list is used as potential coa if current_coa has its ads expired*/
        place regDest entry at head of advertisements linked list
        if (regDest == target_coa)  /* non hierarchy case, HA or FA */
                if (current_coa == regDest)
                        issue home registration to regDest using it as care-of address
                endif
        else /* FA advertising GFA and this is a known FA*/
                /* do nothing */
                /* This processing implies keeping the current_coa the same until its ads expire, as long
                as no other ads were received from a new mobility agent */
        endif
else /* a new mobility agent */
        create a new advertisement entry for regDest
        place newly created entry at head of advertisements linked list
        if (regDest == target_coa) /* non hierarchy case */
                issue home registration to regDest using it as care-of address
        else /* hierarchy case */
                if (current_coa == target_coa) /* same GFA, new FA */
                        issue regional registration to regDest
                else /* new GFA, new FA */
                        if ((current_coa == HA) or (current_coa == UNASSIGNED))
                                issue home registration to regDest using target_coa as the new care-
                                of address
                        else
                                if (no cooperation mode) /* new home registration */
                                        issue home registration to regDest using target_coa as new
                                        care-of address
                                else /* cooperation mode between hierarchies */
                                        if (moving within an already registered hierarchy)
                                                /* The MH has already issued a home-regional
                                                registration from within this hierarchy and was
                                                approved by the root GFA for this MH*/
                                                issue a regional registration to regDest
                                        else
                                                issue a home-regional registration to regDest
                                        endif
                                endif
                        endif
                endif
        endif
endif
```

Fig. 50. The MH's processing for a received mobility agent advertisement.

FA Hierarchies Cooperation

When multiple hierarchies are deployed in the foreign domain, two hierarchy cooperation policies are implemented: non-cooperating and cooperating hierarchies (see section IV). In non-cooperation mode between hierarchies, a handoff to a new hierarchy in the domain implies sending a new home registration, effectively changing the MH's home-registered care-of address. In cooperation mode, a handoff to a new hierarchy implies sending a home-regional registration request, while handoffs within an already approved hierarchy by the root GFA implies sending regional registrations. A return to the MH's root GFA hierarchy triggers a regional registration, since the home-registered care-of address was not changed (see section 4.4.2). The following global OTcl command allows setting the desired cooperation mode between FA hierarchies within the same foreign domain.

***setHFA_RoutingType* <type>**

Set the desired cooperation mode between FA hierarchies within the same foreign domain. The implementation defines two possible alternatives: REGIONAL for a non-cooperation mode, or GFA_COOPERATION for a cooperation mode. If the simulation script does not set the cooperation mode, a default value of REGIONAL is assumed. The possible set of alternatives for the cooperation mode is stored in the global OTcl list VALID_HFA_ROUTING_TYPES. The cooperation mode is inspected from within the C++ implementation of the MIP registration agent to determine the appropriate course of action while processing inter-hierarchy handoffs.

Registration Replay Protection

Timestamp-like replay protection is provided for all MH registrations (home or regional) at the HA and GFA/RFA. The MH supplies, as part of its registration requests, a registration sequential number (continuously incremented for all registration types). The HA and GFA/RFA store and update the latest registration sequential number to aid in judging the freshness of a received MH's request. During the simulation, such registration sequential number maintenance and inspection guards against erroneously processing an older registration when a newer registration has been received. Such scenario can occur due to differences in network delays and congestion state on different simulation links leading from the old and new FA to the HA or crossover FA.

Pending Registrations

The introduction of an *n*-level FA hierarchy necessitates the presence of a pending registrations request data structure (implemented as a linked list) at each RFA/GFA in order to match registration replies to registration requests. The latest pending request is inserted at the head of the pending requests linked list. A pending registration request entry contains the following data fields.

1. *The MH address.*
2. *The registration's sequential number* (provided by the MH as part of the request). Matching registration replies to registration requests is based on comparing the stored pair (MH address, sequential number) in the pending entry versus the registration reply corresponding data fields.
3. *The address of the request's sending node* to identify which child RFA forwarded this request. A later registration reply is forwarded to the stored sending child RFA, until the reply eventually reaches the MH.
4. *How to handle this pending request entry when a reply is received.* This field is necessary because of the introduction of the SINP approach (see section 3.4.2), where a regional registration reply precedes a home registration reply in flowing through the same new path to the MH. If the pending entry is removed after the regional reply is received, an RFA would not know to which child RFA it should forward the later home registration reply. Possible handling options are: delete after receiving any corresponding reply (the default case), delete after only receiving a home registration reply, and delete after receiving a regional registration reply.
5. *The simulation time at which the request was received.*

5.3.3 Smooth Handoff Mechanism Support

The smooth handoff mechanism (PFANE [47]) is supported as follows. A MH supplies as part of its registration request its old FA address (old BS), and a desired binding cache lifetime. The new FA (new BS), upon receiving the registration request, sends a BU message to the old FA. The old FA, upon receiving the BU from the new FA, saves in a "new FA" data structure the current binding cache entry pointing to the new

FA. Such binding cache entry expires after the desired lifetime requested by the MH. The MH uses a default binding cache lifetime of 0.5 seconds. We did not implement the binding acknowledgement feature, by which an acknowledgement message is formulated by the old FA and destined to the MH. Such message should be encapsulated to the new FA, which later delivers it to the MH. The following is the MH's registration agent OTcl command to enable the usage of the smooth handoff mechanism within the MH's registration requests.

set useSmoothHandoff_ <value>

A value of 0 implies that the MH does not request the new FA to initiate a smooth handoff mechanism. A value of 1 implies to request such mechanism from the new FA. If the simulation script does not explicitly select which mode to use, a default value of 0 is assumed.

We extended the decapsulator functionality to add the capability of encapsulating data packets from the old FA to the new FA (the old and new FA are leaf base stations in our foreign agent hierarchy model). We opted for this approach, to support encapsulation to any MH, even if not within the same cluster as the old FA. The existing *ns*-2 implementation only supported encapsulation from a HA to the current MH care-of address, provided that the HA and MH are in the same cluster. An alternative approach would have been to apply the same node design changes presented earlier to support foreign agent hierarchies (see section 5.3.1), to *ns*-2 base station node structure, and use a binding cache tunneling approach. We opted to keep *ns*-2 base station node structure intact, and change the functionality of the decapsulator component of the node. Note that, a data packet would reach the decapsulator component only if it is addressed to this node (see Fig. 48), which fits the scenario that this node is an old FA receiving an encapsulated packet destined to the MH.

Within a base station, a data packet that reaches its decapsulator agent is processed according to the state of this base station as follows (Fig. 51).

- *Base station currently serving the MH.* The packet is decapsulated and forwarded to the MH (through a wireless routing agent).
- *Base station currently an old FA that was informed about the new FA through the smooth handoff mechanism.* The data packet is already an encapsulated

packet; hence the MH address is assumed to be the destination address of the inner IP header. The "new FA" data structure is consulted to extract the new care-of address information, and afterwards the packet is re-tunneled to this new care-of address, by overwriting the outer IP header using the new source and destination information. The new resulting packet is forwarded to the node's hierarchical classifiers to decide on its next hop towards the new FA.

The new FA relays any data packets tunneled from the old FA to the MH, even before a registration reply, corresponding to a MH's registration request, is received. In adopting such approach, we rely on the fact that the old FA is responsible for authenticating the MH's request informing it of its new care-of address.

```
/* On receive a packet p at decapsulator, process p to extract original packet and forward it to MH.
* if this node is an old FA, p is forwarded to the new FA address if found in a "new FA" data structure.
* p is already encapsulated with an inner and outer IP header.
*/
MH address ← p's inner IP header destination address
if (this BS currently serving MH) /* decapsulation */
        replace outer header with inner header
        delete inner header
endif
if (this BS previously serving the MH) /* an old FA */
        lookup new FA address in "new FA" data structure using MH address
        if (new FA address found) /* forward p to new FA */
                /*Overwrite outer header with new source and destination information */
                replace outer IP header destination address with new FA address
                replace outer IP header source address with this node's address
                forward p to address classifiers to decide on p's next destination towards the new FA
                return
        else
                drop p
                return
        endif
endif
forward p to next target        /* Next target would be a wired link towards the MH, or a wireless
routing agent*/
```

Fig. 51. Decapsulator packet processing pseudocode.

5.4 A Sample Simulation Scenario

In this section, we present a sample simulation scenario highlighting how the main simulation aspects are configured, while focusing on the FA hierarchy topology generation. We introduce as sample scenario, the simulation of a foreign domain comprised of a number of FA hierarchies. An individual FA hierarchy's nodes are connected using multiple subnets, where each father and children FA are on the same subnet. The foreign domain and FA hierarchy model are configured as introduced in section 5.3.1. For more information about writing *ns*-2 simulation scripts in general, the reader is referred to the *ns*-2 tutorial [29].

Simulation Command Line Parameters

The simulation accepts the command line parameters depicted in TABLE 4 in their listing order. All parameters are optional, i.e., if not provided, default values are assumed.

TABLE 4

SAMPLE SIMULATION SCENARIO COMMAND LINE PARAMETERS

Parameter	Description
hiernum	The number of FA hierarchies within the foreign domain. The default value is 2
levelnum	The numbers of levels within each FA hierarchy in the foreign domain. The default value is 4.
fanout	The fanout (number of children nodes) of each node. The default value is 2.
level_HFA	The number of levels in the FA hierarchy tree, apart from the GFA, that belong to the RFA overlay tree. A value of 1 indicates only the leaf level (BS level) belong to the overlay tree. The default value is 1.
speed	The movement speed of the MH in meters/second. The default value is 20 meters/second.
InterBS_Distance	The distance between adjacent base stations (see section 5.3.1.1). The default value is 140 meters. Such distance is the same for both dimensions of a two-dimensional grid.
overlap	The desired overlap distance in meters between adjacent BSs coverage areas. The default value is 25 meters. The <interBS_Distance> and <overlap> are used to compute the BS's effective coverage area.

Simulation Configuration

Numerous configuration values need to be assigned from within the simulation script. We classify and group such simulation configuration values according to their function. The following script fragment introduces some of the main configuration values.

```
# Traffic type to be used during simulation. Possible values are UDP and TCP
set opt(transport)     TCP

#instantaneous TCP throughput measurement interval
set tcp_opt(time_interval)  2.0        ;#time interval in seconds for TCP throughput calculations

# TCP instantaneous throughput file, used to log such information every time interval
# a line in this file is formatted as follows <current_time Bandwidth_during_Interval Bandwidth_Mbps>
set opt(tcpf)        "TCPlogfile"

# Default hierarchy settings
set hier_opt(hiernum)       2  ; #default is 2 hierarchies
set hier_opt(levelnum)      4  ; #with GFA at level 1
set hier_opt(fanout)        2  ; #2 children per node
set hier_opt(level_HFA)     1  ; #only leaf level (BSs) belongs to FA overlay tree
set hier_opt(startX)        1.0 ; #X coordinate of first BS in 2D grid from left
set hier_opt(startY)        1.0 ; #Y coordinate of first BS in 2D from left

#Links settings for duplex links
set link_opt(CH_HA_linkB)           1.5Mb; #link bandwidth betwn CH and HA
set link_opt(CH_HA_linkDelay)       20ms ; #link delay betwn CH and HA
set link_opt(HA_GFA_linkB)          1.5Mb; #link bandwidth betwn HA and GFA
set link_opt(HA_GFA_linkDelay)      20ms ; #link delay betwn HA and GFA
set link_opt(inter_GFA_linkB)       10Mb ; #link bandwidth between GFAs
set link_opt(inter_GFA_linkDelay)   1ms  ; #link delays between GFAs

#LAN settings used for hierarchy LANs
set lan_opt(bw)         10Mb
set lan_opt(delay)       1ms

# mobile IP settings
set mip_opt(homereg_lftm)                30          ;#home registration lifetime from within HFA
set mip_opt(hfa_routingtype)             $GFA_COOPERATION
set mip_opt(agent_adv)                   $HIERARCHICAL_ADV ; # type of agent advertisement
set mip_opt(home_renewal_handling)       $KEEP_OLD_ALIVE ; # type of home registration processing
set mip_opt(reg_rtx)                     1.0         ;# time interval to attempt to retransmit a registration
set mip_opt(useSmoothHandoff)            0           ;# 0 do not use, 1 use
set mip_opt(forceHome_LocalHandoff)      0           ;# 0 do not force, 1 force
# Probability of failure to contact HRGFA from current GFA
set mip_opt(rzfa_contactFprob)           0.0

#  UDP settings
# interval and packet size correspond to 64Kbps audio
set udp_opt(interval)          20ms   ;# interval between packets
set udp_opt(packet_size)       160    ;# packet size  in bytes
;# playout delay for Loss Monitor in ms to calculate application-dropped packets
set udp_opt(playout_delay)     48
```

The usage of the FA hierarchy mode within *ns*-2 has to be enabled, along with the desired cooperation between deployed FA hierarchies. In addition, the desired processing for home registrations involving local handoffs is selected. The following script fragment shows such configuration commands.

