12-1-1998

Mobile IP and Route Optimization: A Simulation Study.

Preetha P. Kannadath

Mobile IP and Route Optimization:
A Simulation Study

A Thesis

Presented to the

Department of Computer Science

and the

Faculty of the Graduate College

University of Nebraska

In Partial Fulfillment

of the Requirements for the Degree of

Master of Science

University of Nebraska at Omaha

by
Preetha P. Kannadath
December, 1998

Thesis Advisor: Dr. Hesham El-Rewini

THESIS ACCEPTANCE

Acceptance for the faculty of the Graduate College, University of Nebraska, in partial fulfillment of the requirements for the degree Master of Science, University of Nebraska at Omaha.

Committee

Name	Department/School
	ISQA
	Computer Science
	Computer Science

Chairperson H El Rewini

Date 12/1/98

ABSTRACT

Powerful light-weight portable computers, the availability of wireless networks, and the popularity of the Internet are driving the need for better networking support for mobile hosts. Current versions of the Internet Protocol (IP), make an implicit assumption that the point at which a computer attaches to a network is fixed and its IP address identifies the network to which it is attached. Packets are sent to a computer based on the location information contained in its IP address. Therefore, transparent host mobility is not supported by IP. But there is a growing need for users to be able to connect their portable computers to the Internet at any time, and stay connected even when they are on the move.

Amongst various options available to implement host mobility, Mobile IP, which is an IETF (Internet Engineering Task Force) Draft Standard, is the most feasible one. The Mobile IP protocol, that is compatible with the TCP/IP protocol suite, allows a mobile host to move around the Internet without changing its identity. It is an internet (IP) layer solution to host mobility. Route Optimization, which is an extension to Mobile IP, allows a node to cache the location of a mobile host and to send packets directly to that mobile host.

This thesis describes the development of a model to simulate Mobile IP with Route Optimization. An event-driven simulator was developed to study this protocol. Using this simulator, experiments were conducted to study the performance of the

protocol under various changing network parameters. These experiments also establish the merits of Route Optimization over base Mobile IP.

Acknowledgement

I would like to express my genuine gratitude to Dr. Hesham El-Rewini, who suggested this project and helped me from the start to the finish. Despite his busy schedule, he was always there to help and encourage me. I am also thankful to Dr. Hesham Ali, Dr. Yong Shi and Dr. Hassan Farhat, who were members of my committee, for the many suggestions that they made on the content and presentation of this thesis.

List of Diagrams

List of Graphs

CHAPTER 1 : INTRODUCTION

The Personal computer has grown to become a truly transportable device with dozens of megabytes of main memory, gigabytes of disk storage, and very high processing power. Laptop computers and palmtop computers have become another choice in a wide spectrum of available computer resources. As people move from place to place with their portable computers, keeping connected to the network can become a very challenging and expensive proposition. This leads to mobile networking technologies, the goal of which is to keep automatic communications with globally interconnected computing resources. This communication should be as natural for people on the move as it is for people sitting at high-performance workstations in their offices.

Recent advances in mobile computer hardware and wireless communication technologies have introduced the era of mobile computing. The most important mobile computer system of interest is undoubtedly the laptop computer, which is rapidly becoming indispensable for the business traveler.) There are many kinds of communications adapters that allow convenient access to modern computer networks, and laptop computers typically come equipped with networking software to transmit and receive data over those networks. The proliferation of powerful laptop computers, and other inexpensive portable systems, comparable in power with high end desktops, and the rapid development of wireless communication technology, promises to provide users with network access at any time and in any location. The primary goal of mobile computing research is to provide protocols that would allow a mobile user to transparently access services and resources on the fixed network.

With wireless communication and battery-powered operation, mobile computers like laptops and palmtops can be completely unattached and still have full connectivity to a network. Devices using infrared light or radio frequency signals, that enable a mobile computer to maintain wireless connection to a LAN, have become available. The Cellular Digital Packet Data (CDPD) standard over the cellular telephone technology also enables a mobile computer to stay connected to a network. A mobile computer may use infrared or even Ethernet links while inside a building, may use radio LAN connections when leaving the building, and may use CDPD to remain connected while away from all the enterprise base stations. Thus, using wireless technology, a kind of multi-modal operation is possible, whereby software on a mobile computer can function uninterrupted even when the medium of connection to the network is changed.

The emergence of World Wide Web and the easy availability of Web Browsers like Netscape and Internet-Explorer have made the internet accessible to even non technical people. At the rate at which usage of the internet is growing, it would soon become the most important medium for all kinds of communication.

1.1 Mobility and Its Significance

A variety of applications can make use of wireless uninterrupted connection to information sources. Some key applications are those that provide critical information to active individuals on the move, like stock quotations to a broker or inventory information to a traveling salesman. Multimedia applications, especially those that have continuous time-sensitive data like video, can also benefit from the ability to be mobile. Law enforcement, public safety like fire-fighting, or remote medical diagnosis, are some possible application areas that would require real-time images to be transmitted over wireless links. It would not be far-fetched to think that at some point in the future, computers would not need any wired connection to a network, because they would always be wireless connected.

1.3 Problems with Mobility

Current versions of the Internet Protocol, or IP, make an implicit assumption that the point at which a computer attaches to a network is fixed and its IP address identifies the network to which it is attached. Packets are sent to a computer based on the location information contained in its IP address.

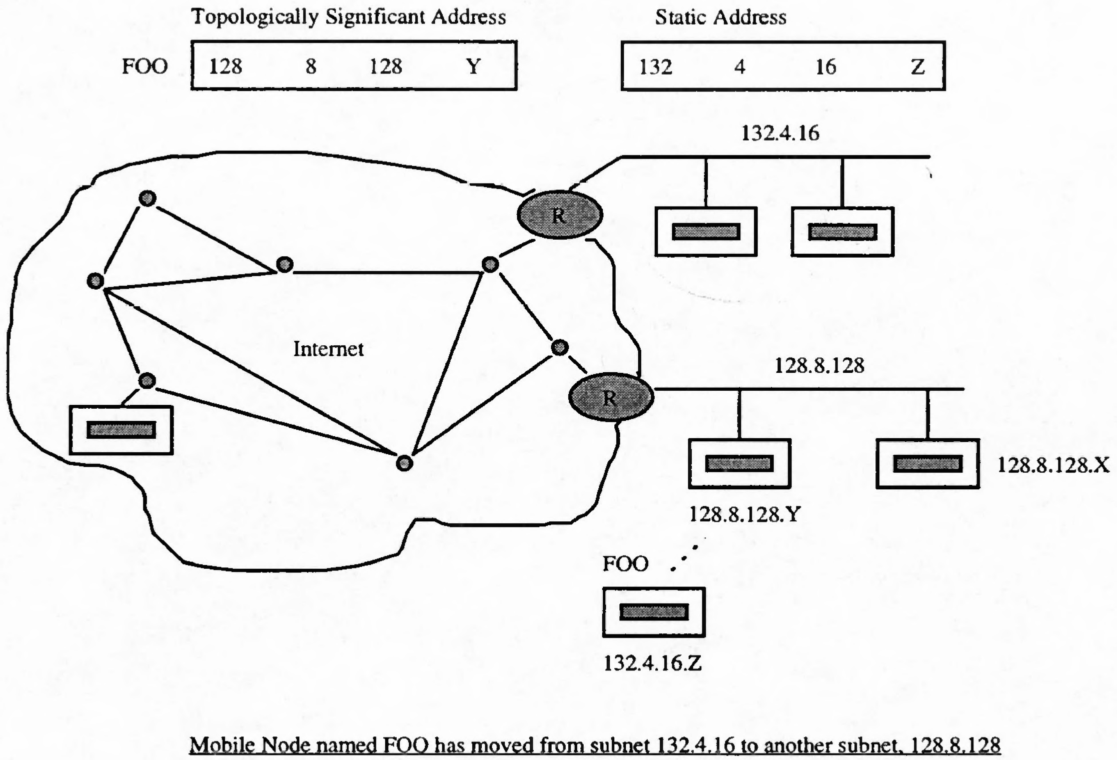

Topologically Significant Address
Static Address
FOO
128
8
128
Y
132
4
16
Z
132.4.16
Internet
R
R
128.8.128
128.8.128.X
128.8.128.Y
FOO
132.4.16.Z
Mobile Node named FOO has moved from subnet 132.4.16 to another subnet, 128.8.128

Figure 1 : Two Tier IP Addressing

If a mobile computer or mobile host, moves to a link with a different network prefix, while keeping its IP address unchanged, its address will not reflect the new point of attachment.

As is shown in Figure 1, the mobile node named FOO has moved from the subnet 132.4.16 to another subnet 128.8.128, its address (132.4.16.Z) no longer reflects its point of attachment. Consequently, existing routing protocols will not be able to find the address, and thus messages sent to this node would be lost. Mobility is the ability of a node to change its point of attachment from one link to another while maintaining all existing communications and using the same IP address at its new location.

Unfortunately, present day internetworking protocols such as TCP/IP, IPX and AppleTalk do not support host migration between networks.

Mobility is not required if all communications are initiated by the user of a mobile node, and the user does not mind shutting down and restarting applications at a new location. Such nomadic behavior, however is not sufficient, if communication with the mobile node is initiated by other nodes, or if the mobile node is running applications that can not be shut down. Remote printing, remote login and file transfer are some examples of applications whose communication can not be interrupted when a mobile node moves from one link to another.

1.4 Solutions to the Mobile Computing Problem

Host Mobility can be supported in three different ways, namely Host Specific Routing, Change of IP Address, and the Mobile IP protocol. These three methods and their pros and cons are discussed in the following sections.

1.4.1 Host Specific Routing

Routing of packets on the internet is done using the prefix portion of the destination address. Computers that belong to the same subnet have the same prefix portion for their address. Routers, therefore, only need to have the prefix portion of an address in the routing table. Host Specific Routing refers to the technique where the entire address, not just the prefix portion of it, is used by the routers to make routing decisions. This, however, leads to having very large routing tables, because every host specific address would need to have an entry in the routing table. To support mobility, one of the solutions is to have Host Specific Routing for all the mobile hosts. There are some issues with this solution that make it impractical to implement. These issues include the following,

1. Minimally, host-specific routes must be propagated to all the nodes along the path between a mobile node's old location and its new location. Therefore, large number of routes must be updated every time a node moves.
1. In the next few years, a very large number of mobile nodes are expected to operate on the internet, which would make updates to routes a very costly and time consuming operation.

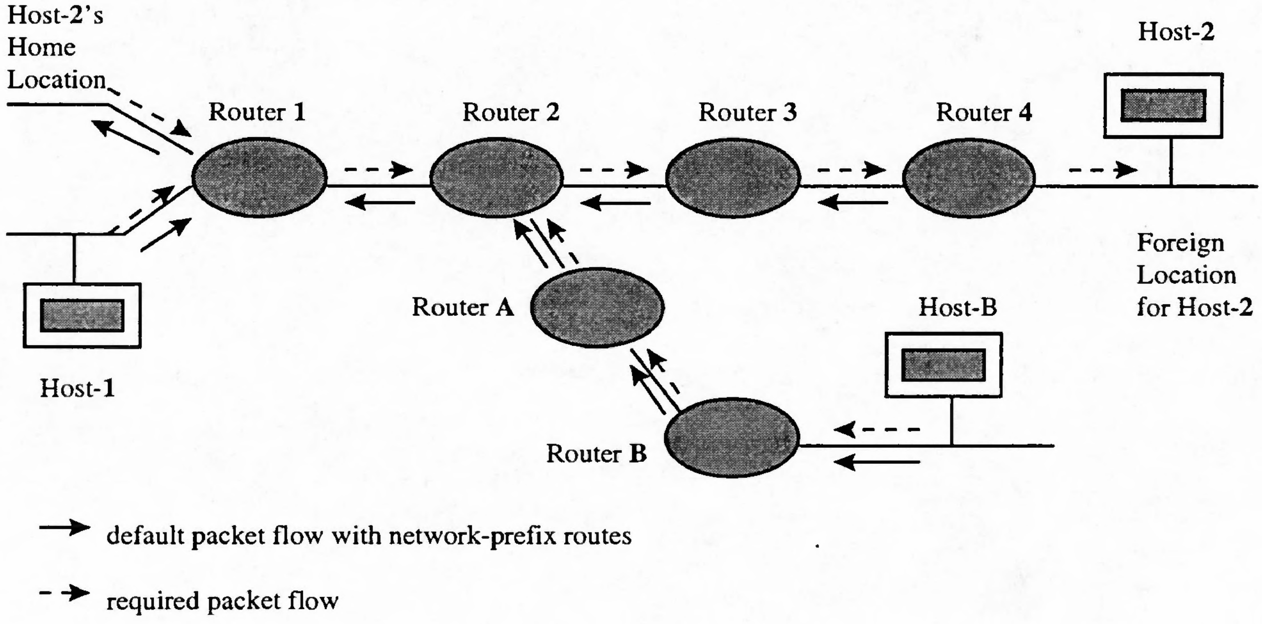


Figure 2 : Routing Packets to a Mobile Node on a Foreign Link

In Figure 2, solid arrows show the direction in which a node would forward packets if that node contained only the route to Host-2's home location, and dashed arrows show the direction in which a node should forward packets in order to reach Host-2 at its new location. If Host-2 changes its location again, so that it is connected to Router B, the following updates should be made to host specific routes,

1. Router 3 and 4's host specific route must be deleted.
2. Router 2's existing host specific route must be modified to have a new Next Hop of Router A.
3. Router A and B must be given a new host-specific route to point to the new location.

If mobile nodes change their points of attachment rapidly, there would be a substantial number of routing updates, which is not desirable. Also there are serious security implications to using host-specific routes to accomplish node mobility in the Internet, which would require authentication and complicated key management protocols to address [4].

1.4.2 Change the IP Address

Both TCP and UDP use the concept of ports. A port is a 16 bit integer which allows the receiving TCP and UDP protocol entities to determine which of the many possible higher layer applications is supposed to receive the data portion of any segment that arrives over the network. This allows many applications to be open simultaneously on a node and have TCP/UDP route the application traffic to the proper destination.

There is an enormous installed base of IP nodes, all of which assume that the source address, destination address, and port will remain constant over the duration of a connection. When a mobile node moves, the ongoing communication between the mobile node and any other node would have to be terminated, with new connections being initiated by the mobile node at its new address. Thus by definition, changing a mobile node's address as it moves does not solve the problem of node mobility.

There are two problems associated with changing a node's IP address. To find a nomadic node (whose address keeps changing), the nomadic node's IP address entry in the Domain Name System (DNS) must be updated every time the node changes link, i.e. every time the node changes its IP address. In addition, a node which looks up a nomadic node's IP address should know that the address returned from a name server is subject to change at any moment. The net result is a large increase in queries and dynamic updates to DNS name server causing network congestion.

1.4.3 Mobile IP

Mobile IP was approved by the Internet Engineering Steering Group in June 1996 and published as a Proposed Standard in November 1996. A Proposed Standard is the first significant step in the evolution of a protocol from an Internet Draft into a full Internet Standard. The Mobile IP standard documents include the following Request For Comments(RFCs).

- RFC 2002, which defines the Mobile IP protocol itself.

- RFC 2003,2004, and 1701 which define three types of tunneling used in Mobile IP.
- RFC 2005, which describes the applicability of Mobile IP.
- RFC 2006, which defines the Mobile IP Management Information Base.

Mobile IP is a network-layer solution to node mobility in the Internet. This means that Mobile IP accomplishes its task by setting up the routing tables in appropriate nodes, such that IP packets can be sent to mobile nodes not connected to their home link. Mobile IP assumes that packets are routed based only upon the IP Destination Address, and typically only the network prefix portion of that address.

1.5 Thesis Objective

In this thesis, we build a simulator to study the Mobile IP environment. We also study the effect of applying Route Optimization techniques to the performance of Mobile IP. We simulate Mobile IP in a large network to investigate performance measures like the percentage of packets delivered and the hop count per packet. It is our goal to find out how the protocol performs under a variety of changing network parameters. By doing so, we demonstrate the feasibility of Mobile IP and establish the relative merits of Route Optimization.

The first phase in this thesis consists of understanding the Mobile IP protocol, so that we can build a model that truly reflects the essential details of the protocol. Next we design and develop the computer programs that simulate the model. Finally, we determine the network parameters that could influence the performance, conduct a series of experiments by varying these parameters, and gather results.

1.6 Thesis Organization

This chapter introduced the need for mobility and the options available to implement it. In Chapter 2, we discuss the background and details of Mobile IP. Chapter 3 presents some of the ongoing research work related to Mobile IP. The architecture of our model for simulating Mobile IP is presented in Chapter 4. Experiments conducted using our model and the results are discussed in Chapter 5. Finally, in Chapter 6, we present our conclusions and introduce possible enhancements to our model.

CHAPTER 2 : BACKGROUND AND PROBLEM SPECIFICATION

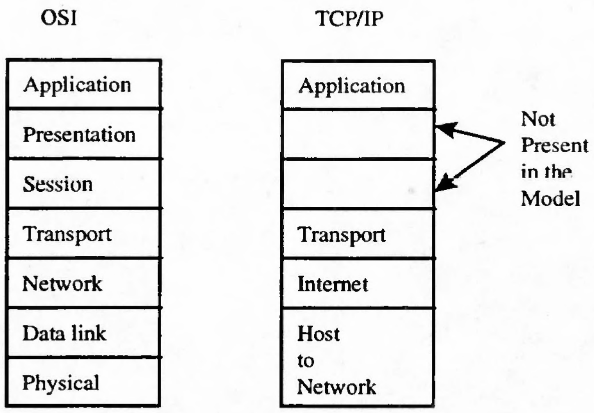
In this chapter we build the background to Mobile IP and then present various components of the protocol. This includes a brief discussion of network communication basics, Internet Protocol (IP), and basic Mobile IP, followed by Route Optimization, which is a proposed extension to Mobile IP.

2.1 Network Communication Basics

To deal with network design complexity, the network communication task is broken into smaller, more manageable subtasks. This leads to a network reference model. The OSI Model and the TCP/IP Model are the well known reference models.

2.1.1 The OSI Model

The Open Systems Interconnection (OSI) reference model for computer networking is defined by the International Organization for Standardization (ISO). The OSI Model, in a structured way, breaks down the design problem into a series of subtasks. Each subtask corresponds to a layer in the network structure. There are seven layers, each one of these layers performs a specific set of functions and in turn provides a distinct set of services to the layer above it. Figure 3 shows the layers of the OSI model, and reference [3] provides a detailed explanation of each layer.