```
# Enable FA hierarchy usage, introduced by CIMS
set HFA_Routing 1

# Select which FA hierarchy cooperation mode
set HFA_Routing_Type $mip_opt(hfa_routingtype)

# Select the home registration processing
set-home-renewal-handling $mip_opt(home_renewal_handling)
```

Node Creation and Topology Generation

Each FA hierarchy is a perfect *n*-level *N*-ary tree (where *n* is the *hiernum* argument, and *N* is the *fanout* argument). The following script fragment presents how to compute various topology related measures such as the total number of nodes, total number of LANs, and the total number of base stations nodes.

```
# num_nodes_level <fanout> <levelnum> computes the number of nodes in a specific level
#get_num_nodes <fanout> <levelnum> computes the total number of nodes in the hierarchy

#Number of BS nodes in 1 hierarchy
set num_bs_nodes      [num_nodes_level $fanout $levelnum]
set totalnum_bs_nodes    [expr $num_bs_nodes * $hiernum]

#Total number in hierarchy
set hier_nodes       [get_num_nodes $fanout $levelnum]
set totalhier_nodes   [expr $hier_nodes * $hiernum]

#Number of wired nodes
set num_wired_nodes   [expr $hier_nodes - $num_bs_nodes]

#Number of required LANs for 1 hierarchy
set num_lans [expr int(floor([expr $hier_nodes/$fanout.0]))]
set totalnum_lans    [expr $num_lans *$hiernum]

#Total number of wireless nodes
# number of base stations + 1 MH +1 HA
set opt(nn)          [expr $totalnum_bs_nodes + $num_mobile_nodes + 1]
```

In terms of *ns*-2 node addressing, all nodes belong to the same domain. The MH and the HA are members of the same node cluster, while other nodes belong to individual one-node clusters. In addition, an *ns*-2 LAN needs to be allocated its own cluster for addressing purposes. The following script fragment shows how to setup node addressing for this configuration.

```
# All nodes in 1 domain
AddrParams set domain_num_ 1

# Number of clusters = total number of hier nodes + 1 (for MH and HA) +1 for CH + totalnum_lans for
LANS
lappend cluster_num [expr $totalhier_nodes + 1 + 1 + $totalnum_lans]
AddrParams set cluster_num_ $cluster_num

# multiple hierarchies
# Number of nodes  = 1 in each cluster for hier and CH + 2 for MH cluster
# The order of clusters is as follows: hierarchy clusters, CH cluster, MH cluster, and LAN clusters for HFA
# CH_cluster = totalhier_nodes
# MH_cluster = totalhier_nodes + 1
# lan_clusters start at  totalhier_nodes + 2
for {set i 1} {$i <= $totalhier_nodes } {incr i} {
        lappend eilastlevel 1
}
lappend eilastlevel 1 2;

#clusters for LANs
for {set i 1} {$i <= $totalnum_lans } {incr i} {
        lappend eilastlevel 1
}
AddrParams set nodes_num_ $eilastlevel
```

The perfect tree features allow us to store hierarchy nodes in two one-dimensional arrays. One array holds all wired nodes (do not have wireless capabilities), in level order from left to right, in each hierarchy, while the other array holds the base station nodes (wireless nodes) within each hierarchy ordered from left to right. Consequently, for wired nodes, the array index difference between corresponding wired nodes in each hierarchy is the number of wired nodes in a single hierarchy. Similarly, for base station nodes, the array index difference between corresponding base stations in each hierarchy is the number of base station nodes in a single hierarchy.

The creation of hierarchy nodes can be subdivided into the following tasks: the creation of the wired nodes which are not part of the FA overlay tree for each hierarchy, the creation of GFA nodes in each hierarchy, the creation of RFA levels within each hierarchy, and the creation of BSs (leaf level) within each hierarchy. The following script fragment illustrates the node creation process for base station nodes for each hierarchy in the foreign domain.

```
# Create base station nodes in each hierarchy
for {set i 1} {$i <= $num_bs_nodes } {incr i} {
   #Calculate the node number in tree
   set nodenum_intree [ expr $num_wired_nodes + $i]

   #Calculate the cluster number corresponding to this node
   set cluster_num [ expr $nodenum_intree -1]
   set index $i

   # For each hierarchy
   for {set k 1} {$k <= $hiernum} {incr k} {
      # configure the transmission power of base station
      $ns node-config -rxPower $power -txPower $power

      # create the base station
      set BS($index) [$ns node 0.$cluster_num.0]

      #set the advertisement method for the BS registration agent
      [$BS($index) set regagent_] set-adv-method $mip_opt(agent_adv)

      # Inform the BS about the hierarchical set of RFA nodes between it and the GFA
      # parents_list <node_number> <fanout> <hier_num> returns a list of node addresses which constitute
the parent list for this BS including the GFA
      [$BS($index) set regagent_] add-adds-coa-list [parents_list $nodenum_intree $fanout $k]

      #setup BS as foreign BS in HFA, introduced by CIMS
      makeHfaBS $BS($index)

      # Prepare for next iteration
      set cluster_num [expr $cluster_num + $hier_nodes]
      set index [expr $index+  $num_bs_nodes]
   }
}
```

Each father FA and children FAs belong to the same subnet. Due to the perfect tree representation, the nodes in each LAN can easily be identified. The following script fragment illustrates LAN creation and configuration for hierarchy nodes.

```
#Create subnets between hierarchy nodes
# Each Father and children nodes form a LAN (subnet)
set lan_cluster_offset [expr $totalhier_nodes +2]

# For each hierarchy
for {set k 1} {$k <= $hiernum} {incr k} {
  # For each node in the current hierarchy
  for {set i 1} {$i <= $num_wired_nodes } {incr i} {

      #get list of nodes in LAN rooted by node number I
      #lan_nodes <node_number> <fanout> <Number of wired nodes> <hier_num> returns a list of node
addresses belonging to the same subnet rooted by node <node_number>
      set current_lan_nodes [lan_nodes $i $fanout $num_wired_nodes $k]

      #Calculate the lan cluster
      set current_lan_cluster [expr $lan_cluster_offset+$i-1]

      # Create the lan
      set lan [$ns newLan $current_lan_nodes $lan_opt(bw) \
                              $lan_opt(delay) -llType $lan_opt(ll) -ifqType $lan_opt(ifq) \
                              -macType $lan_opt(mac) -chanType $lan_opt(chan) \
                              -address "0.$current_lan_cluster.0"]
      #$lan cost 2
  }

  #Calculate the next lan cluster offset
  set lan_cluster_offset [expr $lan_cluster_offset +$num_lans]
}
```

The configuration of the MH can be performed using the following commands.

```
# This command was introduced by CIMS and later extended to include new configuration parameters
makeHfaMH <MH node>  <home registration lifetime>  \
              < Is Smooth Handoff enabled> <Force Home registrations involving local handoffs>

where <home registration lifetime> is $mip_opt(homereg_lftm) \
       <Is Smooth Handoff enabled> is $mip_opt(useSmoothHandoff) \
       <Force Home registrations involving local handoffs> is $mip_opt(forceHome_LocalHandoff)

# Inform the MH's registration agent about the MH's home agent
# $HAaddress is the node address of the HA
[$MH set regagent_] set home_agent_ $HAaddress
```

Other Simulation configuration tasks

Other simulation configuration tasks not covered in this section include the placement of base stations in the two-dimensional grid (see section 5.3.1), the configuration of the MH movement, and the configuration of traffic streams parameters.

5.5 Conclusion

In section V, we introduced the design and implementation details of the network simulation used to evaluate the local-area mobility framework presented in sections III, and IV. The Columbia IP Micro-Mobility Software (CIMS) [15], which is an *ns*-2 source code extension [40], provided limited simulation capabilities for 1-level of foreign agents (base stations) below the GFA, and a restricted foreign domain model requiring that the GFA and HA be the same entity. We extended CIMS to allow modeling foreign agent hierarchies with an arbitrary number of levels, and a true foreign domain where the GFA and the HA can be two separate entities. Separating the GFA and HA entities allows deploying multiple foreign agent hierarchies within the foreign domain. In addition, a number of implementation enhancements were added such as support for regional and periodical home registrations, and the smooth handoff mechanism. To the best of our knowledge, this is the first network simulator extension to implement a true foreign domain where local-area mobility is supported by deploying one or more foreign agent hierarchies implementing the regional registration framework and periodical home registrations.

In section VI, we use the presented extensions and hierarchy simulation model to present a study of some of the FA hierarchy design parameters such as the hierarchy's height, and topology.

SECTION VI

A SUITE OF SIMULATION EXPERIMENTS FOR FOREIGN AGENT HIERARCHIES

In this section, we present a suite of foreign agent hierarchies simulation experiments, using the aforementioned network simulator extension and enhancements detailed in section V. Our objective is to highlight the validity and effectiveness of our extensions and implementation. Furthermore, we exploit these experiments to present a study of some of the network factors affecting the performance of an FA hierarchy based on the regional registration approach, e.g., hierarchy height, delay to the HA, topology, and the usage of the smooth handoff mechanism.

Section VI is organized as follows. Section 6.1 presents an overview of the simulation experiments presented in section VI. The subsequent sections present the following simulation experiments and corresponding UDP/TCP results as applicable: effect of FA hierarchy height and number of RFA levels (section 6.2) and hierarchy link delay (section 6.3) while using a duplex links hierarchy topology, effect of link delay between HA and GFA (section 6.4), and effect of hierarchy topology and the smooth handoff mechanism (section 6.5). Finally, section VI is concluded in section 6.6.

6.1 Simulation Experiments Overview

For the purpose of the experiments, we use the regional registration processing framework (section 3.3), and the KOPA approach for processing home registrations involving local handoffs (section 3.4.1) previously presented in section III. However, we do not enforce a large number of home registrations involving local handoffs as previously adopted in section III. Alternatively, a MH issues a periodical home registration request upon timer expiration as explained in section 5.3.2, which might or might not coincide with a local handoff. In general, the number of HR-LH requests, observed during these simulations, was limited to a handful of occurrences. The main observed scenario where HR-LH requests occurred was due to the MH receiving an

advertisement from a new FA, while the MH had just issued a home registration request (no handoff involved) to its current serving FA. Such event prompts the MH to issue another home registration request involving a local handoff. The home registration lifetime is set to be 30 ms.

We assume a foreign domain comprised of one foreign agent hierarchy. Fig. 52 illustrates the simulated network topology. The FA hierarchy is an n-level perfect binary tree where n is dependent on which experiment is performed. Unless otherwise stated, each FA is only connected to its children foreign agents through individual 100 Mbps duplex links and link delay $LD_{FA\text{-}FA}$ in milliseconds (default link delay is 0.5 ms). The GFA is connected to the MH's HA through a 1.5 Mbps duplex link with delay $LD_{GFA\text{-}HA}$ ms (default link delay is 20 ms). We simulate a single MH within the hierarchy communicating with a fixed *Correspondent Host* (CH). The CH is connected to the HA using a 1.5 Mbps duplex link with 20 ms link delay.

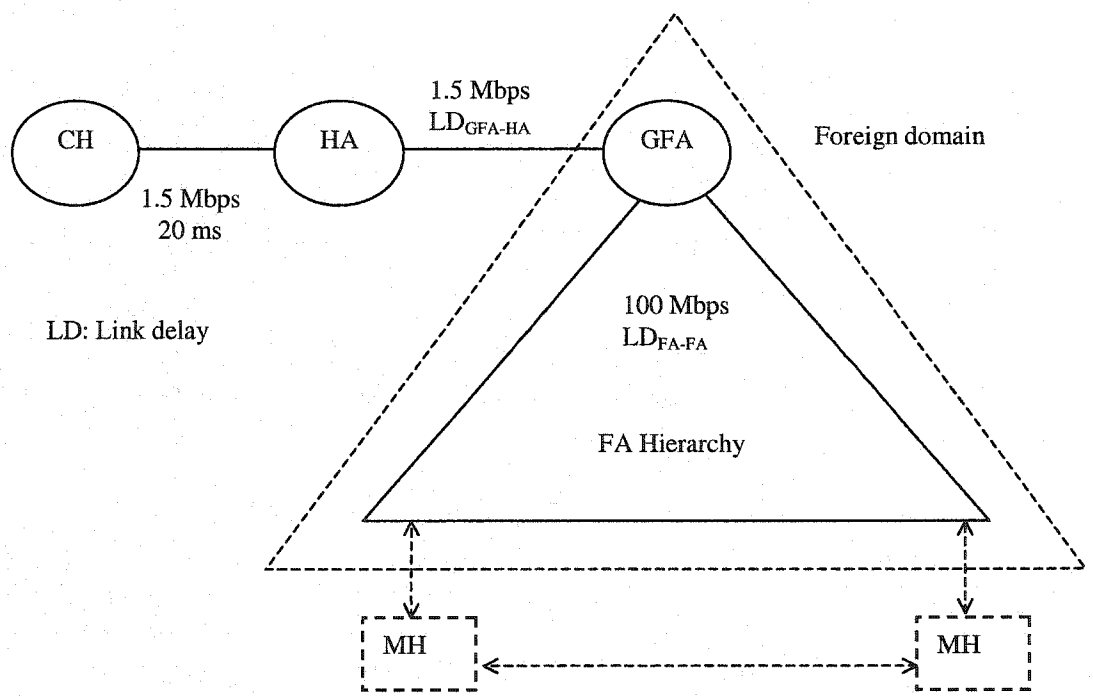


Fig. 52. Simulated network topology.

Unless otherwise stated, the MH is periodically moving back and forth across the entire domain until an estimated number of handoffs are attained (more than 100 handoffs). Consequently, the MH is performing a set of handoffs with a variable number of hops between the serving FA and the crossover FA. In general, within a perfect n-level FA tree, with only individual links between a father and children foreign agents, the type of handoffs performed by the MH can range from 1-hop to (n-1)-hop handoffs.

6.2 FA Hierarchy Height and Number of RFA levels

We investigate the effect of increasing the hierarchy height on performance during handoffs. We perform this task by actually adding extra RFA levels to an FA hierarchy. Adding an extra RFA hierarchy level implies increasing the wireless coverage area of the leaf base stations level, and increasing the tunneling overhead within the hierarchy[25]. In addition, for every hierarchy height n, we investigate the effect of varying the number of consecutive RFA levels above the leaf base stations level from 0 to n-1 (see section 5.3.1) to highlight the effect of bringing the crossover FA closer to the MH. For example, when the number of RFA levels is 0, the GFA is the crossover FA for all MH handoffs, located (n-1) hops away from the serving BS. By installing one level of RFAs above the base stations, some of the (n-1)-hop handoffs are substituted by 1-hop handoffs, with the crossover FA located 1-hop away from the serving BS.

6.2.1 UDP Traffic

The MH is periodically moving back and forth across the entire domain at a speed of 20 m/s. A constant bit rate UDP traffic is applied from the CH to the MH to simulate the behavior of an audio application (a 160-byte data packet is transmitted every 20 ms to simulate a 64 kbps audio stream). The experiment is performed for hierarchy heights 4, 5, and 6. For every hierarchy height, we calculate the average number of lost packets per handoff by dividing the total number of lost packets by the total number of handoffs. In addition, using hierarchy heights 4 and 5 as sample, we calculate the number of regional

[25] A hierarchy height increase from n to n+1 implies that a data packet has to go through one extra re-tunneling step to reach the MH.

registration replies generated by each RFA, and classify such replies according to the number of hops between the crossover FA and the serving FA.

Fig. 53 depicts the average lost packets per handoff while varying the number of intermediate RFA levels for hierarchy height 4. Modeling the foreign domain as an FA hierarchy, even without any RFA levels (RFA levels = 0), achieves a significant improvement versus the base Mobile IP case (average lost packets of 2.9 packets/handoff in base case versus 0.2 packets/handoff in hierarchy case which is a 93% reduction in packet loss). Increasing the intermediate number of RFA levels from 0 to 1 further reduces the average lost packets/handoff from 0.2 to 0.1 packets/handoff. This is due to the fact that some of the 3-hop handoffs, in case RFA levels = 0, are replaced by 1-hop handoffs when RFA levels = 1 (Fig. 54). This substitution reduces the handoff latency, resulting in a reduction of the number of lost packets during such handoff. We note that, the performance of base Mobile IP is not improved or affected by such change, since it is not aware of any RFA levels between the serving FA and the HA. The lowest average lost packets/handoff in the hierarchy case (0.09 packets/handoff) is observed for RFA levels = 2, where all hierarchy levels are part of the FA tree.

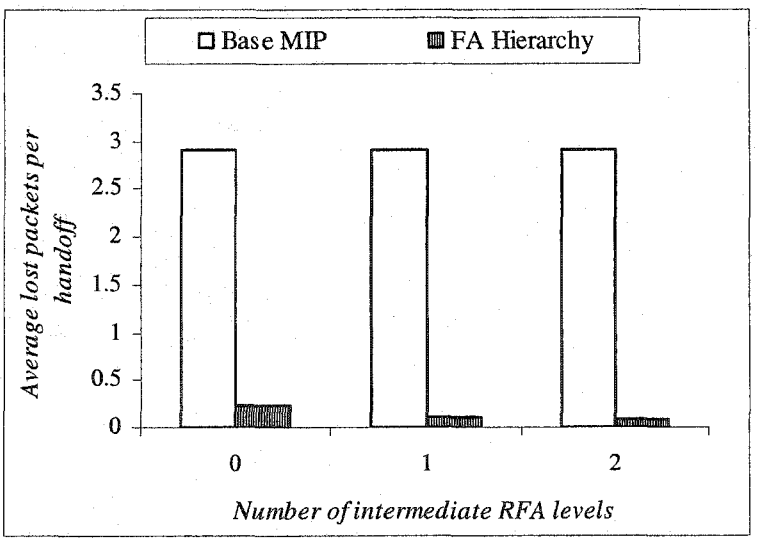


Fig. 53. Average lost packets per handoff for hierarchy height 4.

Fig. 54 illustrates the RFA regional registration replies distribution classified as GFA replies (3-hop RFA replies), 2-hop RFA replies, or 1-hop RFA replies. An increase in the number of intermediate RFA levels implies an increase in the number of i-hop handoffs for $i = 1, 2$ below the GFA, and a reduction in the number of GFA replies. Consequently, the number of control messages (registration requests), reaching the GFA, is reduced, hence making the hierarchy links leading to the GFA less congested. In addition, bringing the crossover FA closer to the serving FA reduces the handoff latency and consequently the number of lost packets.

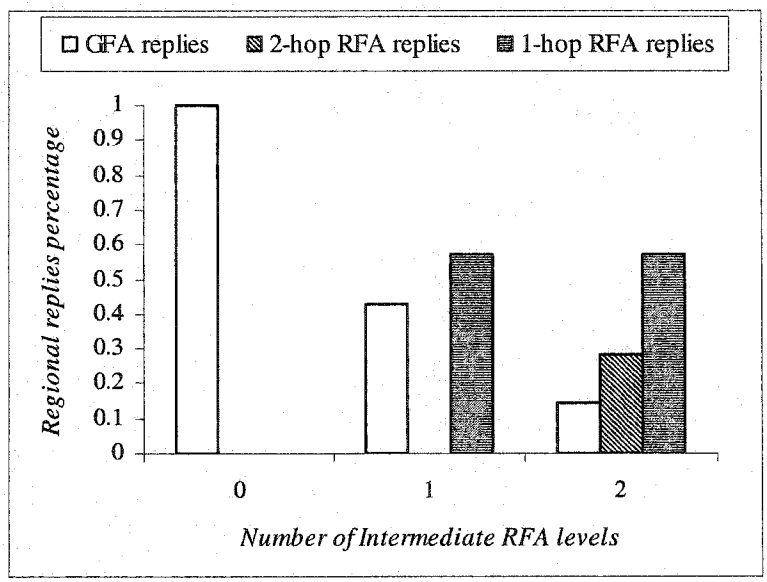


Fig. 54. Distribution of regional registration replies for hierarchy height 4.

Fig. 55 and Fig. 56 show simulation results for hierarchy height 5. We obtain similar results of significant improvements by adopting an FA hierarchy approach (92% reduction in packet loss even without any RFA levels), and a decrease in average lost packets per handoff by increasing the number of intermediate RFA levels (ranging from 0.23 packets/handoff for zero RFA levels to 0.09 packets/handoff for RFA levels =3.

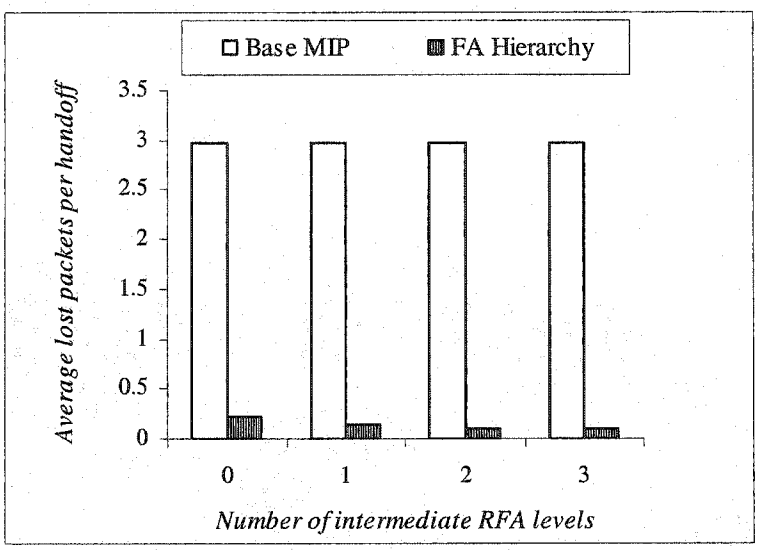


Fig. 55. Average lost packets per handoff for hierarchy height 5.

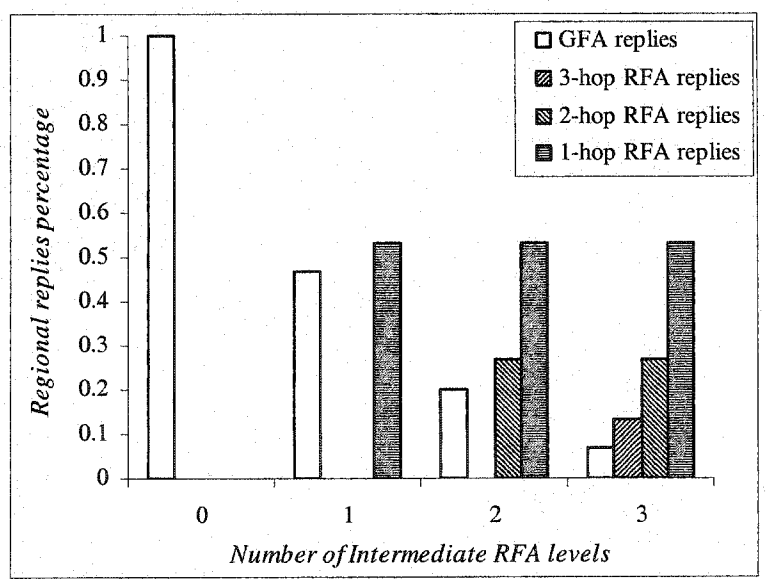


Fig. 56. Distribution of regional registration replies for hierarchy height 5.

Fig. 57 depicts the average number of lost packets for hierarchy height 6. The same results pattern similar to heights 4 and 5 results can be observed. With no RFA levels, the hierarchy approach achieves 89% reduction in packet loss over base Mobile IP. The increase in number of RFA levels results in a decrease in the average lost packets/handoff (ranging from 0.33 packet/handoff for no RFA levels to 0.13 packets/handoff for RFA levels = 4). It is expected that the resulting regional registration replies' distribution exhibits the same pattern as previously reported for heights 4 and 5 (Fig. 54, and Fig. 56). With no RFA levels, all handoffs will be 5-hop handoffs with the GFA being the crossover FA. The increase in number of RFA levels substitutes some of the i-hop handoffs with j-hop handoffs, where $j < i$, consequently reducing the average handoff latency and the average packet loss per handoff.