Network Reference Models

Figure 3 : Network Reference Models

2.1.2 The TCP/IP Model

Under the TCP/IP Model, unlike the OSI Model, as shown in Figure 3, the Host-to-Network layer is a void. The model does not define any protocol here, except to point out that the host has to connect to the network using some protocol, so that it can send IP packets over the connection [3]. The primary protocol layers under this model are the following,

1. The Internet Layer: Under TCP/IP, this layer is responsible for sending packets of data from the source to the destination. Packets may arrive in a different order than the order they were sent in. The internet layer defines an official packet format and protocol called IP (Internet Protocol).
2. The Transport Layer: This layer is designed to allow peer entities on the source and destination hosts to carry on a conversation. Two end to end protocols have been

defined here. The first one, TCP (Transmission Control Protocol) is a reliable connection-oriented protocol that allows a byte stream originating on one machine to be delivered, without error, on any other machine in the internet. It fragments the incoming byte stream into discrete messages and passes each one onto the internet layer. At the destination, the receiving TCP process reassembles the received messages into the output stream. TCP also handles "flow control" to make sure that a fast sender cannot swamp a slow receiver with more messages than it can handle. The second protocol in this layer is UDP. UDP is an unreliable connectionless protocol for applications that do not need TCP's sequencing or flow control.

2.2 Internet Protocol

Internet Protocol, or IP, forms the basis of Mobile IP. This section presents an introduction to the format of an IP packet and discusses Routing under IP. Reference [3] provides a detailed discussion on these subjects.

2.2.1 IP Packet

An IP packet consists of a segment of data passed down from the transport or higher layers plus a small IP header prepended to the data. The fields in the IP header are shown in Figure 4. IP addresses are numbers assigned to each network interface of a node. Nodes with multiple network interfaces, such as routers, have multiple addresses, one per interface. IP Addresses have two components, the network prefix portion and the host portion. The network prefix is a sequence of bits, identical for all nodes attached to the same network, which requires the host portion to be unique for each node on the same network. The network prefix identifies a network and the host portion identifies a specific host or router connected to that network.

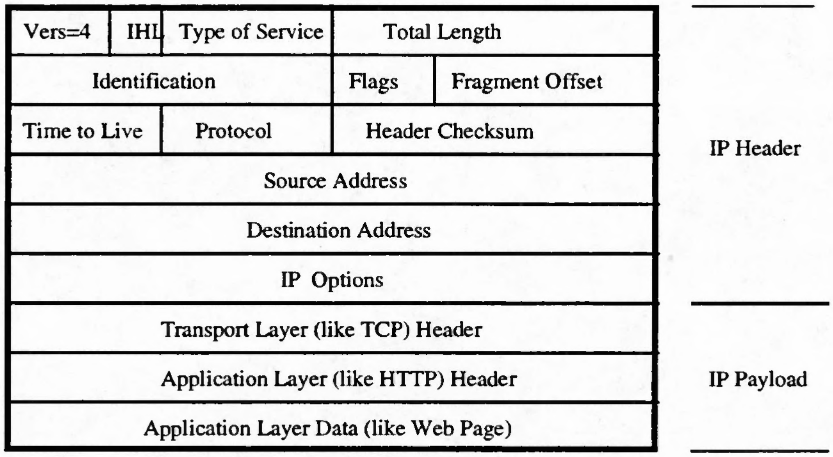


Figure 4 : IP Packet, Header and Payload

2.2.2 Routing of IP packets

Routing decisions in a TCP/IP network are typically based upon the network-prefix portion of the IP Destination Address, rather than the entire address. Using special procedures called routing protocols, routers exchange information among themselves about the networks and hosts to which they are connected. This allows them to build tables, called routing tables, which are used to select a path for any given packet from the source to the destination. Mobile IP deals with protocols that allow routing of IP packets to the current location of a mobile node.

2.3 Mobile IP Terminology

The following terms define entities that are used under the Mobile IP protocol. We will refer to these terms in the subsequent discussion on Mobile IP protocol.

1. Mobile Node: This is a node (a laptop for instance) which can change its point of attachment to the Internet from one link to another while maintaining any ongoing communication.
2. Home Agent: This is a router with an interface on the mobile node's home network which:
   - the mobile node keeps informed of its current location, as represented by its care-of address, as the mobile node moves from network to network
   - intercepts packets destined to the mobile node's home address and tunnels them to the mobile node's current location, i.e. to the care of address.
3. Foreign Agent: This is a router on a mobile node's foreign network which:
   - assists the mobile nodes in informing its home agent of its current care of address.
   - provides a care of address and de-tunnels packets for the mobile node that have been tunneled by its home agent.
   - serves as a default router for packets generated by the mobile node while connected to this foreign network.

4. Tunneling : A tunnel is the path followed by a packet while it is encapsulated within the payload portion of another packet. Tunneling is the procedure used by the home agent to forward packets to a mobile node's current location.
5. Home Address : A mobile node's home address is an IP address assigned to the mobile node permanently in its home network. The home address does not change as a mobile node moves from network to network. A mobile node communicates with all other nodes using only its home address. A mobile node's home address is the IP source address of all packets sent by the mobile node and IP Destination Address of all packets sent to the mobile node.
6. Care Of Address : This is a temporary IP address obtained by the mobile node when it is away from home. The mobile node obtains this address from the foreign agent, in which case the foreign agent decapsulates tunneled packets meant for the mobile node, before forwarding them to the mobile node. Optionally, in the absence of a foreign agent, the mobile node can in some cases, acquire a care-of address externally.

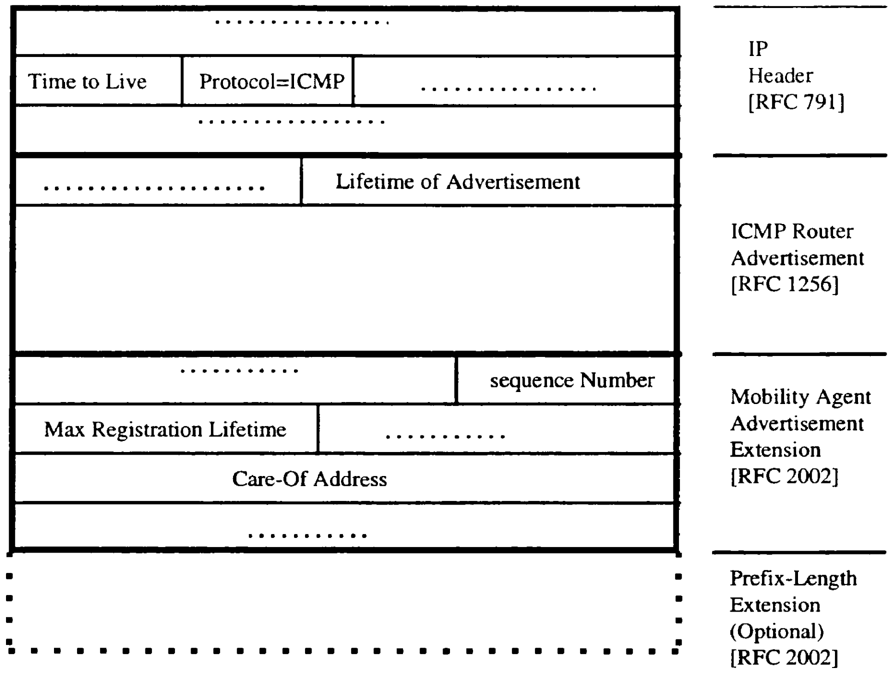
2.4 Mobile IP Protocol

The base Mobile IP protocol consists of three major functions, namely Agent Discovery, Registration and Packet Tunneling. These three functions are briefly discussed in the following sections. References [1], [4] and [5] contain a detailed explanation of these functions.

2.4.1 Agent Discovery

Mobile Agents (Home Agents and Foreign Agents) advertise their presence via agent advertisement messages. The source address in the advertisement message is used by mobile nodes to determine if they are still connected to the home network. If the network-prefix of the source address in the IP header of the advertisement message matches the network-prefix of the mobile node's home address, then the mobile node concludes that it is still connected to its home agent. Otherwise the mobile node assumes that it is not connected to its home agent, and proceeds to get a care of address from the foreign agent at the new location.

As shown in Figure 5, Agent advertisement messages are modified ICMP router advertisements. References [1] and [4] contain a detailed description of the various messages used to implement Mobile IP.



<ins>**Agent Advertisement Message**</ins>

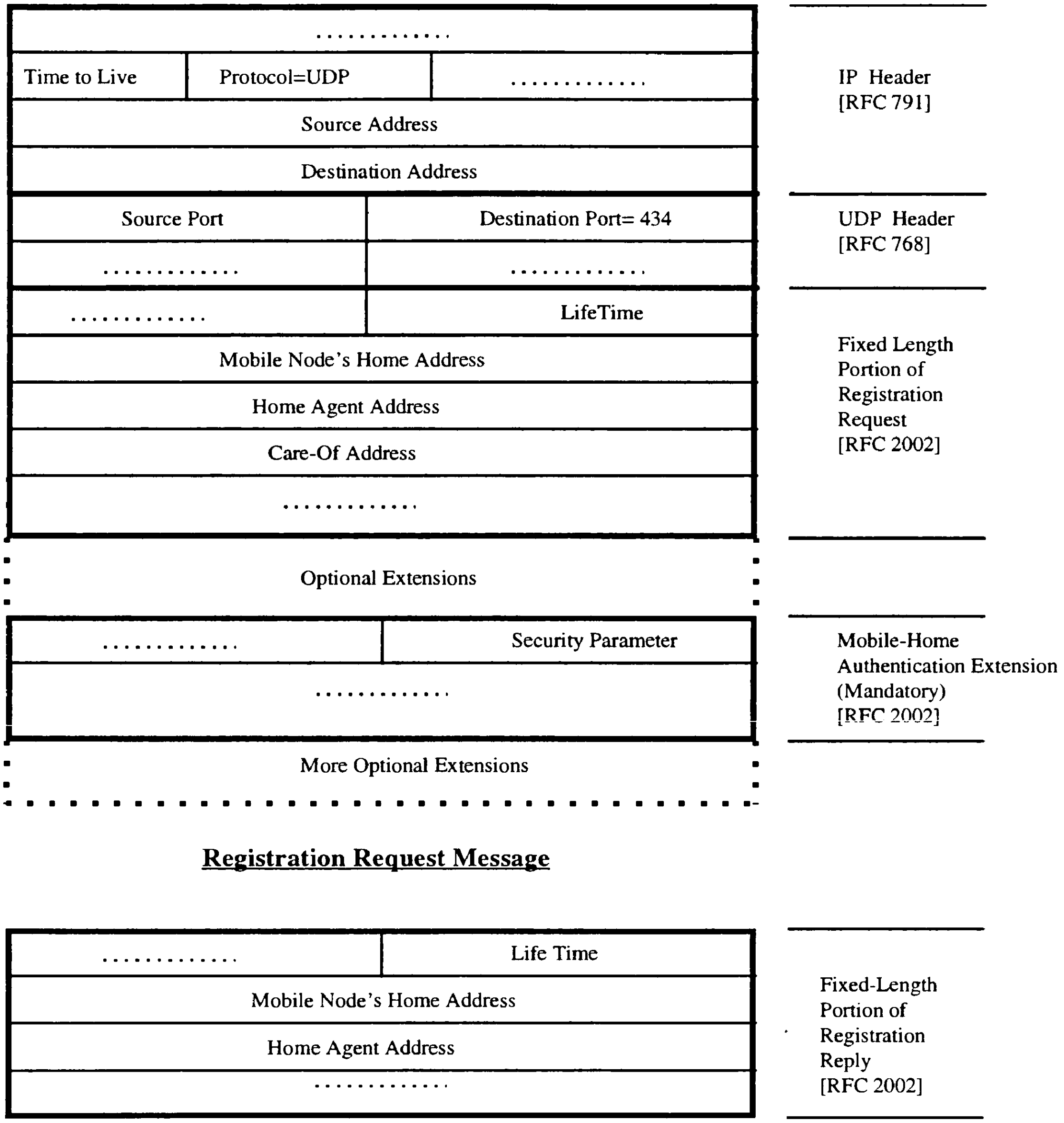
Figure 5 : Agent Advertisement Message

2.4.2 Registration

When a mobile node is away from home, it registers its care of address with its home agent. Mobile IP Registration is the process by which a mobile node,

- requests routing services from a foreign agent on a foreign network.
- informs its home agent of its current care-of address.
- renews a registration which is due to expire.
- de-registers with the home agent when it returns to its home network.

Registration consists of an exchange of a Registration Request message and a Registration Reply message between a mobile node and its home agent, possibly by involving a foreign agent.



Registration Request Message

Registration Reply (fixed portion only)

Figure 6 : Registration Messages

The mobile node sets the Lifetime field in a Registration Request to the number of seconds it would like its registration to last before it expires. The Lifetime field in the corresponding Registration Reply tells the mobile node how long the registration actually lasts before it expires, that is, the maximum length of time that was permissible to the home agent or the foreign agent.

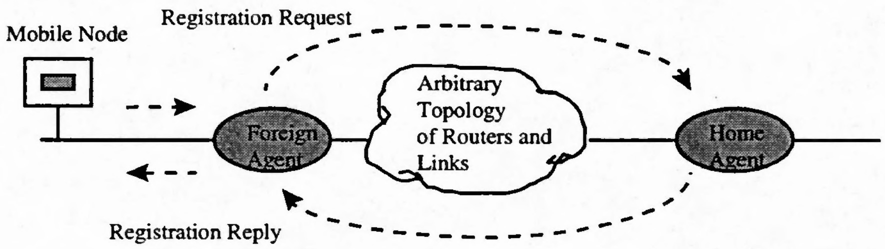


Figure 7 : Registration Protocol

As shown in Figure 7, the home agent receives the Registration Request and sends back to the mobile node a Registration Reply, which tells the mobile node whether the registration was successful. The Code field in the Registration reply would indicate the reason for the failure. If a mobile node does not receive a Registration reply within a reasonable period of time, then it retransmits the Registration Request a number of times until it does receive a Reply. If the reply indicates rejection, the mobile node can attempt to repair the error that caused the rejection and attempt a new registration.

2.4.3 Tunneling

Tunneling is the mechanism by which the home agent forwards packets to mobile nodes. As shown in Figure 8, IP packets are placed within the payload portion of new IP packets, and the destination address of the encapsulating (outer) IP header is set to the mobile node's care of address. When the Foreign Agent receives this IP packet, it decapsulates it by removing the outer IP packet, and sends the original packet to the Mobile Node.

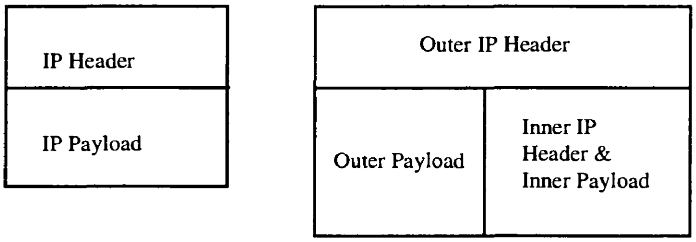
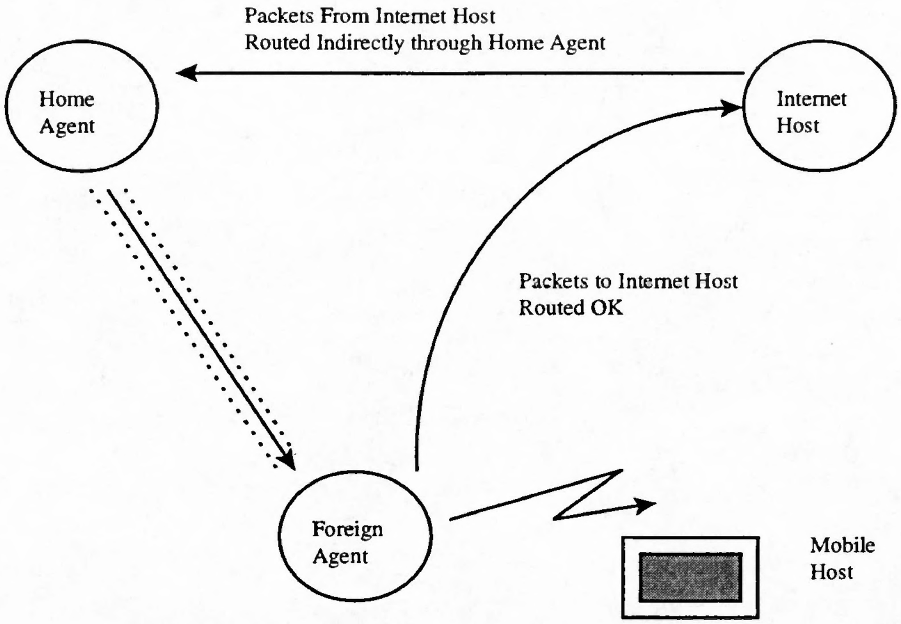


Figure 8 : Tunneling

2.5 Route Optimization

The base Mobile IP protocol allows any mobile node to move about changing its point of attachment to the Internet, while continuing to be identified by its home IP address. Correspondent nodes, sending IP packets to a mobile node, send them to the mobile node's home address in the same way as with any other destination [3]. This scheme allows transparent inter-operation between mobile nodes and their correspondent nodes, but forces all packets for a mobile node to be routed through its home agent. Thus packets to the mobile node are often routed along paths that are significantly longer than optimal, which is called Triangle Routing, as shown in Figure 9.



Triangle Routing

Figure 9 : Triangle Routing

For example, if a mobile node is visiting some network, even packets from a correspondent node on the same network must be routed through the Internet to the mobile node's home agent (on its home network) only then to be tunneled back to the original network for final delivery. This indirect routing can significantly delay the delivery of the packets to mobile nodes and it places an unnecessary burden on the networks and routers along its path through the internet.

Route Optimization defines extensions to the base Mobile IP protocol, to allow for better routing, so that packets can be routed from a correspondent node to a mobile node without going through the home agent.

Route Optimization extensions provide a means for nodes that implement them to cache the current care-of address of a mobile node, and to tunnel packets directly to that care-of address, bypassing the possibly lengthy route to and from that mobile node's home agent. These cache tables, that are maintained in the intermediate routers, are called Binding Cache.

2.5.1 Binding Cache

In the absence of any binding cache entry, packets destined for a mobile node will be routed to the mobile node's home link in the same way as any other IP packet and then tunneled to the mobile node's current care-of address by the mobile node's home agent. If the sending node had a binding cache entry for the mobile node, it would be able to send packets directly to the mobile node without the services of the home agent. Any node may maintain a binding cache to optimize its communication with mobile nodes. A node may create or update a binding cache entry for a mobile node only when it has received and authenticated the mobile node's current location. Each binding in the cache also has an associated lifetime, after the expiration of this time period, the binding is to be deleted from the cache. A node can use any reasonable strategy for managing the space within the binding cache. When a new entry is to be added to the cache, the node can delete any binding entry, usually the least recently used (LRU), as a strategy for cache replacement, works quite well. Four additional messages are defined to maintain the binding cache. The next few sections discuss these messages. References [1] and [14] contain a detailed explanation of these messages.

2.5.2 Binding Request Message

A binding request message is issued by a node to request a mobile node's current mobility binding from the mobile node's home agent. When the home agent receives a binding request message, it consults its home list and determines the correct binding information to be sent to the requesting node.

2.5.3 Binding Update Message

When the mobile node's home agent intercepts a packet from the home network and tunnels it to the mobile node, the home agent may deduce that the original source of the packet has no binding cache entry for the destination mobile node. The home agent should then send a binding update message to the original source node, informing it of the mobile node's current mobility binding.

2.5.4 Binding Acknowledge Message

A binding acknowledge message is used to acknowledge receipt of a binding update message. It should be sent by a node receiving the binding update message, if the acknowledge bit is set in the binding update message.

2.5.5 Binding Warning Message

When any node (for example, a foreign agent) receives a tunneled packet, if it has a binding cache entry for the destination mobile node (and thus has no visitor list entry for this mobile node), it may deduce that the tunneling node has an out of date binding cache entry for this mobile node [3]. In this case, the receiving node should send a binding warning message to the mobile node's home agent, advising it to send a binding update message to the node that tunneled this packet. The mobile node's home agent can be determined from the binding cache entry; often the home agent address is learned from the binding update that established this cache entry. The address of the node that tunneled this packet can be determined from the packet's header, because the address of the node tunneling this packet is the outer source address of the encapsulated packet. As in the case of binding update sent by the mobile node's home agent, no acknowledgment of this binding warning is needed, because future packets for the mobile node, tunneled by the same node, will cause the transmission of another binding warning. However, unlike the binding update message, no authentication of the binding warning message is necessary, because it does not directly affect the routing of IP packets to the mobile node.

2.5.6 Foreign Agent Smooth Handoff

When a mobile node moves and registers with a new foreign agent, the base Mobile IP protocol does not notify the mobile node's previous foreign agent. IP packets intercepted by the home agent, after the new registration, are tunneled to the mobile node's new care-of address. While the mobile node moves from the previous foreign agent to the new foreign agent, the packets that are received by the home agent are still tunneled to the old care-of address, because the home agent does not yet know the new care-of address. These packets are lost and are assumed to be retransmitted by higher level protocols, if needed.

As a part of the registration request, the mobile node can request that its new foreign agent attempt to notify its previous foreign agent of the move. This can be done by including a previous foreign agent notification extension in its registration request message. The new foreign agent then builds a binding update message and transmits it to the mobile node's previous foreign agent as part of registration , requesting an acknowledgment from the previous foreign agent [3]. The notification will typically include the mobile node's new care-of address, allowing the previous foreign agent to create a binding cache entry for the mobile node, to serve as a forwarding pointer. As shown in Figure 10, any tunneled packet for the mobile node that arrive at its previous foreign agent, after the forwarding pointer has been created, will then be re-tunneled by that foreign agent to the mobile node's new care of address.

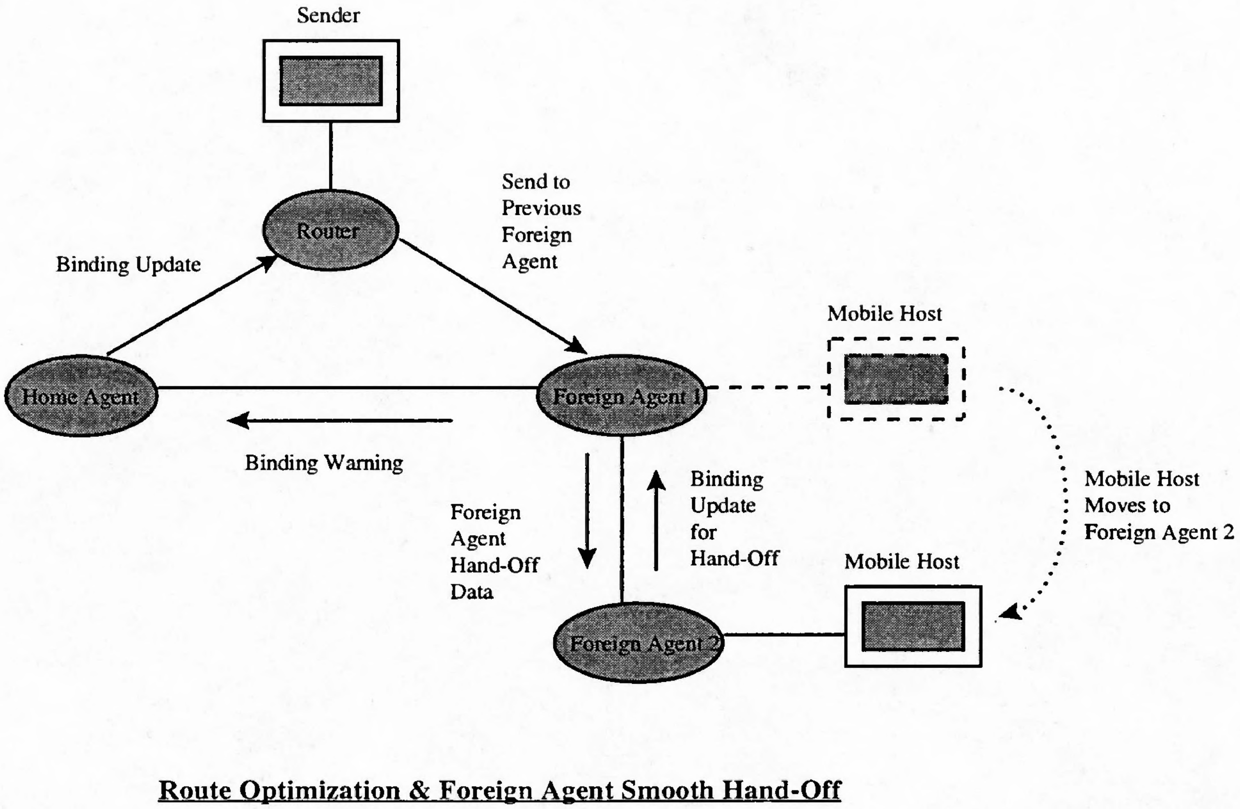

Sender
Router
Send to Previous Foreign Agent
Binding Update
Mobile Host
Home Agent
Foreign Agent 1
Binding Warning
Binding Update for Hand-Off
Foreign Agent Hand-Off Data
Mobile Host Moves to Foreign Agent 2
Mobile Host
Foreign Agent 2
Route Optimization & Foreign Agent Smooth Hand-Off

Figure 10 : Route Op and Foreign Agent Handoff

2.7 Summary

In this chapter, we introduced the essential elements of Mobile IP and Route Optimization. Our model for the simulation includes all the aspects of the protocol that are discussed here. In the next chapter we present some of the ongoing research in the area of Mobile IP.

CHAPTER 3 : RELATED WORK

Location management, which is the mechanism used to locate the intended recipient of a message, is the fundamental issue that Mobile IP deals with. There have been several proposals for location management, we present some of these proposals in this chapter. In the universities, there have also been some test implementations of Mobile IP. In this chapter, we also briefly discuss some of these test beds.

3.1 Location Management Proposals

In this thesis, we have based our study on the Mobile IP protocol, as defined by the Internet Engineering Task Force (IETF). There have been other proposals, previously defined , that have contributed to the evolution of the IETF protocol. These include, Sony's Virtual IP (VIP) proposal, the IBM proposal and the Columbia proposal. In the following sections, we briefly discuss these three proposals.

3.1.1 Sony's Virtual IP (VIP) Proposal

In this proposal ([16], [30]), every mobile host has a virtual address and a physical address, it is identified by the tuple <VN, PN>, which is called its binding. The virtual address which does not change, is the permanent address of the mobile host. The physical address is acquired by the mobile host whenever it moves to a new network. The physical address and the virtual address are the same when the host is at its home network. To send a packet, the sender uses the virtual address of the target host. The home network (router) uses a location directory (LD) to redirect packets to the current location (PN) of the target host. The mobile host informs the home network whenever it acquires a new PN. Intermediate routers that forward location updates from the mobile host to its home network, can also cache the binding. Once the binding information is available at a location, packets from that location are routed directly, without having to go to the home network.

3.1.2 IBM Proposal

In this scheme ([16], [26]), the physical location of the mobile host is defined as the IP address of a base station that it is connected to. The logical identifier is the permanent IP address of the mobile host. The binding is defined as the mapping between the permanent IP address of the mobile host and the IP address of the current base station. The physical location, in this scheme, corresponds to the IP address of the current base station whereas, in the Sony scheme, it is an acquired temporary address. In the Sony scheme, the binding information is cached by routers in the network, while in the IBM scheme, binding information is cached by the senders also.

3.1.3 Columbia Proposal

In this scheme ([16], [1]), binding is defined as in the IBM scheme. However, there is no concept of a location directory. The location directory is maintained at several Mobile Support Stations (MSS) which cover the mobile subnet within a campus. MSSs advertise reachability to the mobile subnet, and normal IP routing forwards the packets to one of several MSSs that constitute the mobile subnet. The location directory at the MSS is then used to determine the MSS with which the mobile host is registered. The MSS that receives a packet, tunnels the packet to the current MSS at the destination. The destination MSS decapsulates and forwards the inner packet to the mobile host. As in the case of Route Optimization with Mobile IP, when a mobile host moves and registers with a new MSS, the new MSS sends a forwarding pointer to the previous MSS. The previous MSS not only forwards any packets wrongly routed to it, but also sends a message to the MSS that wrongly forwarded the packet.

3.2 Mobile IP Testbeds

There have been several packet-level studies of local area networks, however, there is not much literature about such studies on a wide-area network, specially those involving mobile hosts. Since the Mobile-IP standard is still evolving, there are no known commercial implementations. However, a great deal of work is being done at the universities to set up test beds for implementing mobility. In this section, we briefly discuss some of these test beds.

The MosquitoNet Project at Stanford University, as discussed in [6], [7], and [8], is a testbed to provide continuous Internet connectivity to mobile hosts. MosquitoNet addresses the problem of maintaining connectivity when visiting foreign networks that do not explicitly support mobility for visitors. An eight-day network packet trace of MosquitoNet was performed to study factors such as the amount of user mobility between the wired and wireless networks, the amount of mobility within the wireless network, a comparison of usage patterns between the wired and wireless networks, an examination of application end-to-end delays, and an examination of overall packet loss and reordering in the wireless network [6]. To summarize the findings:

- Mobility: On average, a mobile host was found to switch between the wired and wireless networks 14 times during the eight days (with a minimum of three times and a maximum of 34). On average, a mobile host moved within the wireless network five times during the eight days (with a minimum of one move and a maximum of 14 moves). Seven distinct locations at least one-half mile apart in the wireless network were found, that were visited by mobile hosts. The widest spread locations were 70 miles apart.
- Latencies: Round trip latencies in the wireless network was found to be high, with a minimum of 0.2 seconds. Much higher end-to-end delays (up to hundreds of seconds) resulted from packet retransmissions due to loss and reordering. It was found that high latencies and the rate of packet loss and reordering prevent hosts from fully utilizing the available bandwidth of the wireless network. Telnet median delay was found to be 0.97 seconds and NFS median was 0.6 seconds. Sixty percent of telnet delays were 1.3 seconds or less, and sixty percent of NFS delays were 0.7 seconds or less. These numbers could be unacceptable, considering the fact that users normally find interactive response time slow when it exceeds 100 to 200ms.
- Optimizations: It was found that changing telnet from character mode to line mode, improves its interactive response, by requiring 50% fewer round trips. This allows an entire line to be typed and possibly edited before a packet is sent.

In order to support mobile computing research, including the development of software which will allow seamless access to multiple wireless data networks, the Monarch Project as in [10] and [11], is a wireless data network infrastructure being built at Carnegie Mellon University. This project would allow researchers and other members of the campus community to use mobile computers to gain access to networks while on-campus or off in the Pittsburgh area.

Mobile-IP for Linux, as discussed in [2] and [5], is a project at the State University of New York, to create a testbed for Mobile IP, using the Linux Operating System. It is a full implementation of the Mobile IP protocol.

BARWAN (Bay Area Research Wireless Access Network), as discussed in [29], is a wireless network testbed created in the San Francisco Bay Area, by researchers at the University of California at Berkeley. Pilot applications are being developed to drive the design and validation of the interfaces between applications and this network.

During tests with WalkStation II [9], it was found that the time to send a packet to a mobile node when it was at a foreign agent, was roughly 3.5 times the time to do the same when the mobile node was at home. This was attributed to the increased processing time at the sender, the home agent and the foreign agent, the propagation time is also longer. The break-up of the various steps involved showed that the time to administer a packet at the home agent and the foreign agent constitutes a large part of the total time to deliver the packet to the mobile node. It was also observed that over 98% of the time spent in registration is spent during the first phase, where the mobile node is trying to contact a foreign agent. Part of the explanation is that there are a number of packets that have to be sent at the Data Link Layer, to establish LAN addresses, before the first registration request can be sent. Besides, about 4 solicitation requests had to be sent before an agent advertisement was received.

3.3 Thesis Contribution

As can be seen from the foregoing discussion, most of the existing research work deals with implementing Mobile IP in a campus wide network. In this thesis, we have attempted to simulate Mobile IP and Route Optimization in a large network. Since, at present, we do not have the infrastructure to implement Mobile IP, it is assumed that future implementations can take advantage of the results from this simulation study.

Our simulation model incorporates all aspects of Mobile IP, including Registration, Agent Advertisement, Agent Solicitation, Registration Extension and De-Registration. We model Route Optimization over base Mobile IP. This model is used to study the performance of Mobile IP by varying different network parameters. The model of Route Optimization includes, Binding Warning, Binding Update and Foreign Agent Handoff. By simulating Mobile IP, with and without Route Optimization, we demonstrate the relative advantages of using a Cache.

3.4 Summary

The IETF Mobile IP protocol that has been used in our simulation study, is derived from the various proposals for location management namely, Sony's VIP proposal, the IBM proposal and the Columbia proposal. In this chapter, we have briefly discussed these three proposals. This chapter has also reviewed some of the research work that is being conducted at the universities to set up test beds for studying the Mobile IP protocol. We have also stated that unlike other research work, this thesis has attempted to study Mobile IP, by simulating it in a large network. In the next chapter, we present the architecture and design of our simulation system.

CHAPTER 4 : SIMULATOR SYSTEM ARCHITECTURE

In this chapter, we present the implementation details of the simulator we built for our thesis study. This event-driven simulator incorporates all the features of the Mobile IP protocol, along with Route Optimization. It is implemented over a simulated network of Routers, Agents, Mobile and Stationary Hosts, arbitrarily connected by communication Links, as shown in Figure 11. It was developed as a console application in Visual C++ under Win 95. Since it does not have any Windows specific code, it can be easily ported

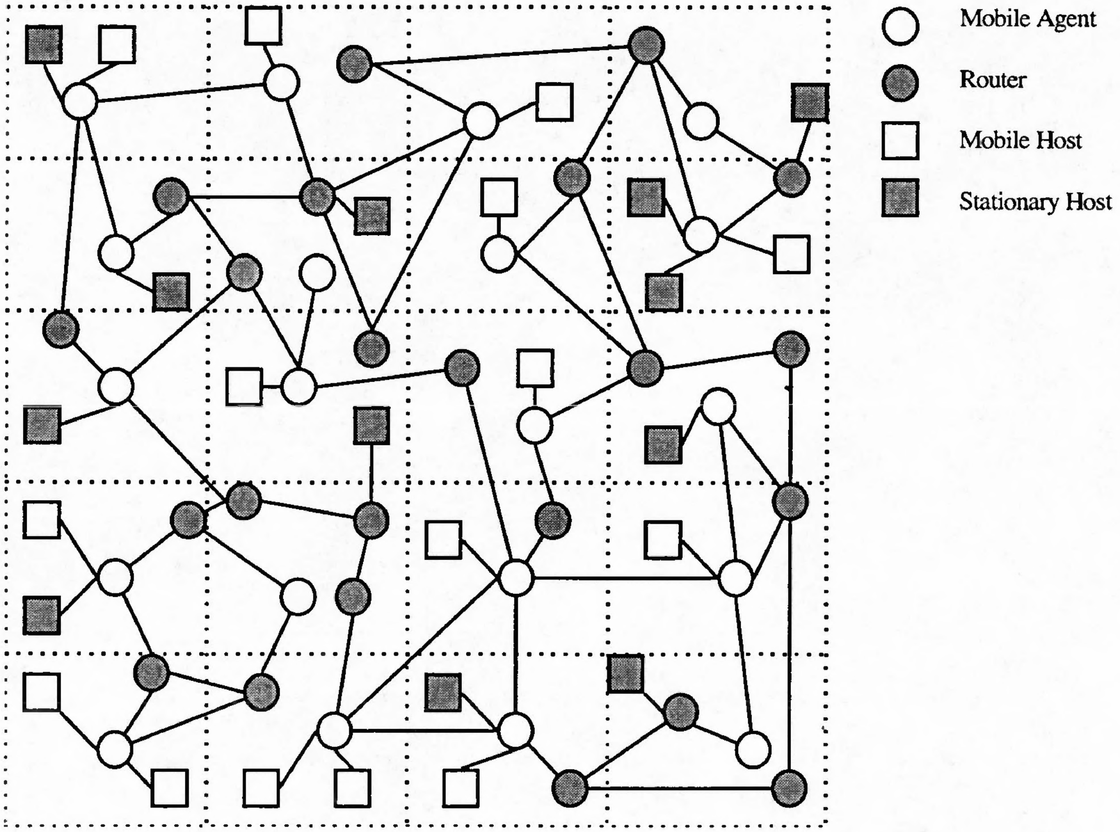


Figure 11 : Sample 4 x 4 Grid Network

to any other environment, including the Unix environment. The architecture of this simulator can be discussed in the context of the objects, the events, and algorithms used in building the network.