Comparing the average packet loss for hierarchy heights 4, 5, and 6 (Fig. 58), hierarchy height 4 achieves the lowest average lost packets/handoff with heights 5 and 6 recording a slight increase for all choices of RFA levels (the maximum number of RFA levels for hierarchy height n is n-2). With the maximum number of RFA levels, an increase in hierarchy height implies an increase in tunneling overhead within the FA hierarchy (because of the added RFA levels), and an increase in data packets latency. Such latency increase is also dependent on the hierarchy link delay between RFA levels (set to 0.5 ms in this experiment). In addition, the increase in hierarchy height implies an increase in average handoff latency for the adopted mobility pattern resulting in an increase in average lost packets/handoff. However, no major irregularities are observed with the hierarchy height increase, hence arbitrary n-level hierarchy designs can be adopted at the expense of average packet loss increase for some mobility patterns compared to a smaller choice of hierarchy height n[26].

[26] If the hierarchy height increase does not affect the type of experienced handoffs, e.g., the MH's handoffs were all 1-hop handoffs; the average packet loss would not be affected. Hence, the mobility pattern is an important factor in shaping the experienced packet loss.

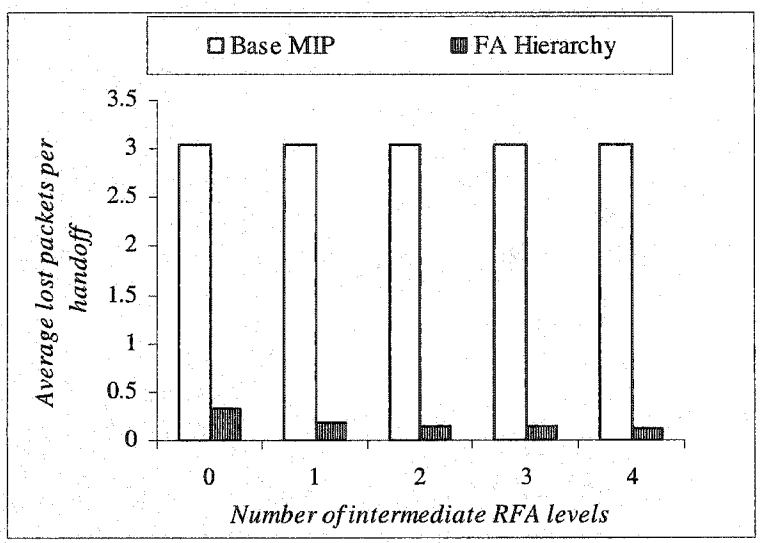


Fig. 57. Average lost packets per handoff for hierarchy height 6.

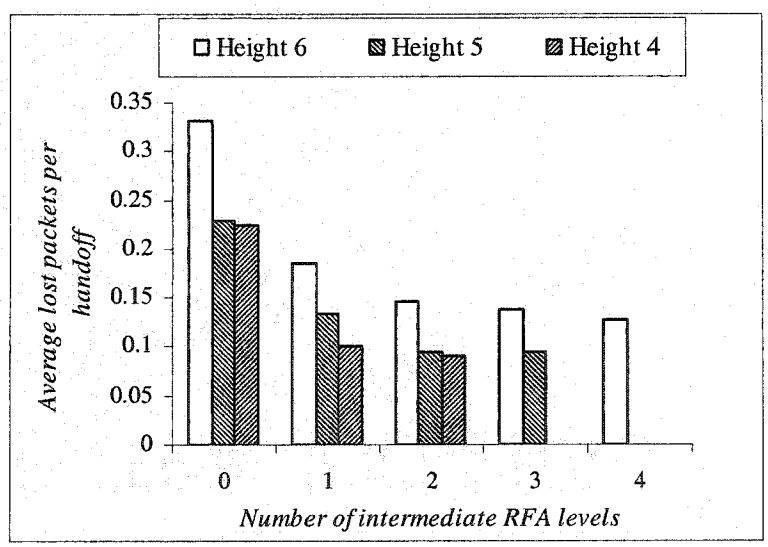


Fig. 58. Average lost packets per handoff for hierarchy heights 4, 5, and 6.

6.2.2 TCP Traffic

We investigate the effect of the number of intermediate RFA levels on TCP throughput (TCP Tahoe [60]). We use a 4-level hierarchy and set up a long-term FTP session between the CH and the MH, where the MH is downloading a large file from the CH. We measure the application-level TCP throughput as seen by the MH at the end of the simulation. The MH is moving back and forth across the domain at a speed of 20 m/s.

Fig. 59 depicts TCP throughput (Mbps) versus the number of intermediate RFA levels. Note that, TCP throughput is bounded by the bandwidth of the bottleneck link (1.5 Mbps). With no RFA levels, the FA hierarchy approach achieves slightly higher TCP throughput than base Mobile IP. The FA hierarchy TCP throughput increases with the increase in number of RFA levels, while base Mobile IP throughput is not affected. The throughput increase is attributed to a decrease in average handoff latency. With Base Mobile IP, TCP is able to cope with packet loss by maintaining a higher packet retransmission ratio[27] (1%) compared to the FA hierarchy's case (average of 0.3%).

TCP throughput is calculated as the application-level total received bytes over the FTP session duration to yield a bytes/s measure. Such measure does not reveal the instantaneous throughput distribution during the FTP session. Therefore, we calculate the instantaneous throughput every 2 ms during the FTP session. We partition the throughput spectrum into m bandwidth intervals, and calculate the frequency $freq(B_i)$ for i:1→m defined as the ratio of the number of times an instantaneous throughput value is within bandwidth interval B_i to the total number of throughput values. Fig. 60 depicts the instantaneous TCP throughput intervals frequency during this experiment. For bandwidth interval [1.4, 1.5] (Mbps), Base Mobile IP records a frequency of 71%, while the FA hierarchy with no RFA levels records 84%, with a highest frequency of 88% attained by 2 RFA levels. Hence, the FA hierarchy approach is capable of achieving a high instantaneous throughput value more frequently than the base approach. We note that, both base Mobile IP and the hierarchy approach experience some low throughput values in the range [0, 0.2] (Mbps). As previously explained in section III, this is attributed to

[27] As introduced in section III, the retransmission ratio is the number of retransmitted packets to the total number of transmitted packets.

batch packet losses due to the poor MH's BS selection process currently implemented in *ns* (see section 3.5.2), and the resulting TCP's exponential back off feature [60].

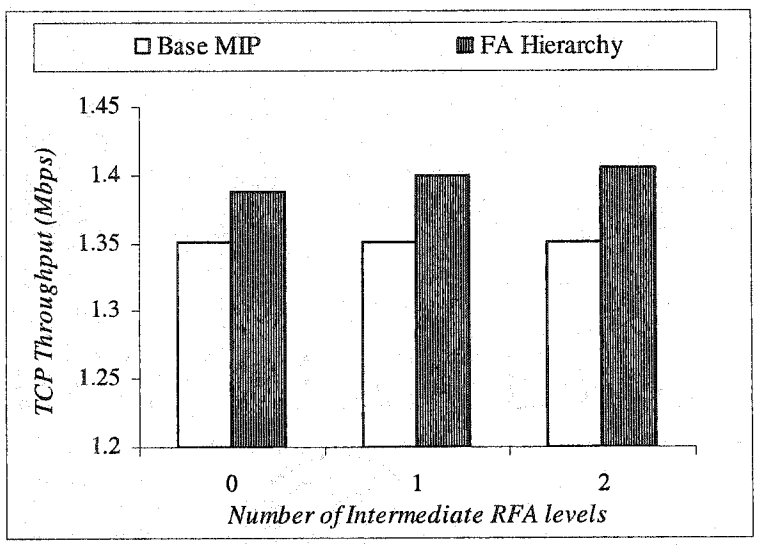


Fig. 59. TCP Throughput versus the number of intermediate RFA levels.

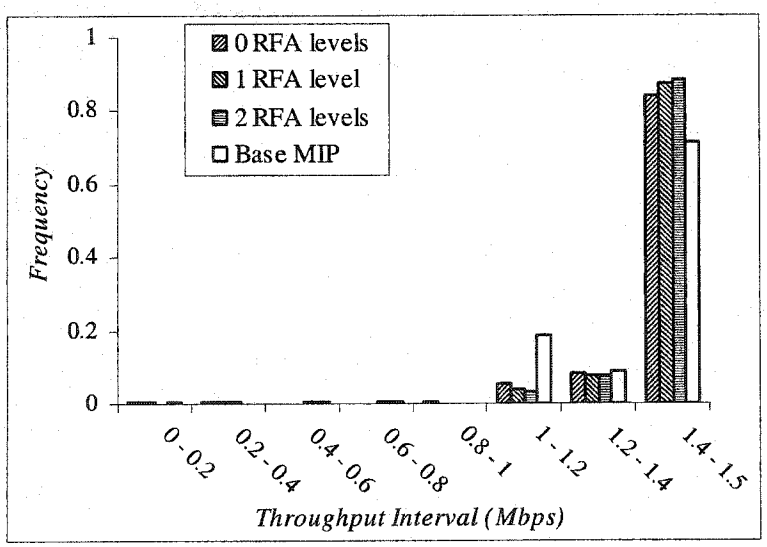


Fig. 60. Instantaneous TCP throughput intervals frequency with varying RFA levels.

6.3 Hierarchy Link Delay

We investigate the effect of the hierarchy link delay $LD_{FA\text{-}FA}$. We use a 4-level hierarchy with 2 intermediate RFA levels (hierarchy configuration "4/2/3"). The MH is periodically moving between 2 foreign agents to create only 3-hop handoffs at a speed of 10 m/s. In addition, we repeat the experiment when the MH periodical movement generates only 2-hop handoffs. Constant bit rate UDP traffic is applied between the CH and the MH and we measure the average lost packets per handoff at the MH. Our expected result is that 2-hop handoffs should produce less average lost packets than the 3-hop handoff case. Recall that we demonstrated in section III, that the increase of hierarchy link delay resulted in an increase of average lost packets per handoff, while enforcing a significant number of HR-LH requests (see section 3.5.1.2).

Fig. 61 shows the average lost packets per handoff versus the FA hierarchy link delay. As expected, the increase of hierarchy link delay yields an increase in average lost packets per handoff for the investigated cases. The average lost packets in the 3-hop handoff case is consistently higher than the 2-hop handoffs case.

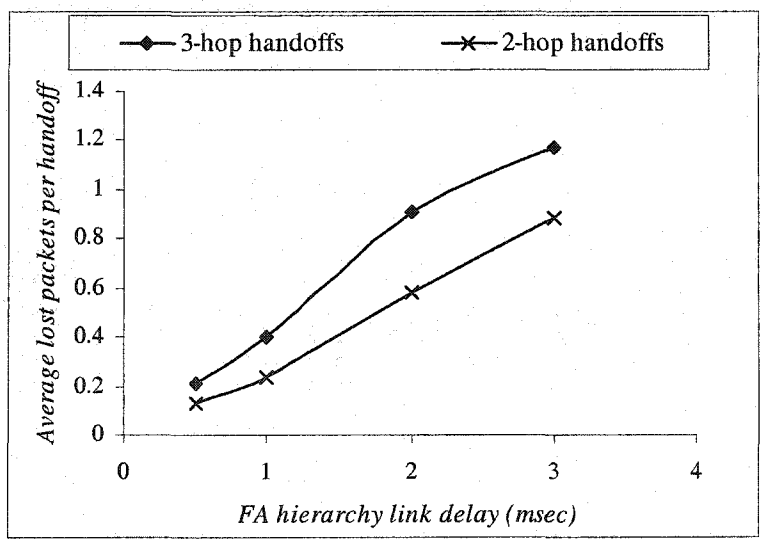


Fig. 61. Effect of FA hierarchy link delay.

6.4 Link Delay between GFA and HA

We investigate the effect of the link delay between the GFA and the HA ($LD_{GFA\text{-}HA}$). We use a "4/2/3" hierarchy configuration where the MH is periodically moving across the entire foreign domain at a speed of 20 m/s until a desired number of handoffs is achieved. We compare base Mobile IP and the hierarchy approach, and experiment with both UDP and TCP traffic. Measurements include the average lost packets per handoff, and application-level throughput, for UDP and TCP traffic, respectively.

6.4.1 UDP Traffic

Fig. 62 depicts the average lost packets per handoff while increasing the link delay between the GFA and HA. As expected, the FA hierarchy approach achieves substantial reduction in packet loss compared to base Mobile IP with the link delay increase. In addition, the average lost packets per handoff are not adversely affected by the link delay increase. Hence, the FA hierarchy approach truly localizes the handoff effect, reducing the lost packets per handoff.

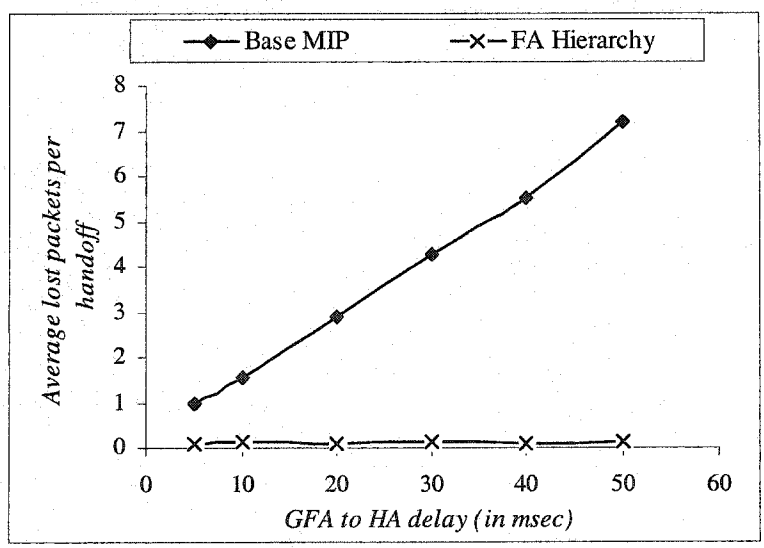


Fig. 62. Effect of $LD_{GFA\text{-}HA}$ on the average lost packets per handoff.

6.4.2 TCP Traffic

Fig. 63 shows the application-level TCP throughput for base Mobile IP and the FA hierarchy approach while increasing $LD_{GFA\text{-}HA}$. As expected, the throughput decreases with the link delay increase for both approaches. However, the FA hierarchy approach exhibits lower throughput reduction with the link delay increase. For instance, at $LD_{GFA\text{-}HA}$ = 50 ms, the hierarchy TCP throughput was reduced around 8% compared to its throughput value at $LD_{GFA\text{-}HA}$ = 5 ms, while base MIP throughput reduction was around 18%. In addition, at 50 ms, the FA hierarchy approach records a throughput higher than the base approach by 13%.

Fig. 64 illustrates the instantaneous TCP throughput intervals frequency during this experiment. The graph only shows the cases for $LD_{GFA\text{-}HA}$ equal to 5, and 50 ms for both approaches. At high link delay (50 ms), neither approaches record an instantaneous throughput value in the [1.4, 1.5] range. The hierarchy approach records a frequency of 94 % in the [1.2, 1.4] range versus a 70 % frequency for the base approach. For small link delay (5 ms), both approaches' results are comparable in terms of long-term throughput, and instantaneous throughput intervals frequency.

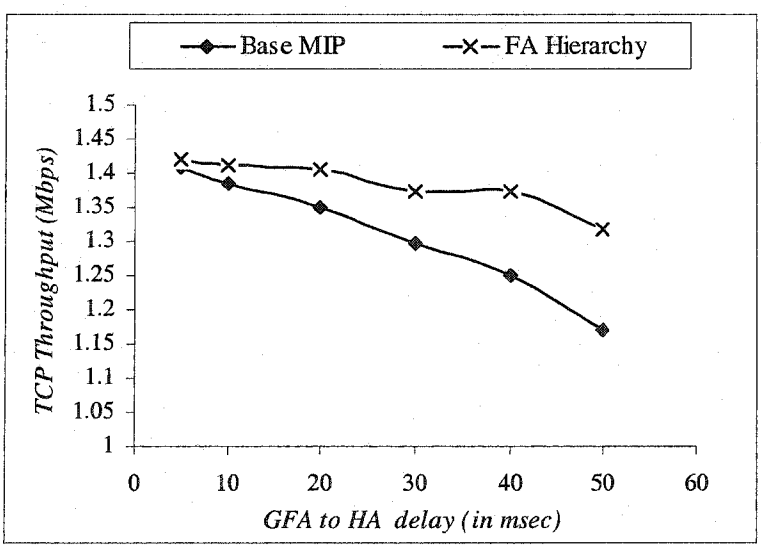


Fig. 63. Effect of $LD_{GFA\text{-}HA}$ on TCP throughput.

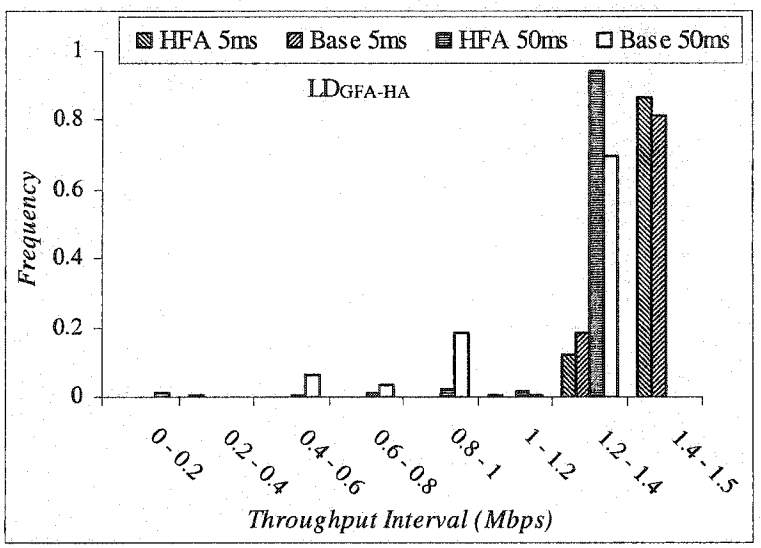


Fig. 64. Instantaneous TCP throughput intervals frequency while varying $LD_{GFA\text{-}HA}$.

6.5 FA Hierarchy Topology and Smooth Handoff

We investigate the effect of the smooth handoff functionality, while experimenting with a number of network topologies for the FA hierarchy as follows.

- Each FA and its immediate children are connected through individual duplex links (section 6.5.1).
- All hierarchy nodes are located in one 10 Mbps subnet (section 6.5.2).
- Each FA and its immediate children are in an independent 10 Mbps subnet, i.e., the FA hierarchy is composed of multiple FA subnets (section 6.5.3).

We use a perfect 4-level binary tree for the FA hierarchy, with a default "4/2/3" configuration. For each topology, we apply constant bit-rate UDP traffic between the CH and the MH, and calculate the average number of lost packets per handoff at the MH. The MH is periodically moving back and forth between 2 foreign agents at a speed of 20 m/s to produce a desired number of handoffs. The 2 foreign agents are selected to produce 3-hop and 2-hop handoffs in the individual duplex links topology, and later reused with the other two topologies. The experiment is repeated where the MH is using the smooth

handoff functionality by supplying a PFANE along with its registration request to the new FA [47]. The default smooth handoff BU lifetime is 0.5 s.

6.5.1 Individual Duplex Links Topology

Fig. 65 illustrates the average lost packets per handoff while increasing the FA hierarchy link delay. Moreover, Fig. 66 shows the same measure when smooth handoff is used. We use two separate figures to present the results instead of one since the smooth handoff results almost coincide with the normal case. This is attributed to the nature of the duplex links topology. Recall that base stations do not provide buffering capabilities to the MH. The routing path for the smooth handoff binding update message from the old FA to the new FA is the same path taken by the MH's registration request until it reaches the crossover FA, and then by the deregistration message propagated from the crossover FA up to the old FA (Fig. 67). The smooth handoff BU does not reach the old FA early enough to forward any future packets to the new FA. Hence, no evident benefits can be gained by using the smooth handoff in this particular case. Such experiment further shows that a buffering capability in base stations would be indeed useful [49].

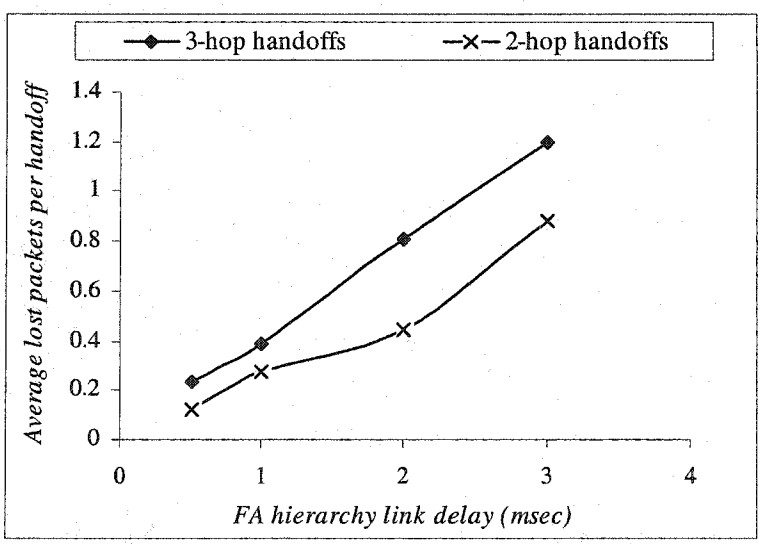


Fig. 65. Average lost packets in the duplex links topology.

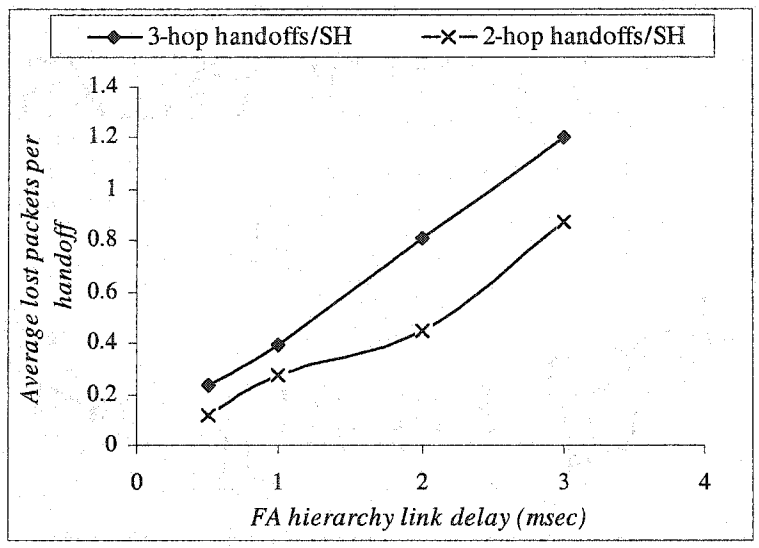
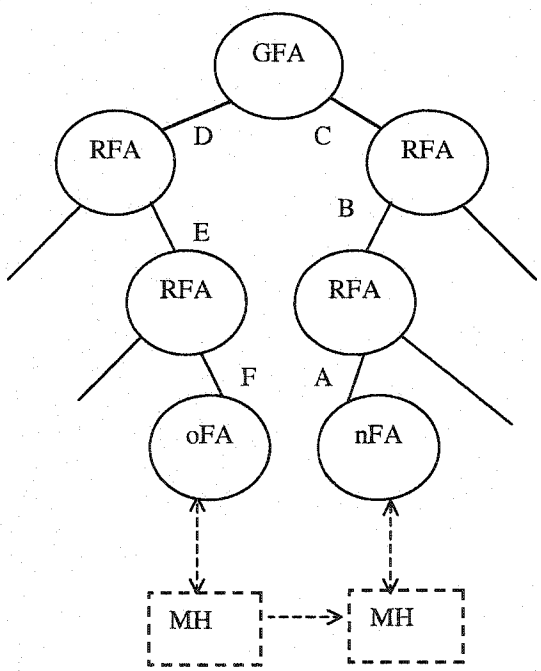


Fig. 66. Average lost packets in the duplex link topology with smooth handoff.



The smooth handoff BU from nFA to oFA traverses links {A, B, C, D, E, and F} in that order.

The MH's Registration request traverses links {A, B, and C} in that order in route to the crossover FA.

The Deregistration message traverses links {D, E, and F} in that order.

Fig. 67. 3-hop handoffs and smooth handoff in the duplex links topology.

6.5.2 One-subnet Topology