Our conceptual model of the network, as shown in Figure 11, is a large area that has been divided into cells. Each cell has one Mobile Agent and any number of ordinary Routers. Mobile Agents act as Home Agents for all the Mobile Hosts that belong to its cell. Mobile Agents also act as Foreign Agents for hosts that visit its cell. Each Mobile Host is connected to the Mobile Agent in its cell. Stationary Hosts may be connected to a Mobile Agent or a Router. Mobile Hosts, randomly move from cell to cell. As they move to a new cell, Mobile Hosts register themselves with the Mobile Agent in that cell.

4.1 Objects Used in the Simulation

Objects used in this simulation are Router/Agent, End Node, Link and Packet. These objects are discussed in the following sections.

4.1.1 The Router/Agent Object

The following table shows the private data members of the router/agent object.

Address	The Address of the router.
Cell	The cell that this router/agent is in. This field is only relevant for mobile agents.
LastAdvrtTime	The time in simulated clock ticks, when this agent last advertised its presence. This field is only used by the mobile agents.
RoutTable	Ponter to the Route Table.
AdjList	Pointer to the Adjacency List of this router/agent.
MblList	Pointer to the list of mobile nodes that have this agent as their home agent.
VisitorList	Pointer to the list of visiting mobile nodes.
PktList	Pointer to the list of packets that are waiting to be processed.
TimeToSend	The minimum time that this router would take to process the first packet from the list of packets.
CachePtr	Pointer to the binding cache. Used only when route optimization is implemented.
CacheNum	Number of entries in the cache that are currently used.

Some of the significant methods implemented for this object are as follows,

ReceivePacket

This method is invoked from a link, whenever there is a packet to be received by this router/agent. The received packet is queued in the list of packets. An event to send a packet is generated, to be processed at a time which is equal to the sum of the time to send from this router and a factor of the queue length.

SendPacket

This method is invoked whenever the simulator comes across an event to send a packet from a router. Based on the type of the first queued packet, this method does a variety of actions. If the packet is of type DATA, and the to router address (corresponding to the network prefix), is the same as the self address, then the packet is forwarded to the destination end node. If the destination end node is mobile, and no longer at home, then the packet is tunneled to the foreign agent. When the destination end node is found in the cache, and the stay time is still current, the packet is directly tunneled to a distant router/agent. Other types of packets, which are listed later under packet types, result in other actions. A BINDING_UPDATE packet, for instance, results in the corresponding mobile node address being entered in the cache, with the new stay time.

Advertise

This method is called periodically to make the mobile agent change its last advertised time. Mobile nodes that move into an agent's cell, check that this advertised time is within reasonable limits, before trying to register with the agent.

BindingUpdate

This routine is called whenever a BINDING_UPDATE packet is processed. If the mobile node does not already exist in the cache, then it is added there, otherwise the existing cache entry is changed to reflect the new binding. If there is no room in the cache to make a new entry, then the entry with the least last access time is cleared to make room for the new one.

BindingWarning

This routine is called whenever a foreign agent receives a packet destined for a mobile node that is no longer registered with it. It sends a BINDING_WARNING packet to the home agent of the related mobile node.

LookInCache

Before forwarding a packet, every router/agent calls this function to check if the mobile node that a packet is to be sent to, has an unexpired binding cache entry.

DeRegister

This routine is called whenever a mobile node returns to its home agent. It results in the mobile node being taken off the mobile list, and re-entered in the adjacency list.

4.1.2 The End Node Object

The following table shows the private data members of the End Node object.

Address	Address of the end node.
MobileFlag	Set to TRUE/FALSE for whether the end node is mobile.
HomeAgent	The home agent of a mobile node or the default router of an immobile node.
ForeignAgent	Used only in case of mobile nodes, this is the address of the foreign agent, when the mobile node is away from home.
CurrCell	The current cell that the mobile node is in.
MsgsToSend	The list of messages that are queued to be sent from this end node. Each message could have one or many packets.
NumPktsSent	Number of packets sent from this end node. Required to gather statistics.
NumPktsReceived	Number of packets received by this end node.
LinkNum	The number(address) of the link that connects this end node to its default router or the home/foreign agent.
StayTime	The amount of time that this mobile end node would stay with each foreign agent, as it is moving.
TimeToSend	The time that this end node takes to process and send a packet.
MaxCellsToMove	The maximum number of cells that this mobile end node moves away from its home, at any time.

NumCellsMOved	While it is moving, this field keeps track of the number of cells that the mobile node has already moved.
RespReceived	A flag to indicate whether a response has been received for the registration request. The mobile node periodically sends registration requests till a response is received from the foreign agent.
MoveDirection	The direction that the mobile node is currently moving in, it could be UP/DOWN/RIGHT/LEFT.

Some of the key methods implemented for the end node object are as follows,

GenerateMessage

This method is called initially after the network has been set up, to create the messages at some end nodes. Each message has a number of packets. If there was no other message at the end node, this method also creates an END_NODE_SEND event for the end node, to start sending packets from there. This method is also called in response to a GEN_MESSAGE event.

Move

Invoked on a MOVE_MBL_NODE event, this function makes a mobile node to change its cell. The direction to move in is an argument to this function, which can be UP/DOWN/LEFT/RIGHT. In case of a mobile node that has already moved once, the direction can be CONTINUE, which indicates that the mobile node should continue moving in the same direction that it was moving in earlier. On reaching the boundary of the simulated area, this function reverses the direction of move of the mobile node. If an advertisement is not available from the agent in the cell that the mobile node is moving to, this function generates a message to solicit advertisement from the agent. On receiving a valid agent advertisement, a registration request is sent and an event of type DECIDE_EXTEND_STAY is generated, so that, at the end of the desired stay time, the mobile node can either decide to extend its stay, or move again. An event of type RE_REG_REQUEST is also generated to be processed after a set time, so that, if a registration reply is not received by then, another registration request can be sent by the mobile node.

SendPacket

This method puts the first packet from the list of messages, on the link to the router/agent. The link is considered INCOMPLETE till a registration request has been received from the foreign agent. Only REG_REQUEST and SOLICIT_ADVERT kind of packets can be put on an INCOMPLETE link. This method also generates the next END_NODE_SEND event, if there are more packets remaining to be sent from the end node.

ReceivePacket

This function is invoked by the link that the end node is connected to. The number of received packets is incremented here. If the packet received is of type

REG_RESPONSE, the reply to a registration request, then a flag is set to indicate that no more registration requests need be sent to the foreign agent.

RcRcgRcqucst

This method is invoked in response to the RE_REG_REQUEST event, it results in another registration request being sent to the agent in the current cell. The registration request is not sent if a reply has already been received.

4.1.3 The Link Object

The following table shows the private data members of the Link object.

LinkIndex	A number assigned to the link, to identify it
Type	FAST/SLOW, can be used to randomly make some links better than others, not used currently.
Status	VALID/INVALID/INCOMPLETE, INVALID links are not used in this version, however it can be used to simulate broken links. INCOMPLETE links are used to disallow DATA packets over wireless links, till registration reply has been received.
WireLessFlag	Flag to indicate whether the link is wireless. Mobile nodes are connected to the network using Wireless links.
Addr0	The node at one end of the link.
Addr1	Node at the other end of the link.
PktsFromAddr0	List of packets sent from one end of the link.
PktsFromAddr1	List of packets sent from the other end of the link.
TimeToSend	Time taken to transmit over the link. Can be used to set different times for different types of links. Was used to make the wireless link slower, relative to wired links.

Some of the significant methods of the Link object are as follows,

ReceivePacket

This function is called from a router/agent or an end node, whenever a packet has to be sent across the link. The packet is queued on one of the lists for received packets. It results in a LINK_SEND event being generated. The LINK_SEND event is set up to be triggered after some time. This time takes into account the time taken to transmit a packet over the link.

SendPacket

This method is triggered by the LINK_SEND event. If there are any packets in the two lists of received packets, then the first ones of these are sent across to the node at the other end.

ChangeAddr

This method is used to change the address of one of the nodes at the end of a link. It is used when a mobile node moves from one cell to another, because the address of the agent at the other end of the link has to be changed. Whenever this method is invoked, all the packets that are queued at the link are dropped, and are assumed to be lost.

4.1.4 The Packet Object

The following table lists the private data members of the Packet object.

FromAddr	The address of the end node that sent the packet.
ToAddr	Address of the end node that the packet has to go to.
EndRouter	The router/agent that the ToAddr is attached to. This corresponds to hierarchical addressing, where the packet first goes to the EndRouter, and the EndRouter delivers it to the end node that is connected to it.
Type	The type of the packet. The different types are defined below.
Priority	HIGH/LOW. This field is meant to allow priority processing of some packets. DATA packets are always of LOW priority. HIGH priority packets are queued ahead of LOW priority packets.
TnldFlag	Flag to indicate whether the packet has been tunneled.
TnldFromAddr	The address of the agent/router that the packet was last tunneled from. This would have the address of the home agent whenever the packet is tunneled from the home agent and sent to a foreign agent.
Size	The size of the packet, not used currently.
HopCount	The number of hops that the packet has made in its journey.

4.1.4.1 Types of Packets

The different types of packets used in this simulation are as follows,

DATA

This, as the name suggests, is the most common type of packet.

REGREQUEST

This is the registration request packet. It has the address of the home agent as one optional field in it.

REGRESPONSE

This packet is sent from the foreign agent to the mobile node, as a confirmation of successful registration.

INFORMMOVE

As a continuation of the registration process, the foreign agent, on receiving a registration request, sends this packet to the home agent, to inform that the mobile node is requesting a new registration. The address of the mobile node is an optional field in this packet.

RESPINFOMOVE

The home agent sends this response to the foreign agent to confirm that the new registration has been received and established.

SOLICIT_ADVERT

This packet is sent by a mobile node to the agent in the current cell, to request an advertisement. It happens when a mobile node visits a new cell and finds out that the last advertisement from the agent there was too old.

EXTEND_STAY

Some mobile nodes decide to extend their stay in a cell beyond the time that was conveyed during registration. This packet is sent by such mobile nodes to inform the foreign agent of its decision to extend the stay. The simulator has been set up such that one out of four mobile nodes decide to extend their stay.

INFORM_EXTEND_STAY

On receiving an EXTEND_STAY packet from a mobile node, the foreign agent sends this packet to the home agent, to make the home agent update the stay time for that mobile node. The home agent, otherwise, does not tunnel packets to the foreign agent when the stay time for a mobile node has expired.

BINDING_WARN

This packet is sent by a foreign agent to the home agent whenever it receives a packet for a mobile node that is no longer with it. Such a situation happens when an intermediate router, after finding a valid entry in the cache, tunnels packets to a previous foreign agent.

BINDING_UPD

This packet is sent by a home agent to an intermediate router, whenever the home agent receives a BINDING_WARNING, or when it receives packets for a mobile node that are no longer with it. It is meant to inform the sending router of the new mobility binding for the mobile node. On receiving this packet, the sending router updates its cache.

HAND_OFF

This packet is similar to BINDING_UPD, it is sent by a mobile nodes new foreign agent, to the previous foreign agent, to inform it of the new binding, so that the previous foreign agent can tunnel any packets to the mobile nodes current location.

4.2 Events in the Simulation

The different types of events that are handled in this simulation are as follows,

END_NODE_SEND

This event causes one packet from the first message queued at the end node, to be sent over the link that connects it to its default router, which in the case of a mobile node, could be a foreign agent. This event is generated whenever a new message is added to the queue of messages, that has no other queued messages. It is also generated at the end of processing of this event, when there are more messages in the queue, or there are more packets in the current message. This event is processed some time after it is generated, the time determined by a variable, can be set to model the time taken by the upper layers in the protocol stack to process a packet.

LINK_SEND

This event causes packets that are queued at a link, to be sent across the link. Since links are bi-directional, there are two queues at a link. The first packet from each queue, if the queue is not empty, is sent across to the node at the other end. This event is generated whenever a link receives a packet. On receiving a packet, the LINK_SEND event can be set up to fire at some time in the future. This time, which is an input parameter to the simulator, is meant to model the different transmission speeds of links that are found in wide area networks. For instance, it was used to vary the time to send from a wireless link, with respect to a wired one.

ROUTER_SEND

This event causes the first packet from the queue of packets, at a router, to be processed. Based on the type of packet, this could mean different kinds of processing. A DATA packet, would be put on the next link towards it's destination. A REG_REQUEST packet, results in a foreign agent sending an INFORM_MOVE packet to the home agent. This event is generated whenever a router or agent receives a packet. It is set to be triggered, some time after the receipt. This time is made up of the time to process a packet, it also factors-in the load, by taking the queue length into account. A small amount of time, involved in looking up the cache, was also included to model possible delay due to cache lookup.

MOVE_MBL_NODE

This event triggers the movement of a mobile node. The direction to move in is decided when the event is generated, and is available as part of the event. This event results in the mobile node changing its current cell and registering with a new foreign agent.

ADVERTISE

This event is generated by a mobile node, to get a fresh advertisement from the agent in its current cell. It results in the mobile agent changing its last advertised time to the current time.

DECIDE_EXTEND_STAY

This event is generated when a mobile node has successfully registered with a foreign agent. It is set up to be triggered a short time before the mobile node's stay time expires. On receiving this event, the mobile node either sends an EXTEND_STAY packet to the current foreign agent or generates a MOVE_MBL_NODE event, to move to another cell at the end of the current stay time. The simulator is currently set up such that one out of every four mobile nodes, decides to extend its stay.

CHECK_EXPIRE_VISIT

This event is generated by a foreign agent whenever a mobile node has successfully registered with it. It is triggered a short time after the expiry of the mobile node's stay time, and results in the mobile node being taken off the visitor list. If the mobile node had, by then, extended its stay, then it is not taken off the visitor list till the new stay time expires.

4.3 Algorithms used in Building the Network

4.3.1 Components and Connectivity

Before the static routing table is built, it is ensured that the graph formed by the routers/agents is connected. The algorithm [13], flowchart for which is shown in Figure 12, identifies the different components, by merging adjacent vertices. This algorithm also connects all the components together, so that there is a path between any two pairs of vertices.

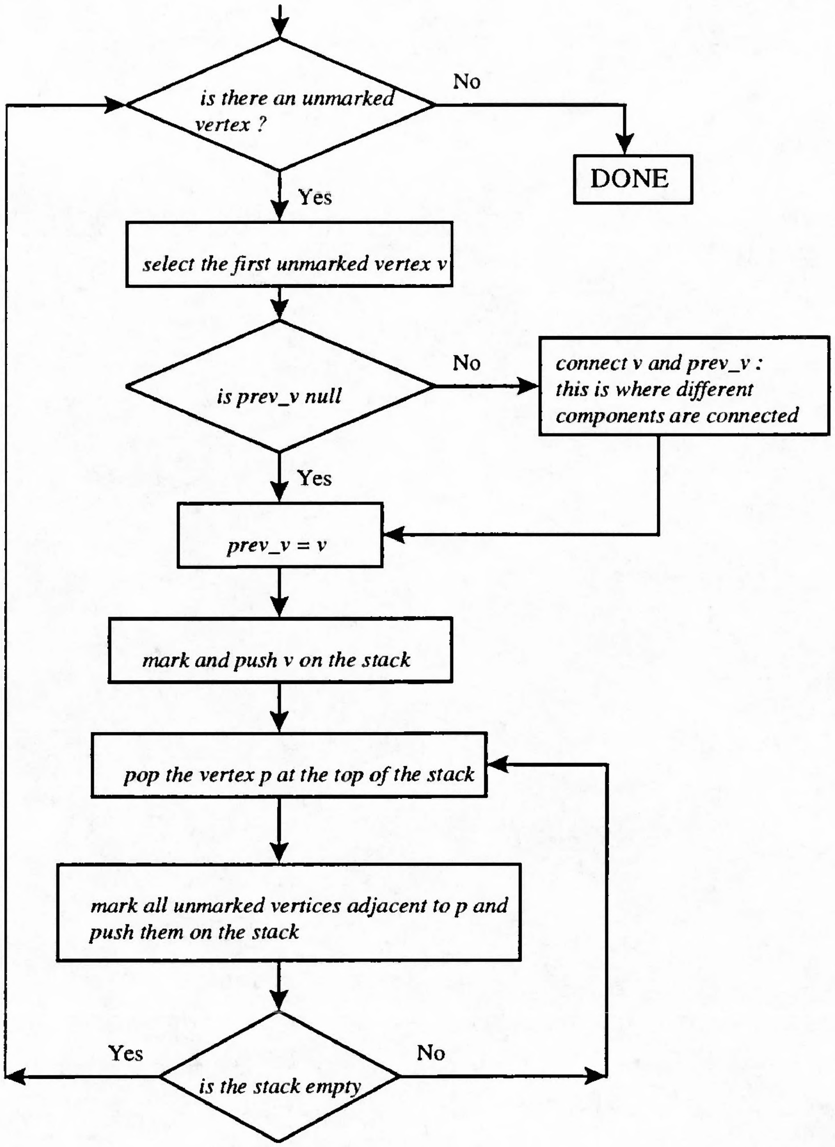

is there an unmarked vertex ?
No
DONE
Yes
select the first unmarked vertex v
is prev_v null
No
connect v and prev_v : this is where different components are connected
Yes
prev_v = v
mark and push v on the stack
pop the vertex p at the top of the stack
mark all unmarked vertices adjacent to p and push them on the stack
Yes
is the stack empty
No

Figure 12 : Algorithm to Connect the Network

4.3.2 Shortest Path Algorithm

The Warshall-Floyd Shortest path algorithm, flowchart for which is shown in Figure 13, is used to build the Static Routing Table. This algorithm works by inserting one or more vertices into paths, whenever it is advantageous to do so. This algorithm has a complexity of N^3, where N is the total number of routers/agents.

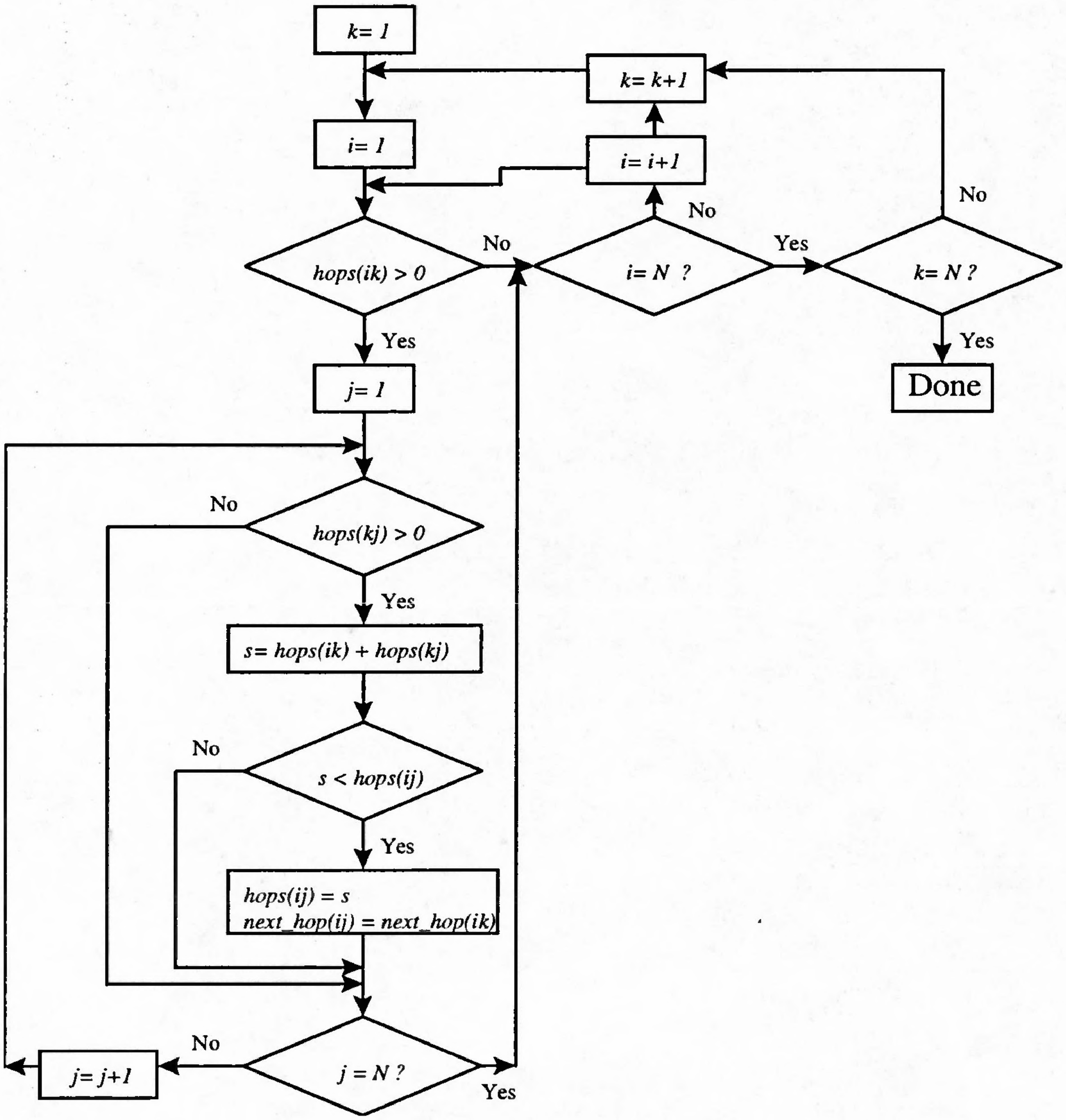


Figure 13 : Shortest Path Algorithm, used to Build Routing Table

4.5 Network Setup

The number of cells in the mobility supporting network is the number of rows times the number of columns. With one agent in each cell, the number of agents is equal to the number of cells. Agents also act as routers. The design provides for having other routers which are not mobile agents. These other routers are randomly distributed over the area under consideration. The routers and agents are then randomly linked together, using the algorithm that was explained earlier. The maximum and the minimum number of links per router/agent is a variable, which can be used to vary the density of interconnection of the network. Another variable is used to control the extent, in terms of cells, that a link can stretch, this parameter is used to control the span of links, long links vs short links. The adjacency list of each router/agent gets built as the links are added. After all the links are created, the algorithm ensures that the graph is completely connected, i.e. there is at least one path between any two router/agent.

After the required number of links have been created, the routing table for each router/agent is built using the shortest path algorithm that was explained earlier. Since this simulation uses static routing , the routing table does not undergo any change after it has been created.

After all the routers/agents are created and linked together, and the route tables have been created, the end nodes are added to the network. The total number of end nodes and the percentage of this total, that are mobile, are variables. These numbers can be used to vary the number of end nodes and the amount of mobility. Each end node that is mobile, is randomly assigned to a cell, the agent in that cell, therefore, becomes the home agent of that mobile node. The agent and the mobile node are connected using a wireless link. End nodes that are not mobile, are randomly connected, using a wired link, to any router/agent. It is assumed that there is only one link connecting an end node to the network, therefore, each mobile node has one agent , and each immobile node has one and only one default router.

4.6 Simulation Run

After the network has been set up, the simulation is started. Initially, a MOVE_MBL_NODE event is set up for each mobile node, to initiate mobile node movement. Messages are also created at some end nodes to start sending packets. The number of initial messages is given by an input parameter. This parameter is used to control the load on the network. Each one of these messages can have some DATA packets. The number of DATA packets per message is also an input parameter. As these initial messages are created, END_NODE_SEND events are generated, to start sending packets from those nodes. The simulator was coded such that the sender of DATA packets is always an immobile node and the receiver is a mobile node. This was done to study the worst case effect of mobility. It was also assumed that, in real life, mobile nodes would normally be receiving information from stationary servers.

We ran the simulation many times to experiment the effect of various system input parameters, and thus studied the performance of Mobile IP, with cache (Route

Optimization) and without cache. The system input parameters and results of various experiments are discussed in the next chapter.

4.7 Summary

In this chapter we have presented the architecture of the simulator. The model of Mobile IP is built over a simulated network of Routers, Mobile Agents, Hosts and Links. The architecture allows us to vary different network parameters to study the performance. Some of the graph-theoretical algorithms that are used in building the simulated network have also been discussed here.

CHAPTER 5 : EXPERIMENTATION AND RESULTS

A series of simulation experiments were conducted to study the influence of changing network parameters on the performance of Mobile IP. Each experiment addresses one changing parameter.

5.1 Network Parameters

Although Route Optimization was added to our model, as an extension to base Mobile IP, we would not consider it one of the network parameters, because it is part of all the experiments. Route Optimization was disabled only when there had to be a comparison with the base Mobile IP. The following network parameters were adjusted to evaluate the performance of the system.

1. Stay Time : As the mobile nodes move from one cell to another, there is a certain amount of overhead in terms of the extra packets that have to be processed for registering. Besides, before successfully registering with an agent, there are periods in time when the mobile node is disconnected from the network, which could result in packets being lost. Therefore, the time that a mobile node spends in each cell can influence the performance of the network.
2. Density of Connection : This parameter is used to study how the number of links per node influences the performance of Mobile IP. If the network is sparsely connected, the distance between nodes would not reflect their true geographical distance. In such a case, Route Optimization with Cache, may not perform very well.

3. Maximum Cells Moved : This is the maximum number of cells, from the home cell, that a mobile node would move to. This is an important parameter because, it is generally felt that Route Optimization with Cache would not show a significant advantage, unless mobile nodes move far away from home.
4. Distance between Sender and Receiver : If the mobile node is corresponding with another node (server) that is close to it, would there be any advantage in implementing Cache? If the mobile node is far from the server, is there any significant reduction in the hops per received packet ?
5. Size of Cache and Cache Look Up Time : Under very heavy traffic, there could be frequent updates to the entries in the cache. Because of the LRU scheme, some cache entries would be purged before they are properly used, leading to more BINDING_UPDATES. This could be solved by increasing the size of cache, however, increasing the cache size may lead to more time being spent in looking up the cache!
6. Number of Mobile Nodes : Of the total number of end nodes (hosts), the percent that is mobile, can be changed to study the effect of mobility on the performance of the network.
7. Relative Time to Send over Wireless Link : Since it is assumed that mobile nodes are connected to their agents over wireless links, and there is a significant overhead in making registrations, the speed of the wireless link is a crucial factor in determining the performance of Mobile IP.

8. Errors over the Wireless Link : For the same reasons, as those under Time to Send over Wireless Link, errors over the link, leading to lost REG_REQUEST and REG_REPLY, can severely impact the performance of Mobile IP.

5.2 Performance Measures

In this thesis, we conducted various experiments and measured the performance of Mobile IP, with cache(Route Optimization) and without cache, in terms of the percentage of packets delivered and the average number of hops taken by each packet to reach the destination. In one of the experiments, we also measured the overhead in terms of the number of extra packets needed to support route optimization. We grouped our experiments in the following categories,

- Experiment with Cache Size
- Experiment with Distance Moved away from Home
- Experiment with Stay Time
- Experiment with Distance between Sender and Receiver
- Experiment with Density of Interconnection
- Experiment with Percentage of End Nodes that are Mobile
- Experiment with Speed of Wireless Link
- Experiment with Time to Lookup Cache
- Experiment with Errors in the Wireless Link.

These experiments and a discussion of results from each is presented in the following sections.

5.3 Experiment with Cache Size

Figures 14 and 15 show results from the test conducted by varying the number of entries in the cache. As shown in Figure 14, the average number of hops taken by a packet goes down as the number of possible entries in the cache goes up, this however depends on the load on the router, because if the router is not heavily loaded, there would not be any need to have a lot of entries in the cache. There is also no need to have any entries in the cache for mobile nodes that have changed their point of attachment, the entry in the cache can not be used because the stay time would have expired. Figure 15 shows the percentage of total packets tunneled directly from the source's router, relative to the number of entries in the cache. This figure reflects the direct gains from avoiding triangular routing.

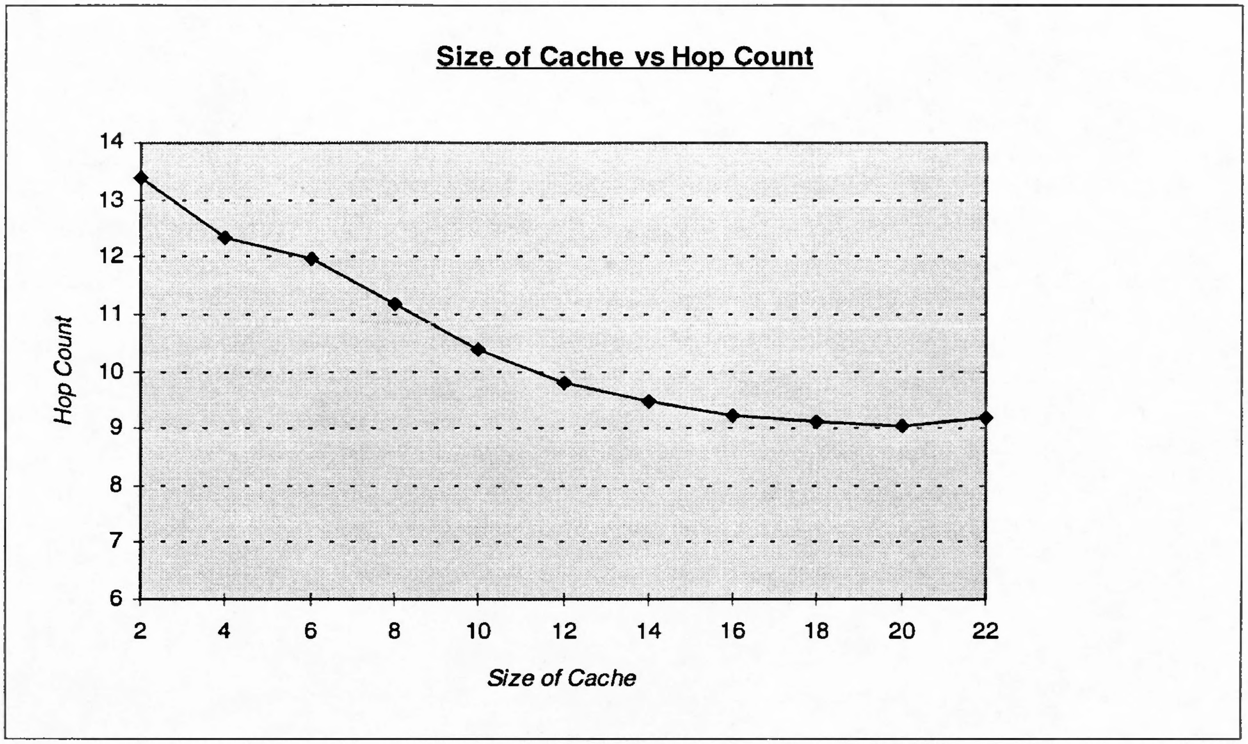


Figure 14 : Size of Cache vs Hop Count

Parameters Used : (10 x 10 Grid Network , Number of Messages simultaneously sent = 15, Average distance between sender and receiver = 8 hops, Maximum cells moved = 10, Maximum Links/Router = 4)

- Hop count reduces as the Cache size increases. But Hop count levels off after a certain point, because the large cache size would not get used if there are not so many packets simultaneously sent from that Router.

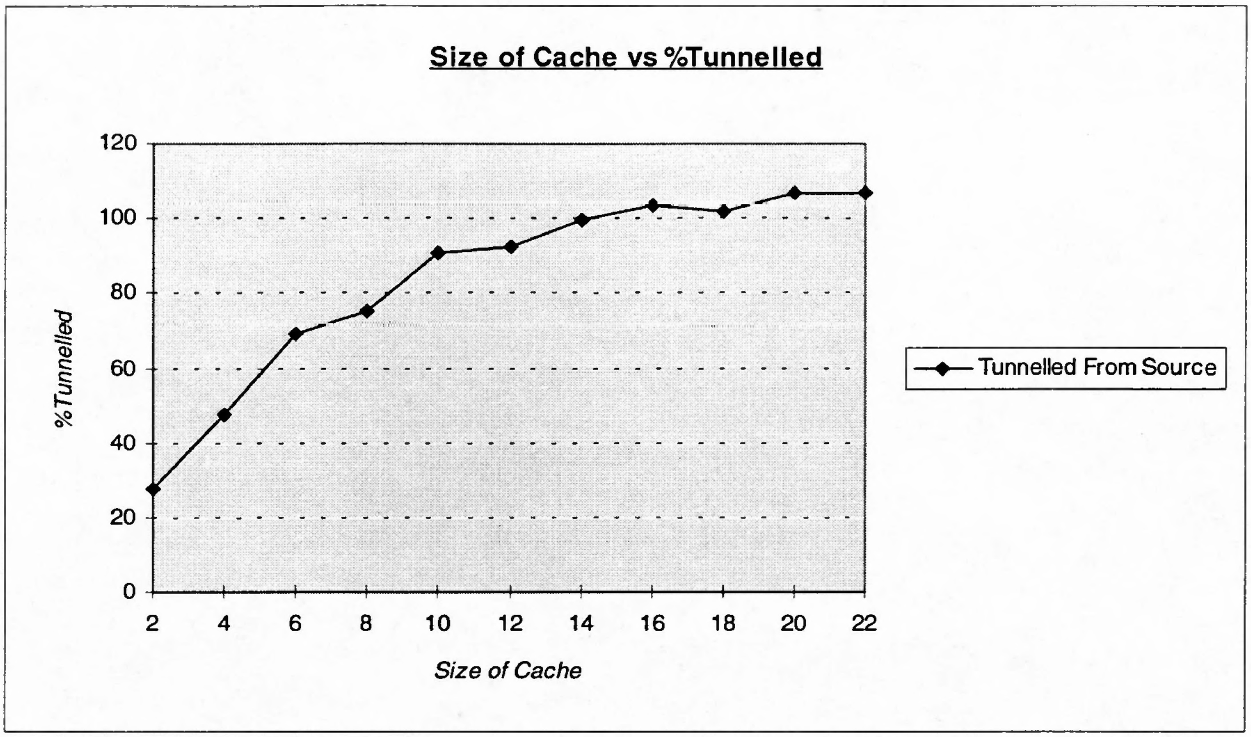


Figure 15 : Size of Cache vs %Tunneled

Parameters Used : (10 x 10 Grid Network , Number of Messages simultaneously sent = 15, Average distance between sender and receiver = 8 hops, Maximum cells moved = 10, Maximum Links/Router = 4)

- Percentage of packets tunneled directly from the source goes up as the cache size goes up.
- After some point, the increase in Cache size does not result in any savings, in fact some of the packets tunneled from the source are lost, as shown in Figure 15, where the percentage goes over 100%. This means that some of the Packets that are tunneled directly from the source are lost, and so do not reach the destination.

5.4 Experiment with Distance Moved Away from Home

Figures 16 and 17 show the impact of the distance traveled from home on the performance. As shown in Figure 16, Mobile IP with cache does much better when the mobile node moves far away from its home. However, if mobile nodes stay close to home, the existence of cache does not result in a lot of gain. Figure 17 shows that there is very little variation in Hops/Received packet even when mobile nodes move farther away from home.