We repeat the same experiment when the FA hierarchy is located in one 10 Mbps subnet. Thus, any FA is only one hop away from any other FA in the hierarchy. We use a "4/2/3" hierarchy configuration. In this case, although the network topology is one subnet, the FA hierarchy levels are logically overlaid over the physical network topology.

Fig. 68 illustrates the average lost packets for 3-hop, 2-hop handoffs, and the corresponding smooth handoff cases while increasing the subnet delay. As expected, the smooth handoff is beneficial in this case reducing the average lost packets per handoff. Such result can be attributed to the fact that the smooth handoff BU message from the new FA to the old FA only takes 1-subnet delay to actually reach the old FA, allowing ample time to forward received packets to the new FA until the crossover FA switches the MH's tunneling path to the new path. The 3-hop handoffs record a higher average than the 2 hop handoffs, while the smooth handoff results are almost the same for the two investigated handoff types because of the 1-subnet delay between any two foreign agents.

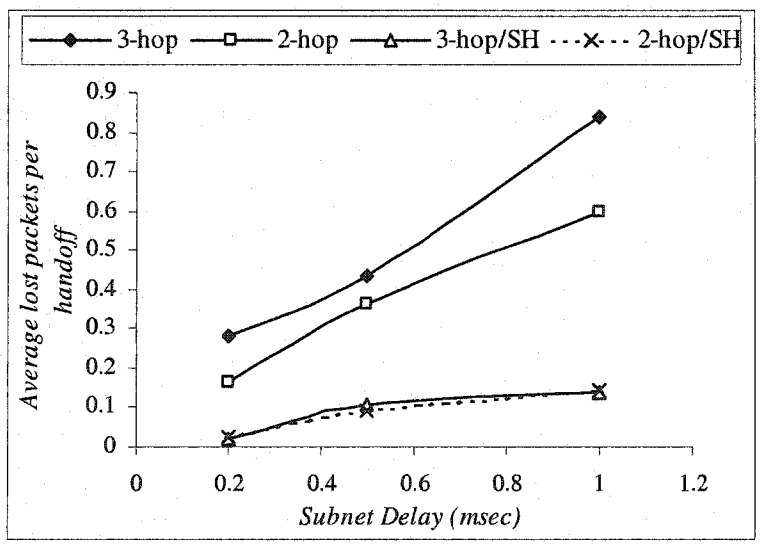


Fig. 68. Average lost packets in the 1-subnet topology for a "4/2/3" configuration.

Fig. 69 illustrates the results of the same experiment for a "4/2/1" hierarchy configuration. In this case, the GFA is the crossover FA for all handoffs and is 1-subnet delay (1-hop) away from any BS. As expected, the observed packet losses without smooth handoff are less than the "4/2/3" hierarchy configuration packet losses (Fig. 68).

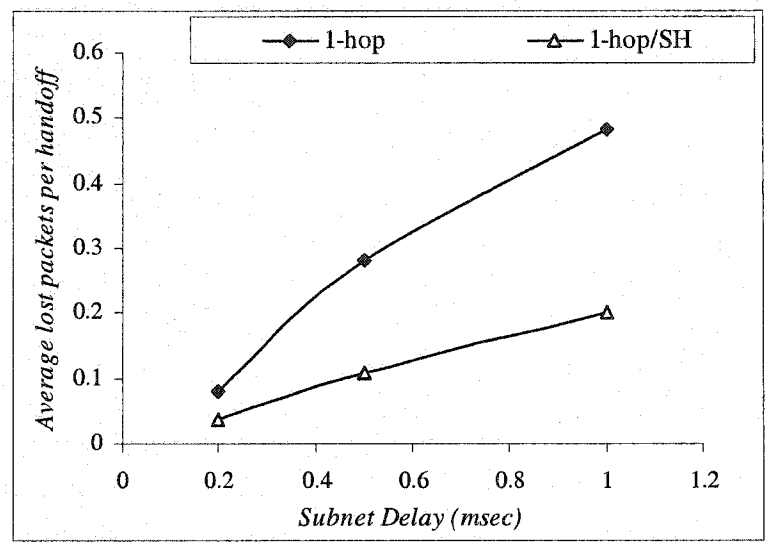


Fig. 69. Average lost packets in the 1-subnet topology for a "4/2/1" configuration.

6.5.3 Multiple Subnets Topology

The experiment is repeated for a multiple 10 Mbps subnets topology using a "4/2/3" hierarchy configuration. In this case, each FA and its immediate children FAs are located in an independent subnet. The same subnet delay is used for all subnets. The smooth handoff BU message for an i-hop handoff reaches the old FA after $2*i - 1$ subnet delays, while the deregistration message reaches the old FA after $2*i$ subnet delays in addition to any processing delay by each intermediate regional foreign agent. Thus, a period of time is available for the old FA to forward any newly received packets to the new FA.

Fig. 70 shows the average lost packets per handoff while increasing the subnet delay. The usage of the smooth handoff mechanism lowers the average number of lost packets for the investigated handoffs, but the reduction is lower than achieved in the 1-subnet topology with a similar configuration (Fig. 68).

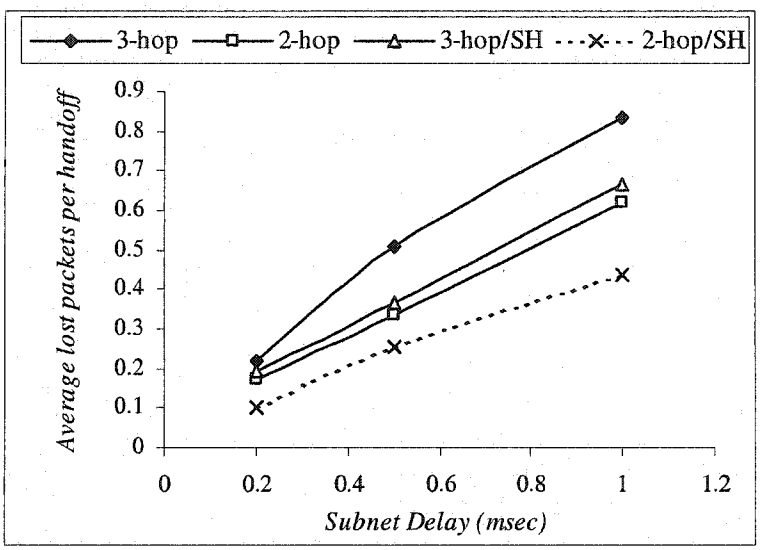


Fig. 70. Average lost packets in the multiple subnets topology for a "4/2/3" configuration.

6.6 Conclusion

In section VI, we presented a suite of simulation validation experiments, while investigating the effect of some network design parameters such as FA hierarchy height and topology. Simulation results have shown that FA hierarchies can be constructed using an arbitrary number of foreign agent levels at the expense of increased average packet loss function of the delay between RFA levels and dependent on the MH's mobility pattern. In addition, we have demonstrated that increasing the number of RFA levels below the GFA while fixing the hierarchy height, i.e., reducing network hops

between the new FA and the crossover FA, reduces average packet loss, and increases TCP throughput. Moreover, the smooth handoff mechanism has been shown to be a viable approach to reduce packet loss, depending on hierarchy topology and availability of buffering capabilities in base stations.

SECTION VII

CONCLUSION AND FUTURE EXTENSIONS

In this section, we conclude this dissertation by summarizing our motivations, objectives, contributions, and the performance of our proposed local-area mobility support framework. Furthermore, we discuss a list of possible future extensions to our work in the context of the local-area mobility support framework, and the corresponding network simulation framework.

7.1 Conclusion

Host mobility is quickly becoming the norm rather than the exception. However, host mobility scenarios break the operation of a TCP/IP based Internet, which relies on a host's IP address to act as a network-layer routing directive and a transport-layer connection identifier. Mobile IP for IPv4, standardized as a network-layer host mobility problem solution [43], deploys mobility agents in the home and foreign domain. Such mobility agents provide the solution to the host mobility problem through a home registration mechanism, in which the home agent is informed about the current care-of address of the MH (the address of a foreign mobility agent, or a co-located care-of address). However, the home registration requirement, upon change of point of attachment (change of care-of address, or handoff), results in considerable registration signaling overhead with the expected increase in the number of mobile hosts. In addition, if every MH's handoff is processed by a possibly distant HA, the handoff latency increases, resulting in increased packet loss and disrupting effects on the MH's application and services. Researchers identified such problem, and adopted Mobile IP as a wide-area mobility solution, while suggesting alternative solutions to handle local-area mobility. Local-area mobility solutions normally operate within the bounds of a foreign domain, and attempt to minimize the resulting home registration signaling by localizing the effect of a MH's handoff. In such manner, a HA is not involved in managing every MH's local movement and handoff, resulting in local handoff processing and lower handoff latencies.

We classified local-area mobility solutions as belonging to the following categories: Mobile IP extensions, host-based forwarding schemes, and multicast-based schemes. The solutions in the Mobile IP extensions category attempt to adhere to Mobile IP's registration framework between mobility agents, while localizing the handoff processing. One such solution is the regional registration approach (MIP_RR) that extends the notion and functionality of a foreign agent through the deployment of a foreign agent hierarchy in the foreign domain and the introduction of a regional registration mechanism used to process local-handoffs [30]. The foreign agents' hierarchy acts as a mobility support overlay network, in which packets destined to the MH are tunneled by the GFA (The hierarchy root is the Gateway Foreign Agent) to subsequent hierarchy levels (regional FAs) until received by the MH (through a leaf FA). We presented an overview and modeling of the MIP_RR approach identifying the necessity of a mechanism we termed a "hierarchy tunneling consistency mechanism," ensuring the removal of a RFA's visitor entries on a MH's old route, hence guaranteeing the integrity of future registration processing by such RFAs.

In this dissertation, we presented a local-area mobility framework based on the deployment of multiple cooperating foreign agent hierarchies in the foreign domain. Our framework is independent of any access technology, relying on generic mobility agents' functionality and registration processing mechanisms adopted by base Mobile IP, and MIP_RR. However, our introduced framework and signaling design alleviate several identified deficiencies within the MIP_RR approach, listed as follows.

- The complete reliance on the smooth handoff mechanism to implement the tunneling consistency mechanism, hence denying service to some MHs (or legacy MHs) that do not support the smooth handoff mechanism.
- The existence of a potential race condition in its tunneling consistency mechanism. Such race condition stems from allowing the registration request and the corresponding tunneling consistency mechanism to proceed in parallel on the old and new path towards a crossover FA. Such race condition might lead to inconsistencies in registration lifetimes and erroneous removal of visitor entries maintained by RFAs for the MH.

- The lack of a scalable registration framework to manage inter-hierarchy handoffs. The proposed registration mechanisms for such handoffs either require the initiation of a home registration along with incurred long processing delays, or the initiation of a regional registration requiring the set up of a large number of security associations between foreign agents in different hierarchies, which might not be feasible or acceptable. Moreover, the latter solution does not scale well with an increased number of foreign agent hierarchies, or number of foreign agents within a hierarchy.

The following issues were addressed in the context of this dissertation while designing the local-area mobility support solution: foreign agent hierarchy model, registration processing for intra-hierarchy handoffs, registration processing for inter-hierarchy handoffs within the same foreign domain, and performance evaluation.

Foreign agent hierarchy model

We suggested a foreign agent hierarchy model with a backward compatible mode that enables the service of legacy MHs not capable of processing local-area mobility extensions. A leaf FA advertises its IP address if not private, along with the GFA IP address. Regional FA addresses between the leaf FA and the GFA are not advertised in order to hide the hierarchy structure from visiting MHs, and to reduce the agent advertisements over a wireless link. Legacy MHs register the agent's IP address as their care-of address, while enabled MHs use the GFA address to benefit from the mobility extensions.

Registration processing for intra-hierarchy handoffs

We identified possible handoff scenarios associated with home and regional registrations: home registrations not involving local-handoffs, home registrations involving local-handoffs (HR-LH), and regional handoffs associated with regional registrations. We proposed a regional registration framework with an associated tunneling consistency mechanism that alleviates the aforementioned drawbacks in MIP_RR. In brief, the crossover FA is responsible for initiating the tunneling consistency

mechanism to clear the visitor entries on the MH's old hierarchy path to avoid the reliance on the smooth handoff mechanism. In addition, message acknowledgments are required between involved RFAs to ensure the proper operation of the tunneling consistency mechanism. Moreover, we suggested an identification value dissemination mechanism as part of a replay protection support mechanism for MHs, when the MH is using either timestamp or nonce replay protection. The crossover FA sends a replay protection update message upwards in the hierarchy towards the GFA, in order to update the identification value associated with the MH. Such mechanism ensures future successful registration processing by upper levels in the hierarchy.