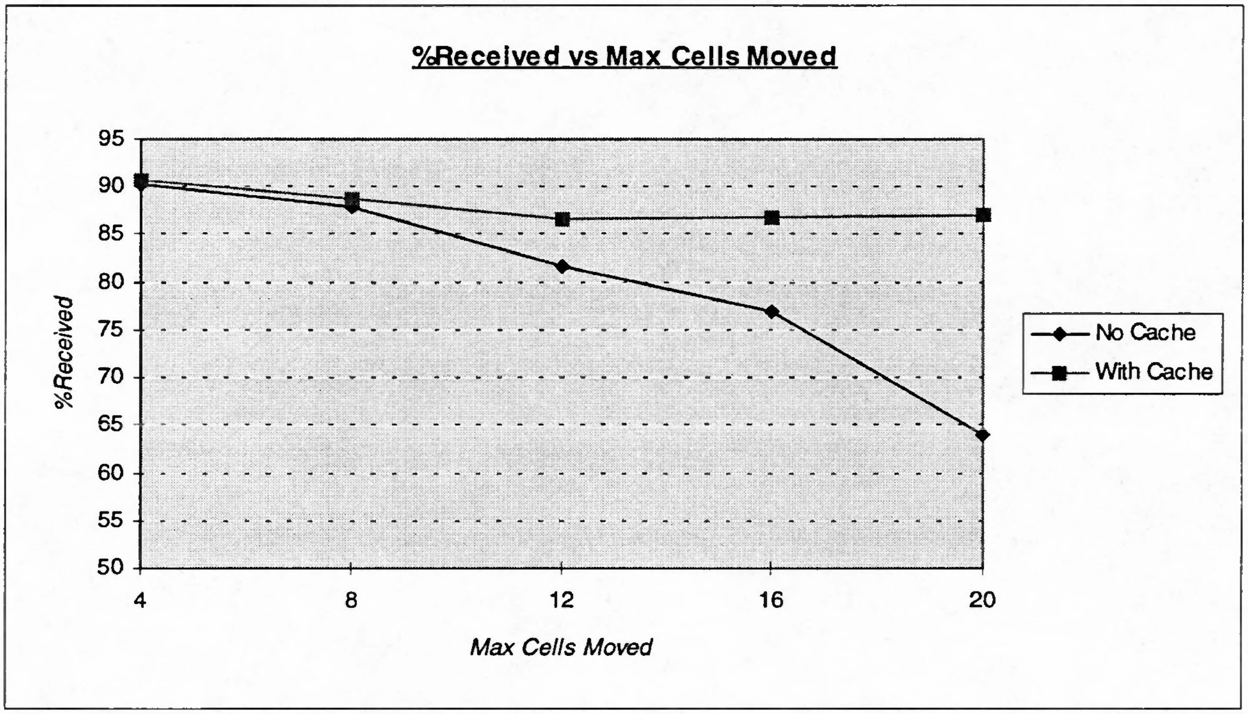


Figure 16 : %Received vs Max Cells Moved

Parameters Used : Run Time = 6000 units, 20 x 20 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 20 hops, Maximum Links per Router = 4

- As the Mobile Nodes move farther away from home, with Route Optimization (cache), the percentage of packets received does not drop as rapidly.

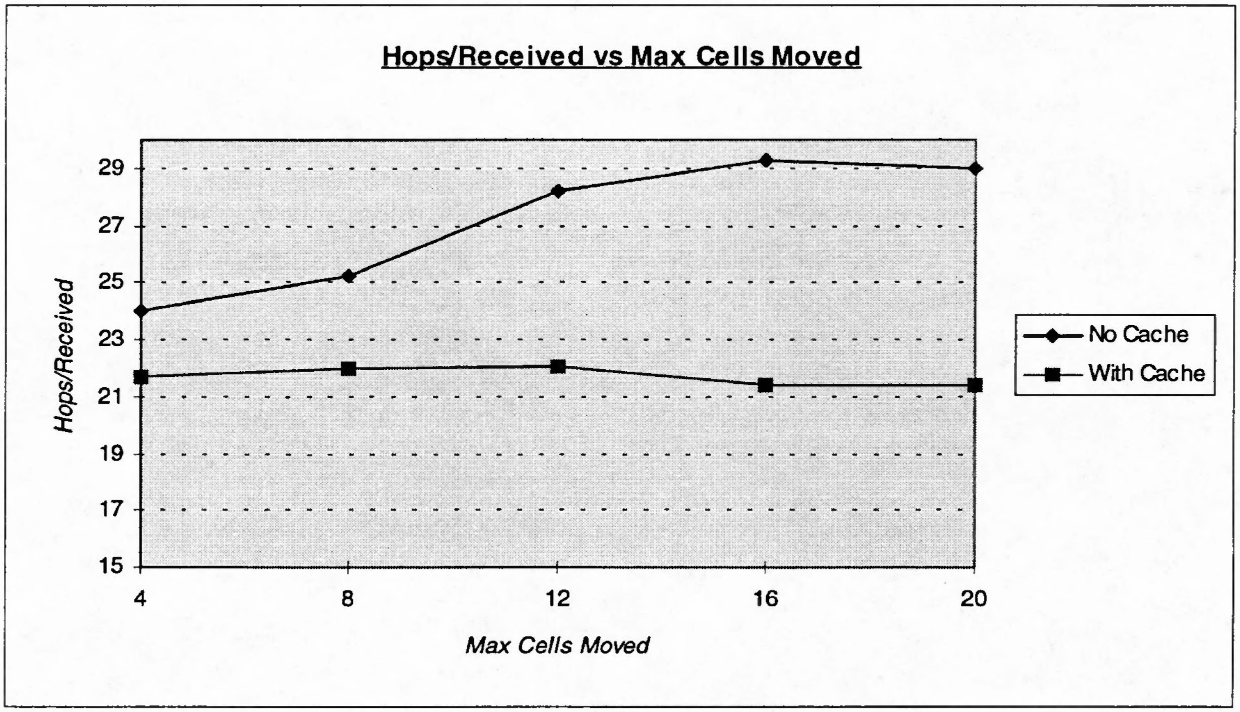


Figure 17 : Hops/Received vs Max Cells Moved

- With Route Optimization, the variation in Hops/Received is very little, even as the Mobile Nodes move farther away from home.

Figure 18 shows the effect of the mobile node moving away from home, on the simulated time taken to deliver packets. This time does not change much, when route optimization is implemented.

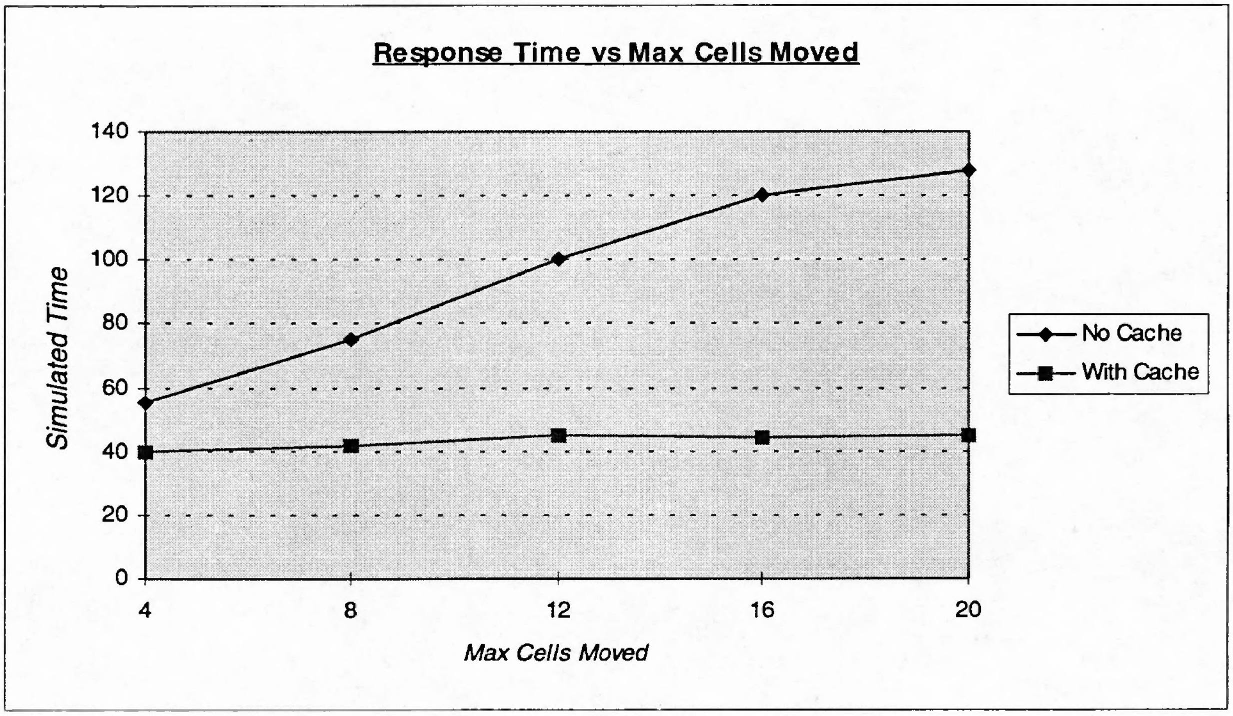


Figure 18 : Response Time vs Max Cells Moved

Parameters Used : Run Time = 6000 units, 20 x 20 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 20 hops, Maximum Links per Router = 4

5.5 Experiment with Stay Time

The effect of Stay Time on the performance of Mobile IP is shown in the Figures 19, 20, 21 and 22. Figure 19 shows that the number of packets received goes up as the Stay Time goes up. In fact, the percentage of received packets goes up by as much as 30% when the stay time changes from 100 units to 500 units. When the Stay Time is low, i.e. when mobile nodes change their point of attachment quickly, more packets are lost because there are points in time, just before a successful registration, when neither the foreign agent nor the home agent knows the location of the mobile node, it would be a good idea to make the mobile agents cache some packets till the forwarding address is known. Figure 20 shows that when the Stay Time goes up, with Route Optimization, the Hops/Received packet goes down, because more and more of the packets that are directly tunneled are able to find their destination.

Figure 21 shows that as the stay time increases, the number of control packets necessary to support Mobile IP goes down. This is because mobile nodes do not register as often, and also because there are fewer number of BINDING_UPDATE and BINDING_WARNING messages. Similarly, Figure 22 shows that the overhead per received packet also goes down as the stay time goes up.

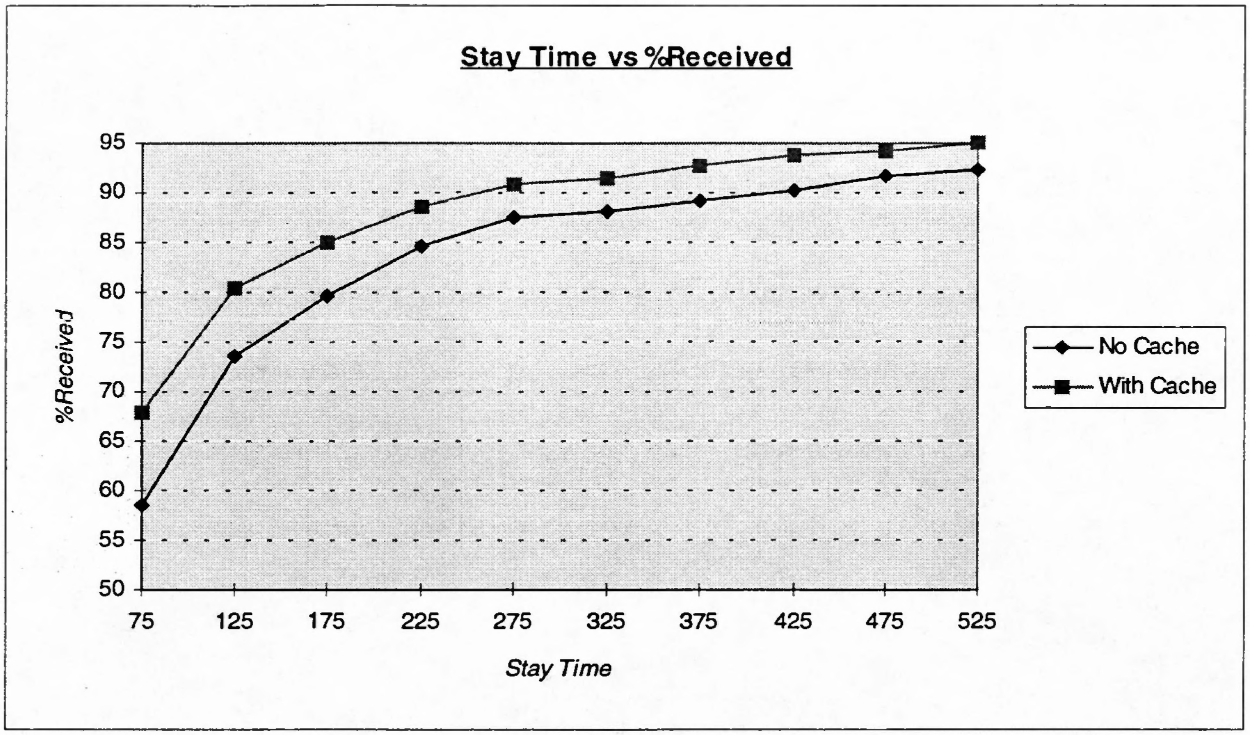


Figure 19 : Stay Time vs %Received

Parameters Used : Run Time = 6000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops, Maximum Links per Router = 4, Maximum Cells Moved=10

- %Received goes up as the Stay Time goes up, however after some time any increase in Stay Time does not have significant effect on the %Received.

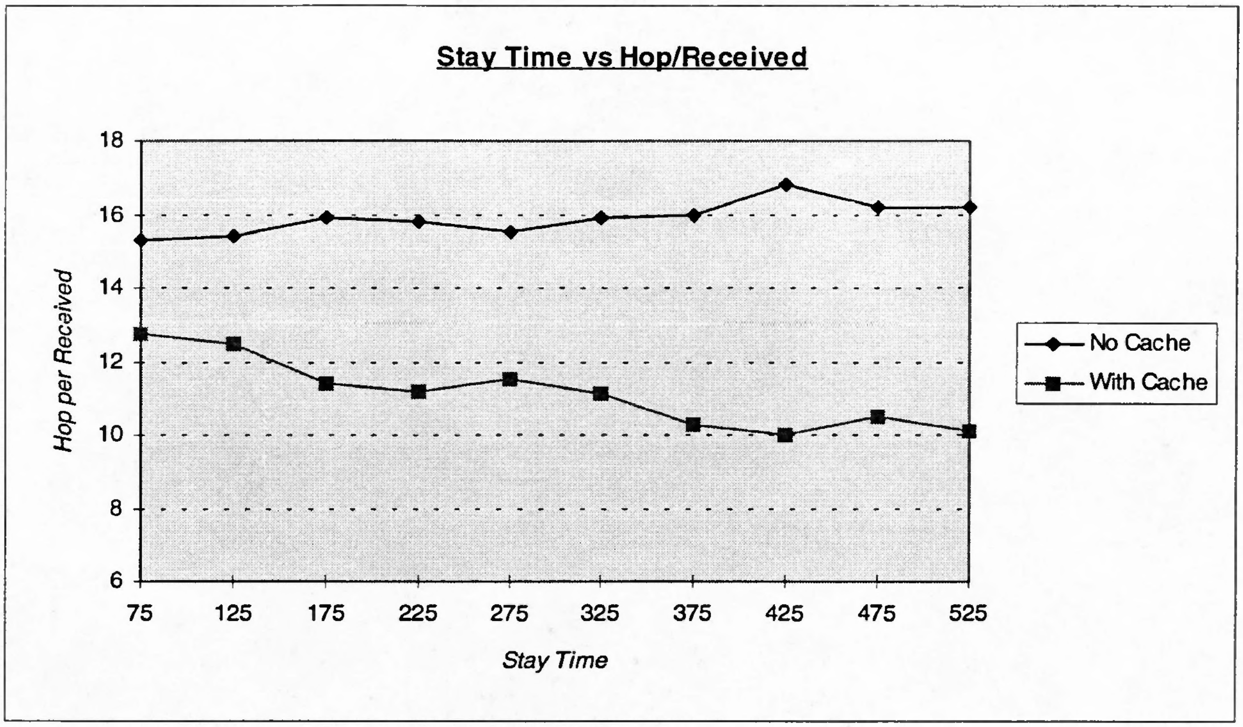


Figure 20 : Stay Time vs Hops/Received

Parameters Used : Run Time = 6000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops, Maximum Links per Router = 4, Maximum Cells Moved=10

- In Route Optimization with Cache, Hops per Packet Received, goes down a little when the Stay Time goes up. This is because more and more Packets that are directly tunneled from the source, are able to find the destination without having to go through the Foreign Agent Handoff.

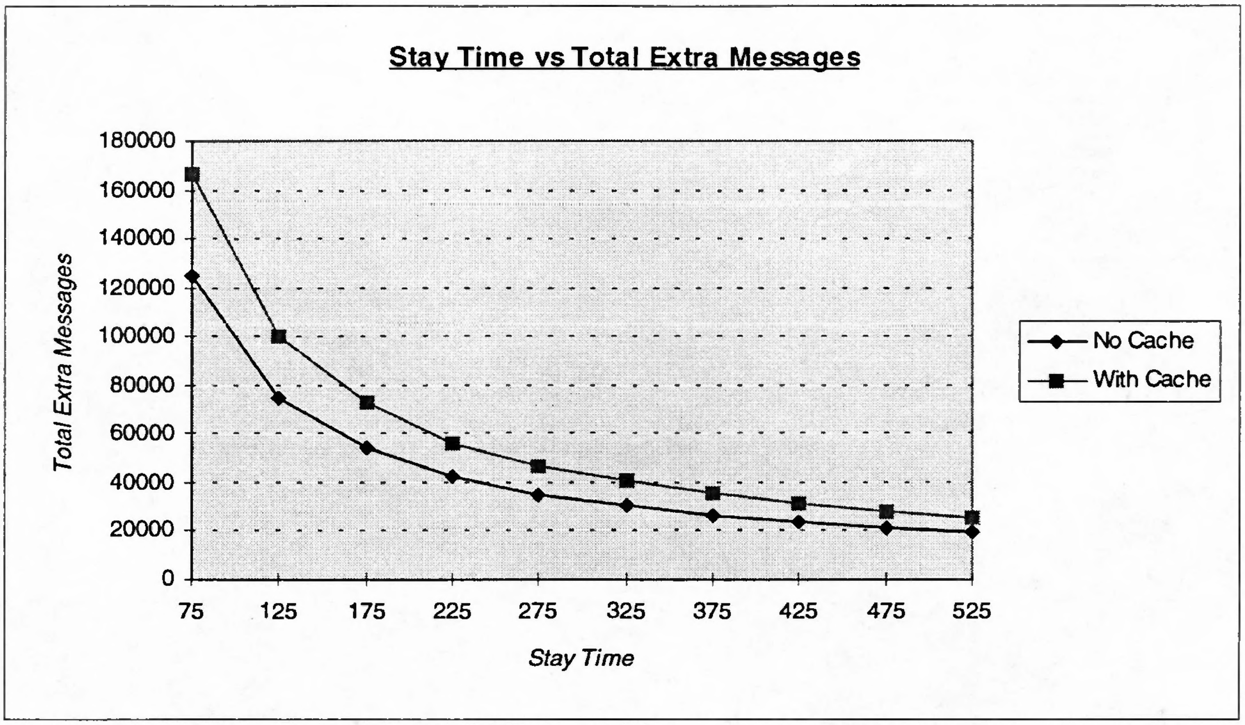


Figure 21 : Stay Time vs Total Extra Messages

Parameters Used : Run Time = 6000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops, Maximum Links per Router = 4

- Extra messages are made of Reg Requests, Reg Response, Solicit Advrt, Req to Extend Stay, Binding Warning , Binding Update.
- Extra messages are 1 packet each.
- Very large number of additional packets required to support Mobile IP.

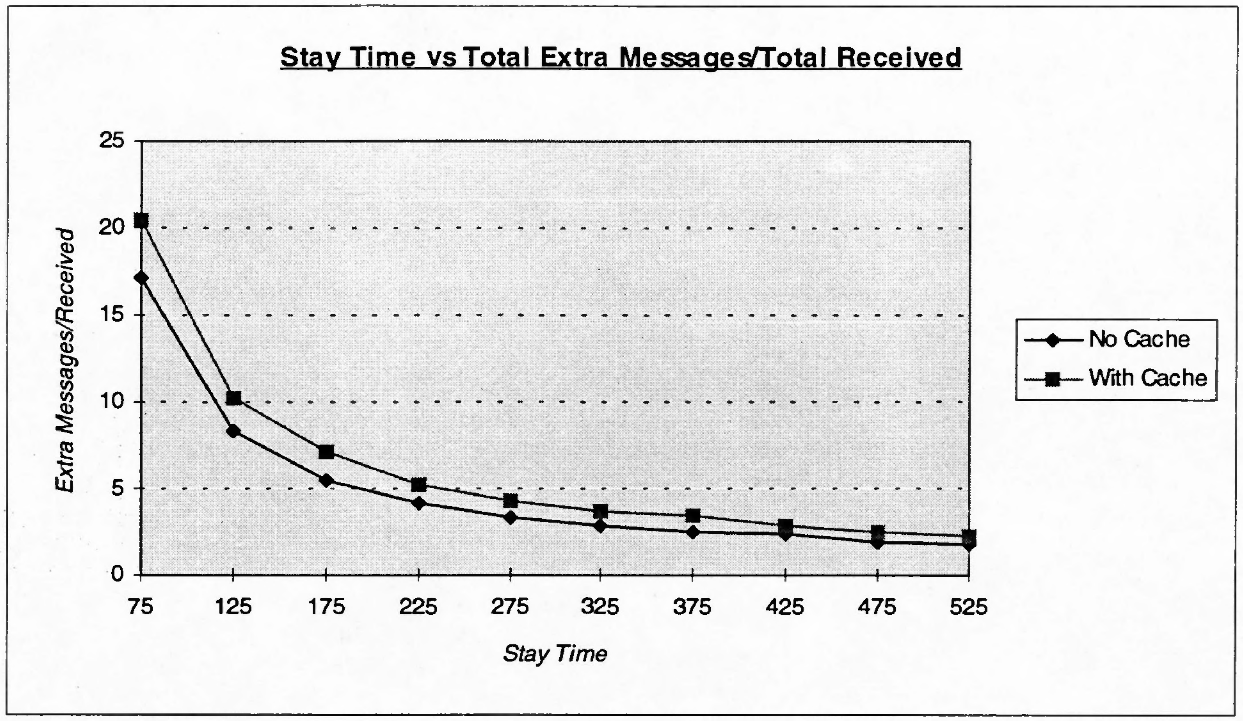


Figure 22 : Stay Time vs Total Extra Messages/Total Received

- When the Stay Time is small, extra messages required to support Mobile IP could be as much as 10-20 times the number of Data Packets delivered.

5.6 Experiment with Distance Between Sender and Receiver

Figure 23 shows the impact of the distance between the sender and receiver on the hops per received packet. If the sender and receiver are far apart, and the mobile node does not travel far away from home, Route Optimization with cache is not very effective.

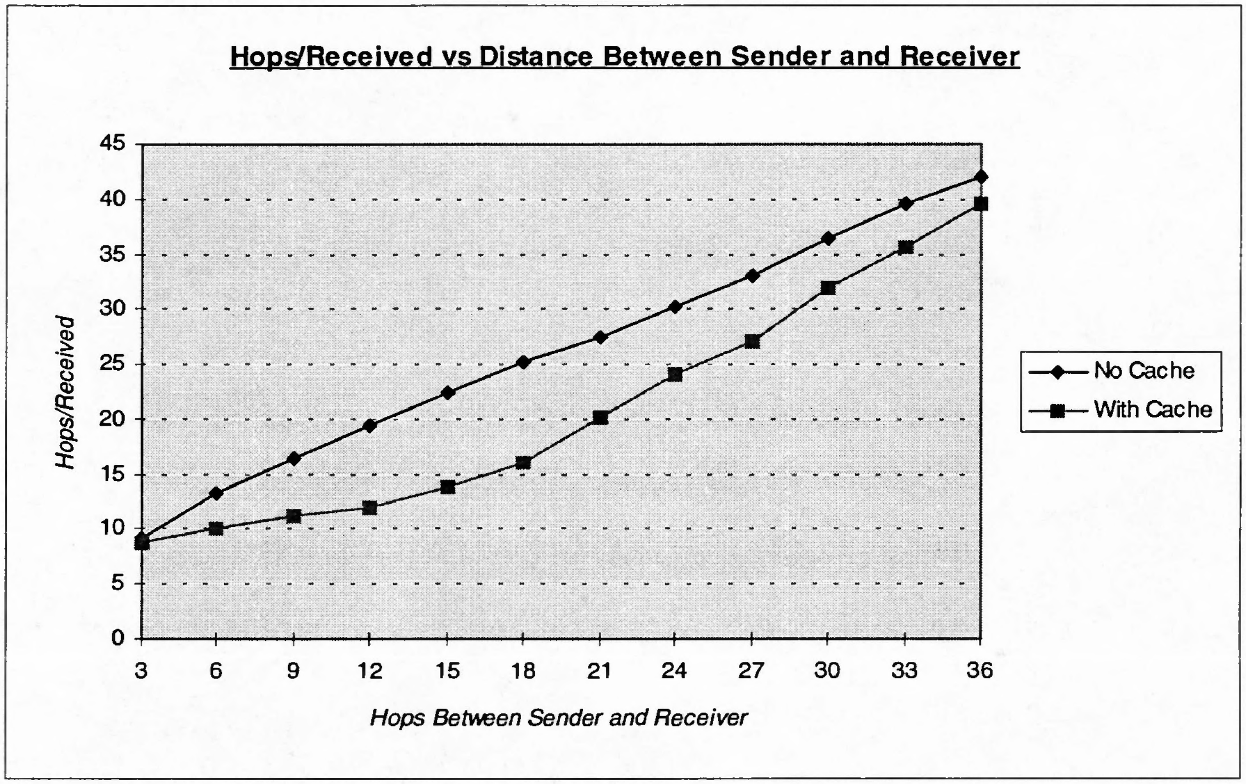


Figure 23 : Hops/Received vs Distance between Sender and Receiver

Parameters Used : Run Time = 6000 units, 20 x 20 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Maximum Links per Router = 4, Max Cells Moved= 10

- As the distance between the Sender and Receiver goes up, the relative effectiveness of cache, in terms of hop count per received, goes down.

5.7 Experiment with Density of Interconnection

Figures 24 and 25 show the effect of the density of interconnection between routers, on the performance of Mobile IP. As can be seen, having more than four connections per router does not improve the performance a great deal.

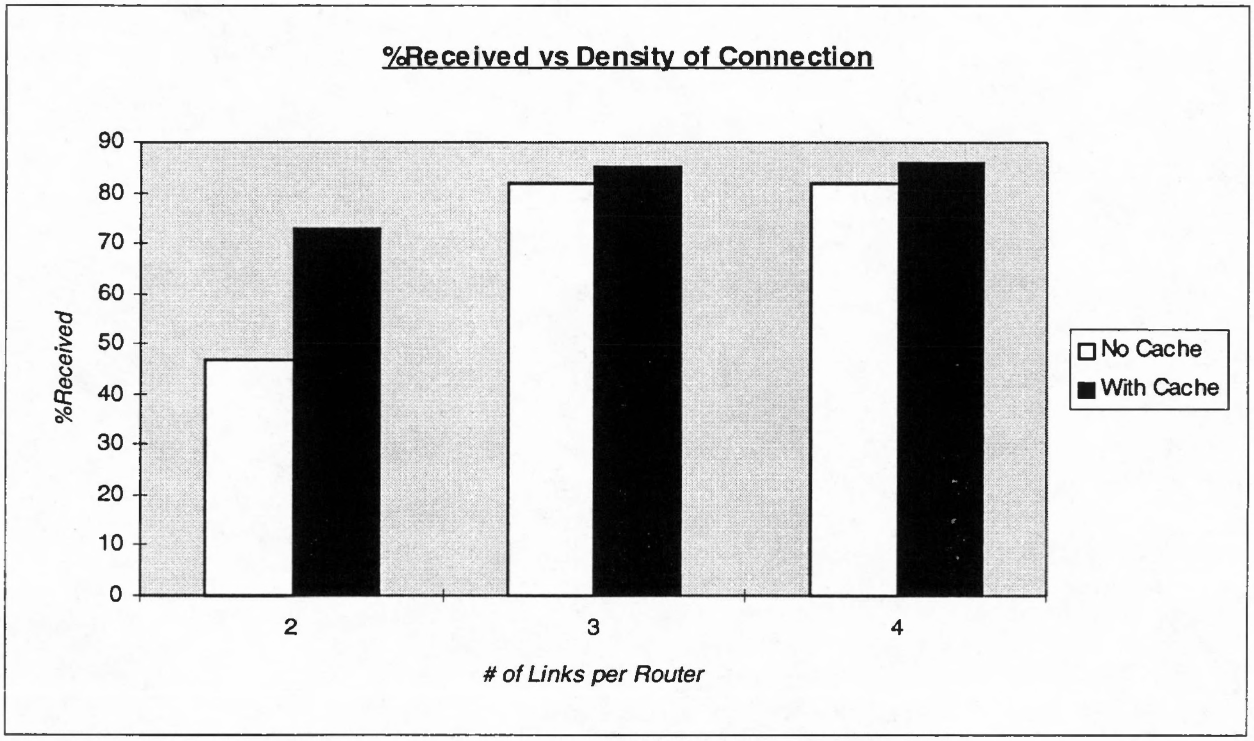


Figure 24 : %Received vs Density of Connection

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Max Cells Moved= 10

- %Received goes up as the number of links per router goes up, however beyond 4 links per router, it does not have a significant effect.

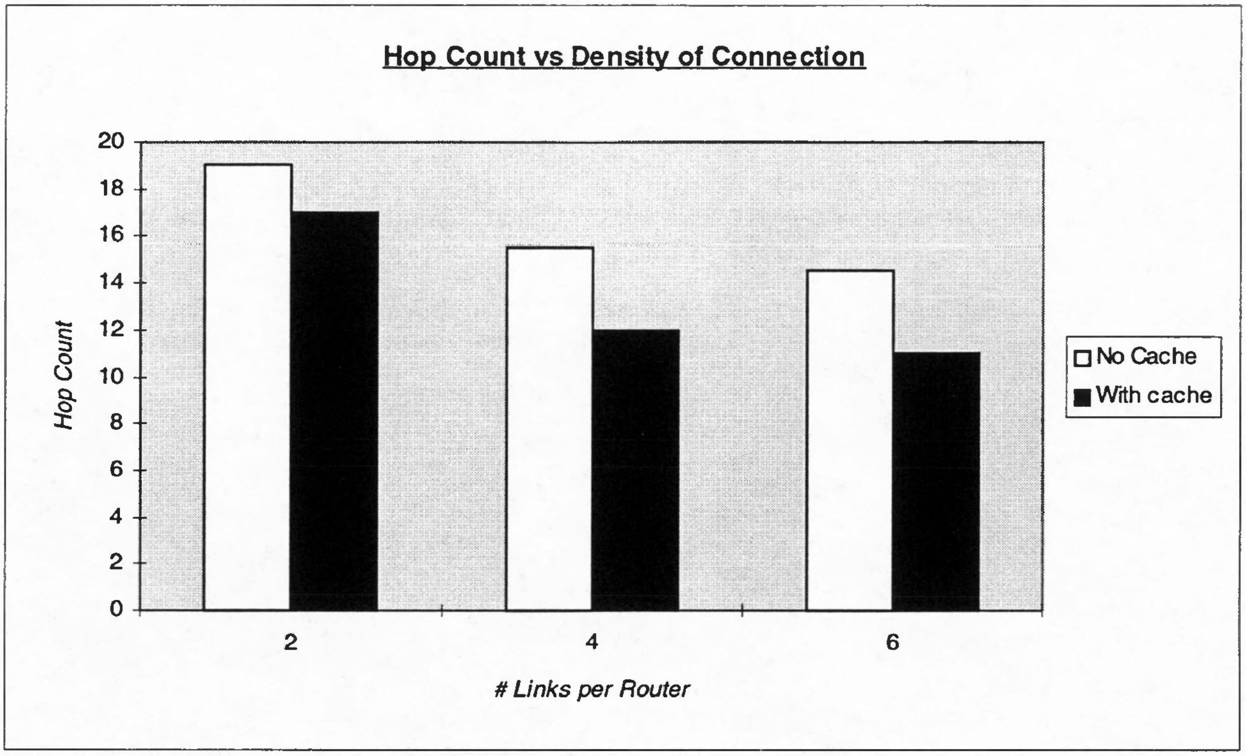


Figure 25 : Hop Count vs Density of Connection

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Max Cells Moved= 10

- Similar to %Received, the Hops per Received goes down as the number of links per Router goes up.

5.8 Experiment with Percentage of End Nodes Mobile

Figure 26 shows the impact of the number of end nodes that are mobile, on the performance of Mobile IP. This test was conducted by letting the sending node also to be mobile. As can be seen, the effect is not very significant, even when all nodes are mobile, the percentage of received packets does not drop by a lot.

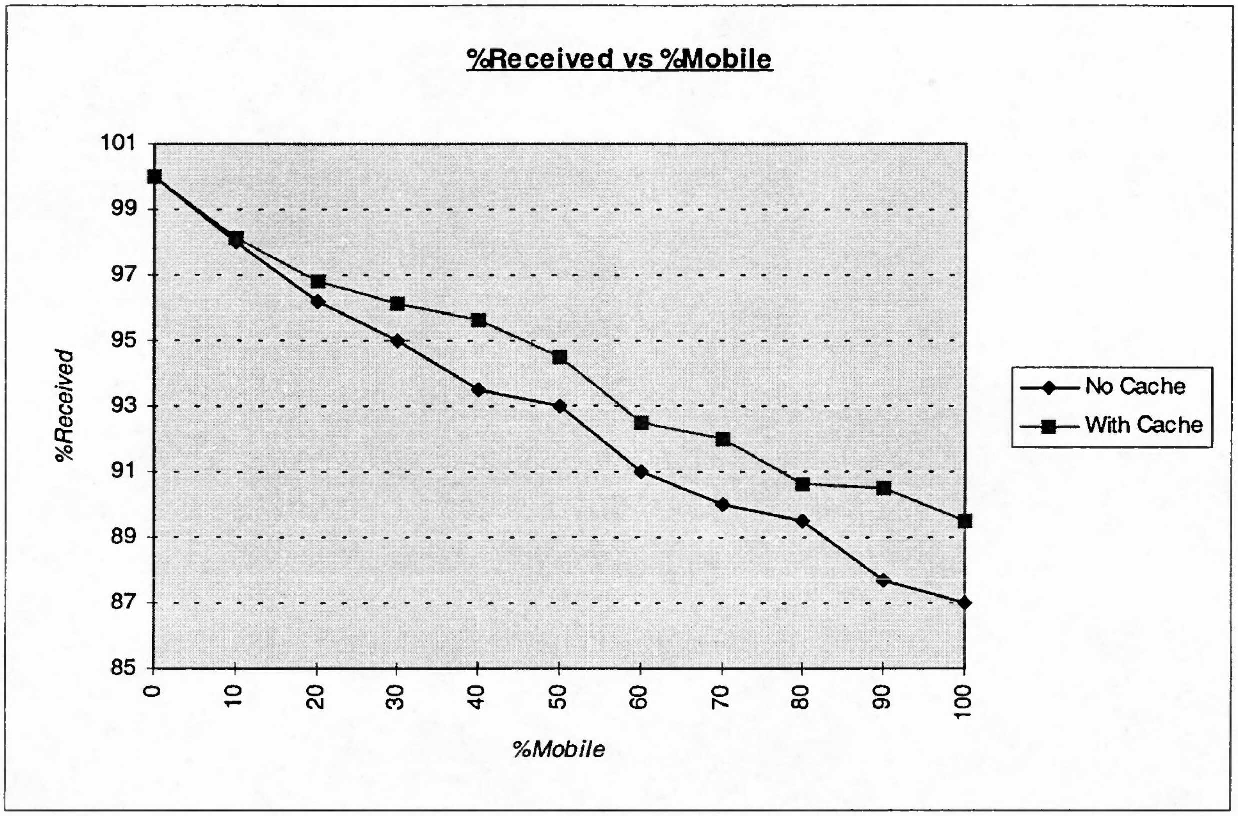


Figure 26 : %Received vs %Mobile

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops,, Maximum Links per Router = 4, Average Wired Link Speed= 3 units, Sender and Receiver could both be mobile

- This graph shows data where the sender and the receiver could both be mobile.
- As more and more End Nodes become mobile, the percent received goes down.
- Route optimization with cache still performs better.

5.9 Experiment with Speed of Wireless Link

Figures 27 and 28 show the impact of having a relatively slow Wireless link on the performance of Mobile IP. As can be seen, if the Wireless link is slow (50 times slower than a wired link, which is possible in some cases), the percentage of packets received goes down significantly. However, if the stay time is high, then the drop in percentage received is not as much. The drop therefore, can be attributed to the delay in making registrations.

When the speed of the Wireless link drops to 1/100 of the speed of wired link, as shown in Figure 28, losses are very high, when the stay time is also low.