For HR-LH scenarios, we proposed 2 registration processing mechanisms that ensure tunneling of data packets to the MH while its home registration request is in transit to the HA, and until a home registration reply is received establishing a new path to the MH. The first approach, termed KOPA, attempts to maintain the old path to the MH "alive" until the new path is established. KOPA relies on replacing visitor entries in the old path by binding cache entries with an estimated lifetime, and informing the old FA about the new FA without relying on the smooth handoff mechanism. The binding cache lifetime is calculated based on observed home registration latencies, and the remaining registration lifetime at the crossover FA. The second approach, termed SINP, adopts a proactive approach in immediately switching the tunneling of data packets from the old path to the new path, without waiting for a home registration reply. The crossover FA generates a regional registration reply in response to a specially formulated home registration request, while simultaneously clearing visitor entries on the old path. Either approaches required the introduction of a local-replay protection extension that conveys current regional registration identification information, and a local care-of address extension that conveys the new FA IP address to eventually reach the old FA.

We evaluated the performance of our proposed registration frameworks for intra-hierarchy handoffs using our extension of the Columbia IP Micro-mobility Software (CIMS) [15], which is a network simulator *ns*-2 source code extension [40], implementing a suite of local-area mobility protocols. Network simulation results have demonstrated the effectiveness of our framework versus a base Mobile IP implementation in reducing UDP packet loss (96% packet loss reduction without relying on a smooth

handoff mechanism) and achieving better TCP throughput (34% throughput increase over base MIP) in the case of a distant HA. In addition, for TCP streams, results have shown that base MIP increases the number of retransmitted packets (a retransmission ratio of 5%) to combat increased packet loss due to increased round trip times between a MH and a CH, while our framework is capable of maintaining a negligible retransmission ratio and achieving better TCP throughput values. Moreover, network simulation of the KOPA and SINP approaches have shown that the SINP approach would need to maintain a smaller playout delay in order to achieve zero packet loss, due to not relying on tunneling data packets from the old FA to the new FA for an extended period of time as adopted by the KOPA approach.

Registration processing for inter-hierarchy handoffs

For inter-hierarchy handoffs, we proposed a configurable scalable cooperation framework between deployed foreign agents hierarchies. Such cooperation framework attempts not to change the home-registered care-of address unless deemed necessary by an involved GFA or the MH. Cooperation in registration processing is only allowed between the roots of the FA hierarchies, hence reducing the number of required security associations between FAs in different hierarchies, and scaling with an increased number of such hierarchies. Configurable cooperation is achieved by altering the agent advertisements to include cooperation initiation and acceptance options between GFAs. In addition, the proposed framework attempts to cope with the failure of the current care-of address by introducing a specially formulated home registration message. Such message enables a current GFA to forward the MH's registration request to the HA upon detecting that the GFA, acting as the MH's care-of address, is not reachable.

We evaluated the performance of the proposed cooperation framework for inter-hierarchy handoffs using the aforementioned network simulator extension. For UDP traffic, when compared against a non-cooperative approach for a distant HA, the cooperation approach between GFAs achieves 87% and 90% packet loss reduction for audio and video traffic, respectively. For TCP traffic, the cooperation approach records 5% higher throughput values, with the non-cooperative counterpart maintaining a

retransmission ratio of 5% in order to achieve comparable throughput values (compared to a retransmission ratio of 0.7% for the cooperation approach).

Performance evaluation

The network simulation experiments were performed using an extension of CIMS. We designed and developed a network simulation framework for local-area mobility. We added to CIMS the capability of modeling a true foreign domain with multiple foreign agent hierarchies, each with an arbitrary number of levels and various topologies. We automated the creation of FA hierarchies modeled as perfect trees, with FA levels constituting an overlay tree superimposed on the tree topology. In addition, we incorporated a number of enhancements including the support for periodical home registrations, regional registrations, and the smooth handoff mechanism. In the future, we plan to contribute our extension to CIMS and *ns*-2 distributions. Moreover, we used our extension to conduct a number of validation network simulations to investigate the effects of various factors such as hierarchy height, topology, and usage of the smooth handoff mechanism. We concluded that bringing the crossover FA closer to the MH reduces packet loss, and the hierarchy network topology plays an important role in determining whether the smooth handoff mechanism is effective in reducing packet loss.

In conclusion, this dissertation encompasses two major contributions: a local-area mobility support framework deploying multiple cooperating FA hierarchies in the foreign domain, and a network simulation framework for local-area mobility. The mobility support framework introduces a backward compatible FA hierarchy model, enhanced regional registration processing and new home registration processing for intra-hierarchy handoffs, a cooperation framework for registration processing for inter-hierarchy handoffs. The network simulation framework allows simulating a foreign domain comprised of a multiple FA hierarchies, with an arbitrary number of levels, and implements our proposed mobility support framework.

7.2 Future Extensions

The work in this dissertation can be extended in two orthogonal directions: extending the local-area mobility support framework, and extending the network simulation framework.

Local-area Mobility Support Framework Extension

The local-area mobility support framework can be extended as follows.

- Extend the registration processing frameworks, when the FA hierarchy is actually a forest instead of a tree, i.e., a FA can have a number of parent FAs instead of one parent FA. Such task entails identifying which parent FA is forwarded a MH's registration request in route to the GFA. A related issue is the advertisement of multiple GFAs by a leaf FA, and directing the MH's request to the appropriate GFA
- Optimize the case of inter-domain handoffs. An inter-domain handoff would normally trigger a home registration to establish a new care-of address in the new foreign domain. Can cooperation be enabled between FA hierarchies in multiple domains? Alternatively, one can rely on the smooth handoff mechanism to reduce packet loss, until the home registration reply is received.
- Investigate a protocol to dynamically set up FA hierarchies. For instance, Malinen [37] suggests configuring each FA with its parent FA. When an FA is booted, it performs a secure registration process to inform its parent FA of its availability. We envision introducing a hierarchy registry component within the foreign domain that is responsible for dynamically reorganizing FA hierarchies within the foreign domain. The hierarchy registry and FAs would be members of some control multicast group to exchange status and control messages.
- Investigate techniques to improve the fault-tolerance of the registration of the FA hierarchy. We assume the existence of a recovery protocol that allows an RFA to recover its visitor entries after a failure or restart. Such protocol can be executed on startup so that an RFA queries its children FAs about any

current visiting MHs, hence restoring its tunnel endpoint for each involved MH.

- Use the introduced CIMS extension to compare the FA hierarchy approach, HAWAII, and Cellular IP as local-area mobility support frameworks. Recently, Campbell et al. [12] compared the aforementioned approaches while using a limited 1-level FA hierarchy below the GFA. The introduced extension allows creating *n*-level FA hierarchies.
- Establish a network testbed for the local-area mobility support framework. A good starting point is the Dynamics-Hut Mobile IP project at Helsinki University of Technology [22], which offers a user space implementation of their hierarchical Mobile IP variant [27] on Linux.

- Investigate the extension of the introduced IPv4 registration processing techniques to process binding updates in an IPv6 hierarchical mobility support solution such as [38].
- Investigate other techniques to enable a crossover FA to properly determine its crossover status for the MH's registration request. Currently, the presence of a MH's visitor entry is the deciding factor for such crossover FA. Alternatively, Malinen [37] suggests that each RFA is informed about the NAI of all FAs below it in the hierarchy, and that a MH supplies the NAI of the previous FA as part of its registration request. If the previous FA is part of the subtree under this RFA, no upper RFAs need to be informed about such registration request, and this RFA is actually the crossover FA for this request.
- Leverage multicast technology in the foreign domain as a means for data packets delivery to a MH, where the MH is assigned a unique multicast group within the foreign domain, e.g., [32]. Such task entails investigating a number of related multicast issues including multicast group allocation, and multicast routing for a large number of multicast groups with a small number of participants.

Network Simulation Framework Extension

The network simulation framework can be extended as follows.

- Introduce the notion of multiple foreign domains and handoffs between domains. Such task entails that a foreign agent advertises its network address identified (NAI), to aid the MH in identifying handoffs between domains.
- Incorporate security mechanisms related overhead in the reported simulation results. Such task entails adding protocols to set up security associations, generating an authenticated message, ensuring that a message is properly authenticated, and generating registration keys.
- Automate other base station placement techniques in a two-dimensional grid. For instance, Perkins and Wang [49] arrange base stations in the foreign domain to constitute a rectangle.
- Automate the generation of network link orientations for *n*-level, *N*-ary perfect trees to allow visualizing output network trace files using the network animator tool *nam* [26].
- Develop an *FA hierarchy editor* to aid in the design of FA hierarchies in the foreign domain. Such tool would allow placing RFAs in a domain using drag and drop approaches, specifying the type of connectivity between hierarchy levels, and specifying the placement and overlap areas of base stations. The output of such editor would be an *ns*-2 script that can be further specialized and extended by a *ns*-2 user to speed up the programming process.

REFERENCES

[1] A. Abdel-Hamid and H. Abdel-Wahab, "A Generalized Foreign Mobility Agents Architecture," Technical Report TR_2000_05, Department of Computer Science, Old Dominion University, Norfolk, VA 23529, USA, June 2000.

[2] _____, "Local-area Mobility Support through Cooperating Hierarchies of Mobile IP Foreign Agents," in *Proc. of the 6th IEEE International Symposium on Computers and Communications (ISCC'2001)*, Hammamet, Tunisia, July 2001, pp. 479-484.

[3] B. Aboba and M. Beadles, "The Network Access Identifier," Request For Comments (Proposed Standard) 2486, Internet Engineering Task Force, Jan. 1999.

[4] O. Altintas, A. Dutta, W. Chen, and H. Schulzrinne, "Mobility Approaches for All IP Wireless Networks," in *Proc. of the 6th World Multi-Conference on Systemics, Cybernetics, and Informatics (SCI'2002)*, Orlando, Florida, USA, July 14-18, 2002.

[5] A. Avancha, D. Chakraborty, D. Gada, T. Kamdar, and A. Joshi, "Fast and Effective Wireless handoff Scheme using Forwarding Pointers and Hierarchical Foreign Agents," in *Proc. of the SPIE International Symposium on Convergence of IT and Communications (ITCom)*, Denver, USA, Aug. 20-24, 2001.

[6] H. Balakrishnan, S. Seshan, and R. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," *Wireless Networks (WINET)*, vol. 1, no. 4, pp. 469-481, Dec. 1995.

[7] P. Bhagwat, C. Perkins, and S. Tripathi, "Network Layer Mobility: an Architecture and Survey," *IEEE Personal Communications*, vol. 3, no. 3, pp. 54-64, June 1996.

[8] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in Network Simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59-67, May 2000.

[9] R. Caceres and V. Padmanabhan, "Fast and Scalable Handoffs for Wireless Internetworks," in *Proc. of the 2nd IEEE/ACM International Conference on Mobile Computing and Networking (MobiCom'96)*, Rye, New York, 1996, pp. 56-66.

[10] P. Calhoun and C. Perkins, "Mobile IP Network Access Identifier Extension for IPv4," Request For Comments (Proposed Standard) 2794, Internet Engineering Task Force, Jan. 2000.

[11] A. T. Campbell and J. Gomez, "IP Micro-Mobility Protocols," *ACM Mobile Computing and Communication Review (MC^2R)*, vol. 4, no. 4, pp. 45-54, Oct. 2001.

[12] A. T. Campbell, J. Gomez, S. Kim, Z. Turanyi, C-Y. Wan, and A. Valko, "Comparison of IP Micro-Mobility Protocols," *IEEE Wireless Communications*, vol. 9, no. 1, pp. 72-82, Feb. 2002.

[13] A. Campbell, J. Gomez, C-Y. Wan, S. Kim, Z. Turanyi, and A. Valko, "Cellular IP," Internet Draft (Work in Progress), draft-ietf-mobileip-cellularip-00.txt, Jan. 2000.

[14] C. Castelluccia, "HMIPv6: A Hierarchical Mobile IPv6 Proposal," *ACM Mobile Computing and Communication Review (MC^2R)*, vol. 4, no. 1, pp. 48-60, Jan. 2000.

[15] Columbia IP Mirco-Mobility Software (CIMS). [Online]. Available: http://www.comet.columbia.edu/micromobility/.

[16] B. Crow, I. Widjaja, J. Kim, and P. Sakai, "IEEE 802.11 Wireless Local Area Networks," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 116-126, Sept. 1997.

[17] S. Das, A. Dutta, A. McAuley, A. Misra, and Sajal Das, "IDMP: An Intra-Domain Mobility Management Protocol for Next Generation Wireless Networks," *IEEE Wireless Communications*, vol. 9, no. 3, pp. 38-45, June 2002.

[18] S. Das, A. Misra, P. Agrawal, and Sajal Das, "TeleMIP: Telecommunications-Enhanced Mobile IP Architecture for Fast Intra-domain Mobility," *IEEE Personal Communications*, pp. 50-58, Aug. 2000.

[19] S. Deering, "Host Extensions for IP Multicasting," Request For Comments (Proposed Standard) 1112, Internet Engineering Task Force, Aug. 1989.

[20] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," Request For Comments (Proposed Standard) 2460, Internet Engineering Task Force, Dec. 1998.

[21] G. Dommety and T. Ye, "Local and Indirect Registration for Anchoring Handoffs," Internet Draft (Work in Progress), draft-dommety-mobileip-anchor-handoff-03.txt, July 2001.

[22] Dynamics-HUT Mobile IP Project, Helsinki University of Technology. [Online]. Available http://www.cs.hut.fi/Research/Dynamics/index.html.

[23] P. Eardley, A. Mihailovic, and T. Suihko, "A Framework for the Evaluation of IP Mobility Protocols," in *Proc. of the 11th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'2000)*, London, UK, Sept. 2000.

[24] K. El Malki and H. Soliman, "Fast Handoffs in Mobile IPv4," Internet Draft (Work in Progress), draft-elmalki-mobileip-fast-handoffs-03.txt, Sept. 2000.

[25] T. Ernst, MobiWan: A NS-2.1b6 simulation platform for Mobile IPv6 in Wide Area Networks, MobiWan User Guide, PLANETE Project, INRIA Rhone-Alpes, June 2001. [Online]. Available: http://www.inrialpes.fr/planete/mobiwan/.

[26] K. Fall and Kannan Varadhan (Editors), The *ns* Manual, The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, Feb. 2001. [Online]. Available: http://www.isi.edu/nsnam/ns/ns-documentation.html.

[27] D. Forsberg, J. T. malinen, J. K. Malinen, T. Weckstrom, and M. Tiusanen, "Distributing Mobility Agents Hierarchically under Frequent Location Updates," in *Proc. of the 6th IEEE International Workshop on Mobile Multimedia Communications (MoMuC'99)*, San Diego, CA, USA, Nov. 15-17, 1999, pp. 159-168.

[28] R. Ghai and S. Singh, "An Architecture and Communication Protocol for Picocellular Networks," *IEEE Personal Communications Magazine*, vol. 1, no. 13, pp. 36-46, 1994.

[29] M. Greiss, Tutorial for the Network Simulator *ns*. [Online]. Available http://www.isi.edu/nsnam/ns/tutorial/index.html.

[30] E. Gustafsson, A. Jonsson, and C. Perkins, "Mobile IPv4 Regional Registration," Internet Draft (Work in Progress), draft-ietf-mobileip-reg-tunnel-06.txt, Mar. 2002.

[31] S. Hanks, T. Li, D. Farinacci, and P. Traina, "Generic Routing Encapsulation (GRE)," Request For Comments (Informational) 1701, Internet Engineering Task Force, Oct. 1994.

[32] A. Helmy and M. Jaseemuddin, "Efficient Micro-Mobility using Intra-domain Multicast-based Mechanisms (M&M)," Technical Report USC-CS-TR-01, Computer Science Department, University of Southern California, Aug. 2001. [Online]. Available ftp://ftp.usc.edu/pub/csinfo/tech-reports/papers/01-747.pdf

[33] D. Johnson, C. Perkins, and J. Arkko, "IP Mobility Support for IPv6," Internet Draft (Work in Progress), draft-ietf-mobileip-ipv6-18.txt, June 2002.

[34] J. Kempf, D. Blair, P. Reynolds, and A. O'Neill, "Leveraging Fast Handover Protocols to Support Localized Mobility Management in Mobile IP," Internet Draft (Work in Progress), draft-kempf-mobileip-fastho-lmm-00.txt, June 2002.

[35] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol," Request For Comments (Proposed Standard) 2401, Internet Engineering Task Force, Nov. 1998.

[36] F. Kuo, W. Effelsberg, and J. J. Garcia-Luna-Aceves, *Multimedia Communications: Protocols and Applications*, Upper Saddle River, NJ: Prentice Hall, 1998.

[37] J. K. Malinen, "*Using Private Addresses with Hierarchical Mobile IPv4,*" Master Thesis, Department of Computer Science and Engineering, Faculty of Information Technology, Helsinki University of Technology, Sweden, Mar. 2000.

[38] J. Malinen, F. Le, and C. Perkins, "Mobile IPv6 Regional Registrations," Internet Draft (Work in Progress), draft-malinen-mobileip-regreg6-01.txt, Mar. 2001.

[39] J. Manner and M. Kojo (Editors), "Mobility Related Terminology," Internet Draft (Work in Progress), draft-ietf-seamoby-mobility-terminology-00.txt, Aug. 2002.

[40] S. McCanne and S. Floyd, *ns* Network Simulator. [Online]. Available: http://www.isi.edu/nsnam/ns/.

[41] J. Mysore and V. Bharghavan, "A New Multicasting-based Architecture for Internet Host Mobility," in *Proc. of the 3rd ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Budapest, Hungary, Sept. 26-30, 1997, pp. 161-172.

[42] H. Omar, T. Saadawi, and M. Lee, "Supporting Reduced Location Management Overhead and Fault Tolerance in Mobile IP Systems," *in Proc. of 4th IEEE Symposium on Computers and Communications (ISCC)*, Red Sea, Egypt, July 6-8, 1999, pp. 347-353.

[43] C. Perkins (Editor), "IP Mobility Support for IPv4," Request for Comments (Proposed Standard) 3344, Internet Engineering Task Force, Aug. 2002.

[44] C. Perkins, *Mobile IP: Design Principles and Practices*, Addison-Wesley Wireless Communications Series, Reading, MA: Addison Wesley, 1998.

[45] _____, "Minimal Encapsulation within IP," Request For Comments (Proposed Standard) 2004, Internet Engineering Task Force, Oct. 1996.

[46] C. Perkins and P. Calhoun, "Mobile IPv4 Challenge/Response Extensions," Request For Comments (Proposed Standard) 3012, Internet Engineering Task Force, Nov. 2000.

[47] C. Perkins and D. Johnson, "Route Optimization in Mobile IP," Internet Draft (Work in Progress), draft-ietf-mobileip-optim-11.txt, Sept. 2001.

[48] C. Perkins, D. Johnson, and N. Asokan, "Registration Keys for Route Optimization," Internet Draft (Work in Progress), draft-ietf-mobileip-regkey-03.txt, July 2000.

[49] C. Perkins and K-Y. Wang, "Optimized Smooth Handoffs in Mobile IP," in *Proc. of the 4th IEEE Symposium on Computers and Communications (ISCC'99)*, Red Sea, Egypt, July 6-8, 1999, pp. 340-346.

[50] J. Postel, "Internet Protocol," Request for Comments 791, DARPA Internet Program Protocol Specification, Sep. 1981.

[51] _____, "Internet Control Message Protocol," Request For Comments 792, DARPA Internet Program Protocol Specification, Sep. 1981.

[52] B. R. Preiss, *Data Structures and Algorithms with Object-Oriented Design Patterns in Java*, John Wiley and Sons, 1999. [Online]. Available http://www.brpreiss.com/books/opus5/html/book.html.

[53] R. Ramjee, T. La Porta, and L. Li, "Paging Support for IP Mobility using HAWAII," Internet Draft (Work in Progress), draft-ietf-mobileip-paging-hawaii-00.txt, June 1999.

[54] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, and L. Salgarelli, "IP micro-mobility support using HAWAII," Internet Draft (Work in Progress), draft-ietf-mobileip-hawaii-00.txt, June 1999.

[55] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, and S. Wang, "HAWAII: A Domain-based Approach for Supporting Mobility in Wide-area Wireless Networks," in *Proc. of the 7th IEEE International Conference on Network Protocols (ICNP'99)*, Toronto, Canada, 1999.

[56] P. Reinbold and O. Bonaventure, "A Comparison of IP Mobility Protocols," in *Proc. of the 8th IEEE Symposium on Communications and Vehicular Technology (SCVT'2001)*, Delft, Netherlands, Oct. 2001.

[57] _____, "A Survey of IP Micro-Mobility Protocols," Technical Report Infonet-2002-06, Infonet Group, University of Namur, Belgium, Mar. 2002. [Online]. Available http://www.infonet.fundp.ac.be/doc/reports/Infonet-TR-2002-06.pdf.

[58] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," Request For Comments (Proposed Standard) 3261, Internet Engineering Task Force, June 2002.

[59] A. C. Snoeren and H. Balakrishnan, "An End-to-End Approach to Host Mobility," in *Proc. of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'2000)*, Boston, MA, USA, Aug. 2000, pp. 155-166.

[60] W. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Reading, MA: Addison-Wesley, 1994.

[61] C. Tan, S. Pink, and K. Lye, "A Fast Handoff Scheme for Wireless Networks," in *Proc. of the 2nd ACM International Workshop on Wireless Multimedia (WoWMoM)*, Seattle, WA USA, Aug. 1999, pp. 83-90.

[62] A. Tanenbaum, *Computer Networks,* 3rd Edition, Upper Saddle River, NJ: Prentice Hall, 1996.

[63] D. Thaler, M. Handley, and D. Estrin, "The Internet Multicast Address Allocation Architecture," Request For Comments (Informational) 2908, Internet Engineering Task Force, Sept. 2000.

[64] A. Valko, "Cellular IP: A New Approach to Internet Host Mobility," *ACM Computer Communication Review*, vol. 29, no. 1, pp. 50-65, Jan. 1999.

[65] E. Wedlund and H. Schulzrinne, "Mobility Support using SIP," in *Proc. of the 2nd ACM International Workshop on Wireless Multimedia (WoWMoM)*, Seattle, WA, USA, Aug. 1999, pp. 76-82.

[66] D. Wetherall, OTcl Tutorial, Version 0.96, Sept. 1995. [Online]. Available ftp://ftp.tns.lcs.mit.edu/pub/otcl/doc/tutorial.html.

[67] J. Widmer, "Network Simulations for a Mobile Network Architecture for Vehicles," Technical Report TR-00-009, International Computer Science Institute, Berkeley, CA, May 2000.

[68] C. Williams (Editor), "Localized mobility Management Requirements, " Internet Draft (Work in Progress), draft-ietf-mobileip-lmm-requirements-02.txt, June 2002.

[69] K.D. Wong, H.-Y. Wei, A. Dutta, and K. Young, "Performance of IP Micro-Mobility Management Schemes using Host Based Routing," in *Proc. of the 4th International Symposium on Wireless Personal Multimedia Communications (WPMC'2001)*, Aalborg, Denmark, Sept. 2001.

APPENDIX A

NETWORK SIMULATION FRAMEWORK:

FURTHER DETAILS AND API

In this appendix, we present some additional details for the network simulation framework for local-area mobility support, including the format of introduced trace files resulting from the simulation, and the internal OTcl API available to the C++ implementation for various simulation objects[28]. In section V, we introduced the OTcl API usable from within network simulation scripts to configure various simulation aspects.

Source Code Compilation

The C++ source code compiles successfully on Solaris™ 2.6, with *gcc* version 2.8.1. We were not able to successfully compile the CIMS package on Solaris™ 8. In the future, we plan to port the source code to Solaris™ 8 with a recent *gcc* version such as 2.96 20000306 (experimental).

Output Trace Files

ns-2 has built-in capabilities to output a configurable simulation trace file, and wireless trace file for wireless traffic [26].We create two MIP registration trace files to track the MH's registrations. The first trace file, named "basicRegLog," tracks the registrations for a non-hierarchical setting (for base MIP). A line in the file constitutes a single registration record and is comprised of the following fields: <registration request send time, registration reply receive time, registration latency> where registration latency is computed as the difference between the registration reply receive time and the registration request send time. The second trace file, named "regionalRegLog," tracks the regional registrations within a FA hierarchy setting. A line in the file constitutes a single

[28] The API listing in this appendix is not an exhaustive listing.

registration record and is comprised of the following fields <MH care-of address, registration request send time, MH HA, registration reply receive time, regional registration latency>. We do not currently track home and home-regional registrations in a FA hierarchy setting.

The UDP's traffic monitoring agent generated a trace file, named "cbrseqfile," which signaled packet loss, duplication, or out-of order. We extended the format of the trace file as follows. For each UDP packet received, a line is written to the file, comprised of the following fields: <Packet Sequential number, Packet Unique ID, Packet Send time, Packet playout time, Packet Receive time> where the packet playout time is computed as the packet send time + a desired playout delay to be configured by the simulation script. If a packet is received after its packet playout time, it is dropped. Such event is signaled in the trace file by printing a line "PLAYOUT DROP ***" after the packet trace line.

The simulation scripts can be configured to generate an instantaneous TCP throughput file. Each line in the file constitutes the TCP throughput as observed in the previous measuring interval. Each output line is comprised of the following fields: <Current time, throughput during measuring interval in bytes, throughput as a Mbps measure>.

BS/RFA/GFA Registration Agent OTcl API

The following methods permit management of maintained tunnel entries. A tunnel entry can be a visitor entry or a binding cache entry. Upon receiving a BU message, the visitor entry is converted to a binding cache entry.

encap-route <mhaddr> <coa> < lifetime> <seqno>

Set up a tunnel entry for <mhaddr> pointing to <coa> with the specified <lifetime> and marking the sequential number for the corresponding registration request as <seqno>. If this node is an RFA/GFA the tunnel entry is set to be a visitor tunnel entry. A clear-reg method is scheduled such that it executes after the lifetime expires, in order to clear the entries for this MH. The encap-route method is used to set up tunneling entries by a HA for the current MH's care-of address, or by a RFA/GFA to one of the children FAs.

***clear-reg* <mhaddr>**

Clear any registration information for this MH.

***tunnel-exit* <mhaddr>**

Returns the current tunnel exit for this MH, or –1 if no data could be found for this MH.

***tunnel-entry-type* <mhaddr>**

Returns the tunnel entry type for <mhaddr>. The possible return values are: -1 for an entry not found, 1 for a visitor entry, and 2 for a binding cache entry. This method is used to decide whether or not a crossover FA should issue a registration reply back to the MH based on the current tunnel entry type

***to-binding-cache* <mhaddr> <lifetime>**

Converts the visitor entry for <mhaddr> into a binding cache entry, and sets the entry lifetime to <lifetime>. The care-of address for the binding cache entry is kept the same as in the visitor entry (to the same child FA). This method is used in response to receiving a BU message as part of a tunneling consistency mechanism.

The following methods allow the management of MH's registration sequential numbers for purposes of replay protection, and message freshness.

***set-last-seqno* <mhaddr> <seqno>**

Set the current sequential number for a MH to <seqno>. The sequential numbers are stored as a Tcl array indexed by the MH node address.

***last-seqno* <mhaddr>**

Get the current sequential number stored for a MH, or –1 if no current number is stored for this MH

The following method allows for querying the remaining registration lifetime for a specific MH. A crossover FA uses such method in order to identify the remaining lifetime used in forming a regional registration reply for this MH.

***remaining-reg-time* <mhaddr>**

Returns the remaining registration lifetime for a MH, or –1 if the registration lifetime has expired, or no data could be found for this MH.

The following methods allow for setting and querying the home registration latency as part of the KOPA approach. The measured home registration latency is used by a crossover FA to calculate the binding update lifetime as part of tunneling consistency mechanism (see section 3.4.1).

setHomeRegLatency **<mhaddr> <latency>**

Set the home registration latency for a MH to <latency>. Upon receiving a home registration reply, each RFA on the hierarchy path towards the MH, stores such measure. The latency is computed by subtracting the registration request send time, stored in a pending registration request, from the registration reply receive time.

getHomeRegLatency **<mhaddr>**

Get the home registration latency stored for this MH, or –1 if no information is stored.

An old FA (BS node) uses the following methods in order to implement the smooth handoff mechanism. These registration agent's methods invoke the corresponding methods in the decapsulator object of the node (The corresponding methods have the same method names and parameters). The decapsulator object stores a new FA data structure to implement the smooth handoff mechanism. In addition, such methods allow for the tunneling consistency mechanisms to coexist with the smooth handoff mechanism, by inspecting and updating the remaining binding update time (see sections 3.3.3 and 3.4.1.4).

setNewFA **<mhaddr> <new FA> <lifetime>**

Install a <new FA> entry for the <mhaddr> with the specified <lifetime>. The new FA data structure is maintained within the Decapsulator object of the node. During <lifetime>, any data packets arriving at this old FA are tunneled to <new FA>. A clear-newFA method is scheduled such that it executes after the lifetime expires, in order to clear stored data for the MH.

clear-newFA **<mhaddr>**

Clear the new FA entry for <mhaddr>. As a result, any data packets arriving at this old FA after a call to this method are dropped.

getNewFA **<mhaddr>**

Get the <new FA> entry for this MH, or –1 if no data could be found.

remaining-cache time **<mhaddr>**

Returns the remaining lifetime of the binding cache entry found within this old FA for the <mhaddr>, or –1 if entry expired or not found.

update-binding-time **<mhaddr> <lifetime>**

Update the lifetime of an existing binding cache entry for <mhaddr>. The care-of address in the binding cache entry is maintained the same.

APPENDIX B

ACRONYMS

BS	Base Station
BU	Binding Update
C-GFAs	Cooperating Gateway Foreign Agents
CH	Correspondent Host
CIMS	Columbia IP Micro-Mobility Software
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DOP	Delete Old Path
FA	Foreign Agent
GFA	Gateway Foreign Agent
GRE	Generic Routing Encapsulation
HA	Home Agent
HRGFA	Home Registered Gateway Foreign Agent
HR-LH	"Home Registration"-"Local Handoff"
ICMP	Internet Control Management Protocol
KOPA	Keep Old Path Alive
LMM	Localized Mobility Management
MA	Mobility Agent
Mbps	Mega bit per second
MH	Mobile Host
MIP	Mobile IP
MIP_RR	Mobile IPv4 Regional Registration Framework [30]
MSA	Mobility Security Association
NAI	Network Access Identifier
NC-GFAs	Non-Cooperating Gateway Foreign Agents
NS, ns	Network Simulator
PFANE	Previous Foreign Agent Notification Extension
RFA	Regional Foreign Agent
SH	Smooth Handoff
SINP	Switch Immediately to New Path

VITA

Ayman A. Abdel Hamid was born in Alexandria, Egypt, on October 21, 1970. He received his Bachelor of Science in Computer Science and Automatic Control from the Faculty of Engineering, Alexandria University, Egypt, in June 1993. He worked as a Teaching Assistant for the Department of Computer Engineering at the Arab Academy for Science and Technology and Maritime Transport from January 1995 to July 1997. In August 1998, he received his Master of Science from the Department of Computer Science and Automatic Control, Faculty of Engineering, Alexandria University, Egypt. He started working on his Ph.D. degree in Computer Science at Old Dominion University, in August 1997. During the course of his Ph.D. study, he co-authored ten scientific papers and technical reports. He is a member of IEEE, IEEE Computer Society, and ACM.

Permanent address: Department of Computer Science
Old Dominion University
Norfolk, VA 23529-0162
USA