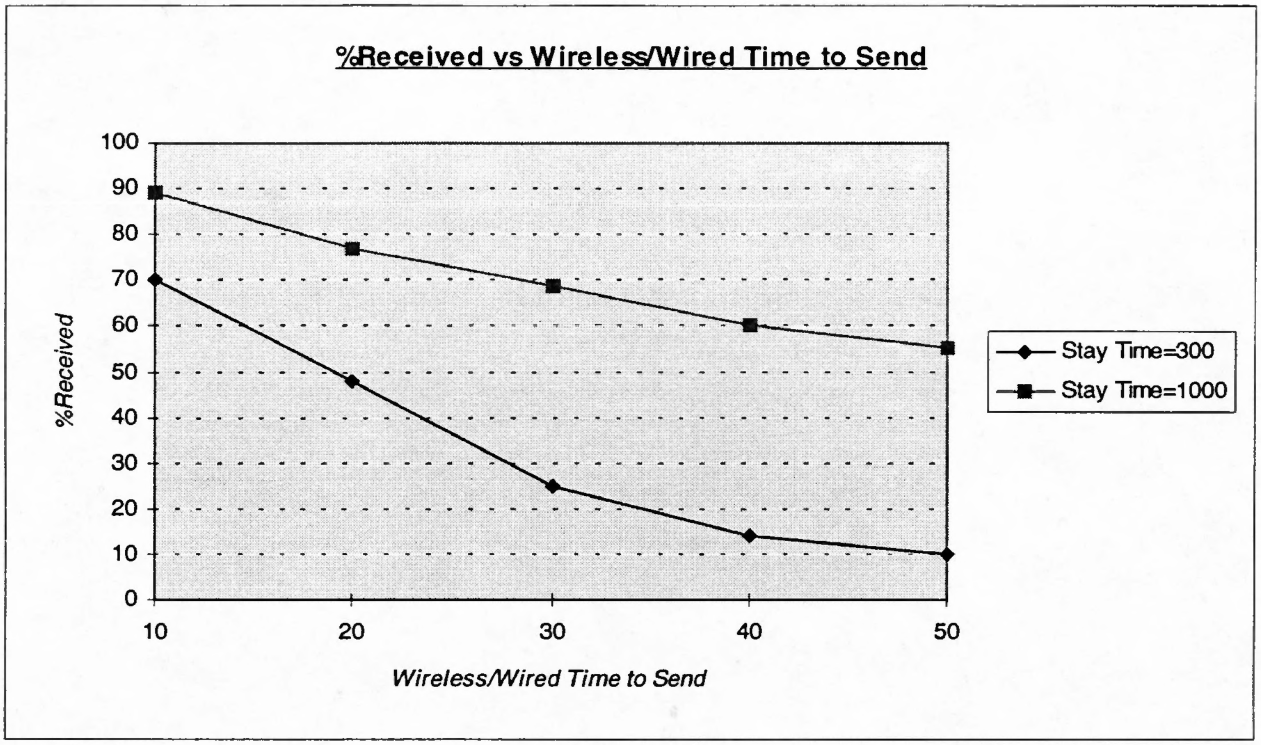


Figure 27 : %Received vs Wireless/Wired Time to Send

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops, Maximum Links per Router = 4, Average Wired Link Time= 3 units

- %Received goes down as the speed of the Wireless link goes down, however the drop is not much when the stay time is high.

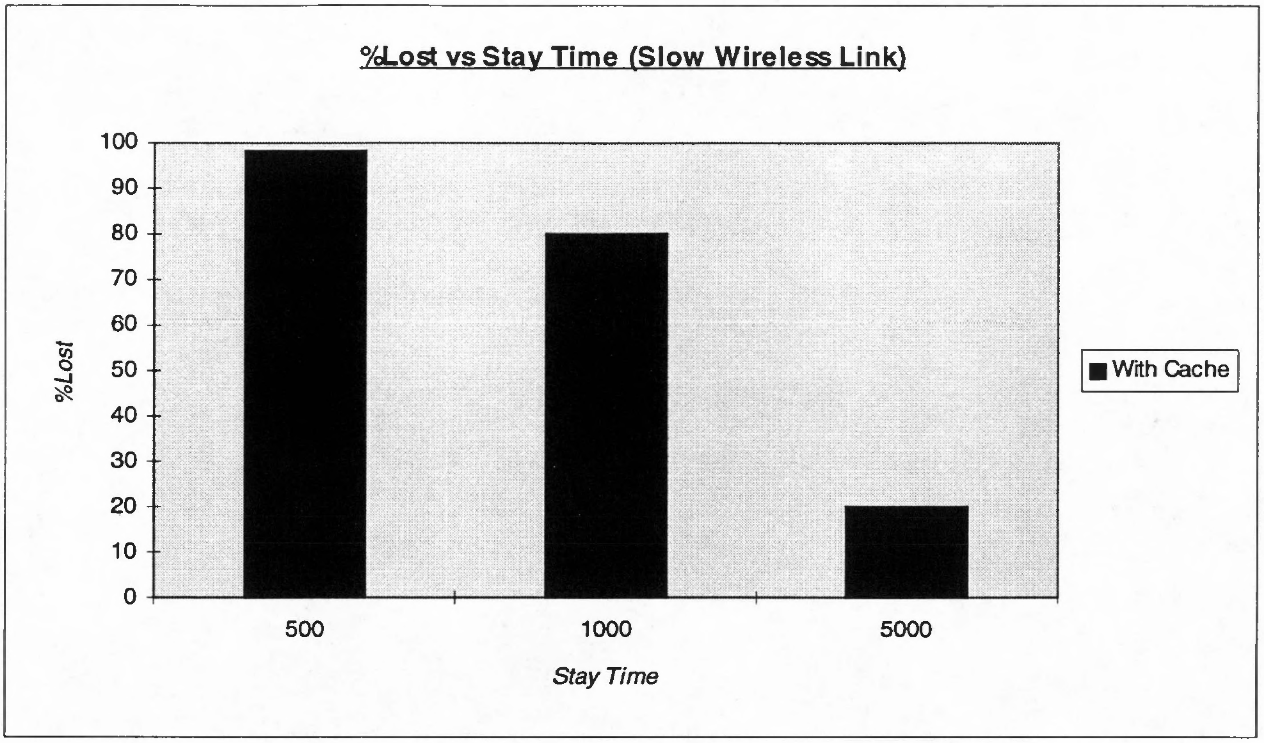


Figure 28 : %Lost vs Stay Time

Run Time = 50000 units, 10 x 10 Grid Network, Cache Size = 10, Average Distance Between Sender and Receiver = 10 hops, Maximum Links per Router = 4, Wireless/Wired Link Time= 100

- The speed of the Wireless Link has a large impact on the performance of the network.
- CDPD has a speed of about 10-100Kbps, whereas a wired link speed can be of the order of upto 10 Mbps. Therefore, Wireless/Wired speed of the order of 100 would not be unusual.
- For it to be efficient, Stay Time has to be high when the Wireless link is slow.

5.10 Experiment with Time to Lookup Cache

Figure 29 shows the effect of time spent in looking up the cache, on the performance of the network. Cache look up time has been exaggerated here to clearly show the impact. This was necessary because tests were conducted with relatively small size of cache. In reality, cache sizes are likely to be much more than what was used in this test.

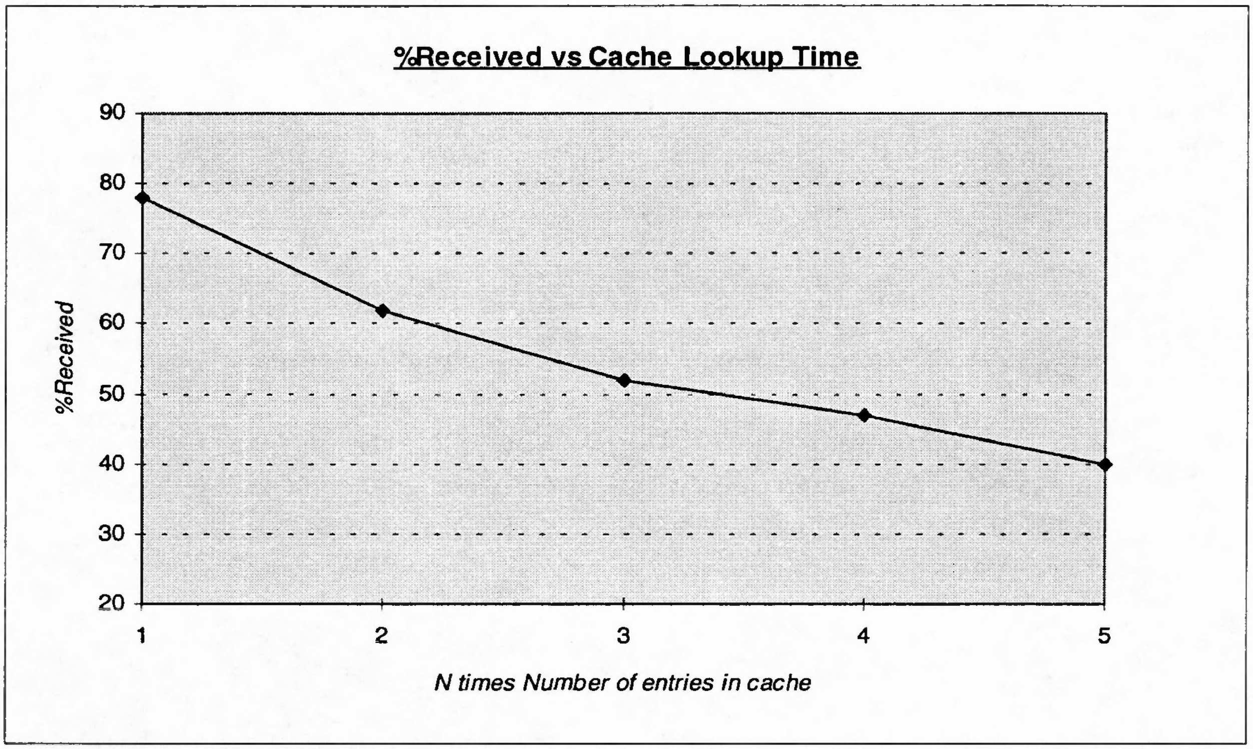


Figure 29 : %Received vs Cache Lookup Time

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Max Cells Moved= 10, 50% of End Nodes Mobile

- %Received goes down as the time spent in looking up the cache goes up
- Lookup time has been exaggerated here, however, in real life, cache sizes are likely to be much higher and the traffic on some routers could be much higher than in this

test. Therefore, even the exaggerated lookup time should give a good picture of what could happen in reality.

5.11 Experiment with Errors in the Wireless Link

Figure 30 shows the effect of error in the Wireless Link, on the performance of the network. As can be seen, packet loss due to errors in the wireless link can severely impact the performance of the network. This is more so because mobile nodes take longer to register when the REG_REQUEST and REG_RESPONSE messages are lost.

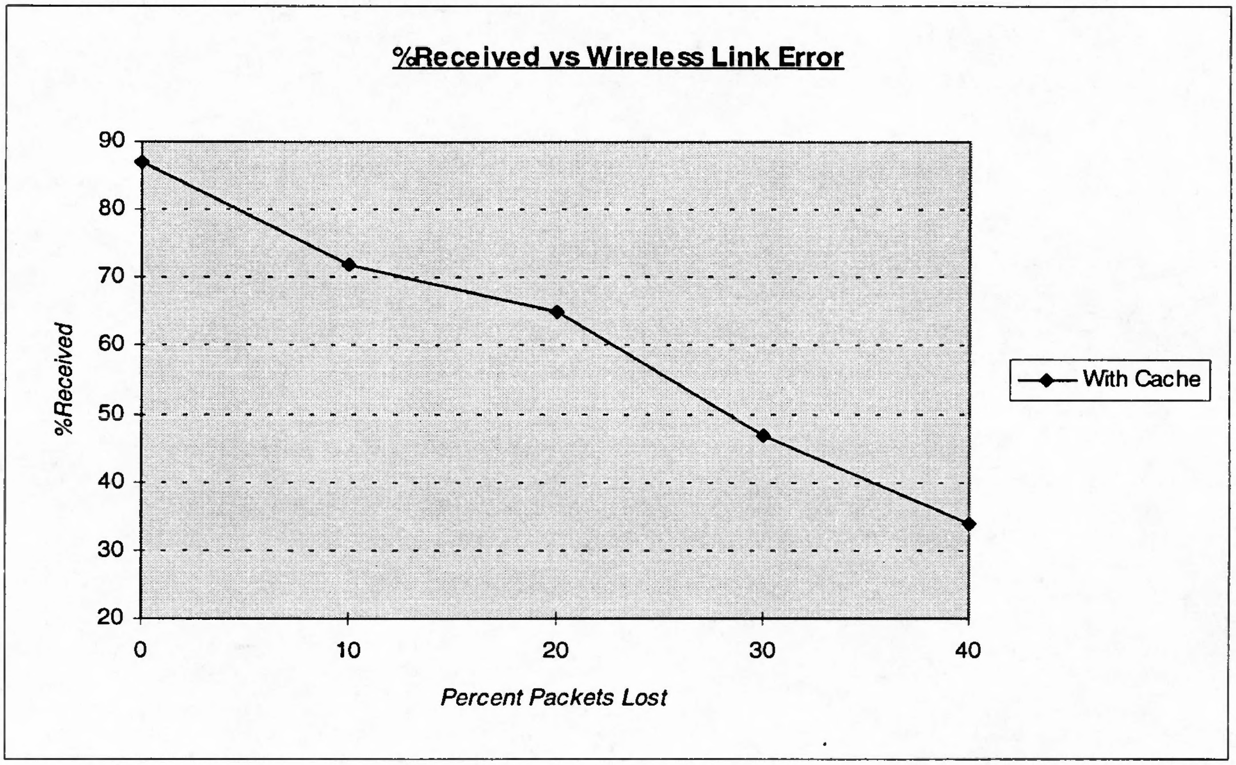


Figure 30 : %Received vs Wireless Link Error

Parameters Used : Run Time = 4000 units, 10 x 10 Grid Network, Average Stay Time = 300 units, Cache Size = 10, Max Cells Moved= 10, 50% of End Nodes Mobile

5.12 Summary

In this chapter we have presented results from the different experiments that were conducted during our simulation study. It was found that parameters like the time a mobile node stays in each cell, errors in the wireless link and the distance that mobile nodes move away from home, have a significant impact on the performance of Mobile IP. Performance was measured in terms of the percentage of packets delivered and the average number of hops taken by each packet to reach the destination. These experiments also showed that Mobile IP with Route Optimization performs better when the mobile node moves far away from home and when the mobile node spends some time at each foreign network that it visits.

CHAPTER 6 : CONCLUSION AND FUTURE WORK

Since it is difficult, in reality on a large network, to study the performance of Mobile IP, because the infrastructure does not exist widely, software simulation of Wide Area Networks offers a good alternative. As had been pointed out earlier, not many attempts have been made in the past to simulate Mobile IP on large networks. This thesis has attempted to model a section of the internet. It is assumed that various parameters like the number of routers, the number of end nodes (hosts) and the density of interconnection, which have been left open, would allow the construction of a model that resembles a real network. Attempt has also been made to make the modeled network look less uniform. In the real world, links between nodes in a network, are not all alike, therefore in this model, the transmission speed of links can be varied randomly around an average value. Similarly, it is also possible to introduce random errors in transmission over wireless links.

Results from the various experiments shows that parameters like the time a mobile node stays in each cell, the performance of the wireless link and the distance that mobile nodes move away from home, have a significant impact on the performance of Mobile IP. Under ideal conditions, the percent of packets that are successfully delivered, when the stay time goes up from 100 to 500 units, is about 25% higher. Similarly, as the speed of the wireless link, compared to a wired link goes down, the percentage of packets delivered also goes down. By far, the parameter that influences the performance most, is errors in the wireless link. As the rate of error goes up to 50%, i.e. 1 out of 2 packets is lost because of error in the link, the percentage of packets that are successfully received, drops down to about 30%, from 95% when there are no errors.

Overall, it was found that Mobile IP with Route Optimization performs better than Mobile IP without any optimization. When the distance moved from home, in terms of number of cells, is small, there is no significant gain. However, as the mobile node moves farther away from home, there is significant saving in the number of hops taken by packets to reach their destination. Similarly, performance in terms of the number of packets delivered, is about 20% better, with Route Optimization, than that without it, irrespective of the amount of time that a mobile node spends at each cell. As the distance between the sender and receiver goes up, the relative advantage of having Cache goes down. Most of the experiments were conducted with the assumption that the sending end nodes are stationary, however, when the sender and the receiver could both be mobile, it was found that the percentage of packets delivered goes down as the percentage of end nodes that are mobile goes up

The simulator was built on the assumption that error handling and sequencing of packets would be handled by the higher layers. Therefore, no provision was made for acknowledgments, re-transmissions etc. While these are not handled by IP, they do place a significant overhead on the network. In a future version of this simulator, to make it model the real traffic, there should be provision for sequencing and error handling by the upper layers. Routing of packets, in this simulator, is done using static routing tables. In real life networks, static routing is rarely used. The only kind of control messages (ICMP) used in this simulator are the ones required to support Mobile IP. In a future version, there should be provision for other kinds of control messages between the routers, to enable dynamically changing the routing tables based on congestion and failure of routers. This thesis has not dealt with the important area of authentication. A future version of the simulator should provide for intruder detection and authentication, since these are part of the Mobile IP standard.

While simulation offers an excellent tool to study the behavior of a wide area network, under Mobile IP, it can not be a replacement for studying the system using a real network. As has been done at some other Universities, it is a good idea to create a test bed for validating the simulation results, as well as to fine tune the input parameters that the simulation uses. This, however, has to be a much bigger effort because of all the hardware infrastructure and software components that have to be built.

REFERENCES

1. Charles E. Perkins, *Mobile IP - Design Principles and Practices*, Addison Wesley, 1998.
2. Ben Lancki, Abhijit Dixit and Vipul Gupta, *Mobile-IP: Supporting Transparent Host Migration on the Internet*, http://anchor.cs,binghampton.edu.
3. Andrew S. Tanenbaum, *Computer Networks*, Prentice Hall PTR, 1996.
4. James D.Solomon, *Mobile IP - The Interne Unplugged*, Prentice Hall PTR, 1998.
5. Abhijit Dixit and Vipul Gupta, *Mobile IP for Linux,* http://anchor.cs,binghampton.edu.
6. Kevin Lai, Mema Roussopoulos, Diane Tang, Xinhua Zhao, Mary Baker, *Experiences with a Mobile Testbed,* http://mosquitonet.stanford.edu/mosquitonet.html.
7. Xinhua Zhao, Mary Baker, *Flexible Connectivity Management for Mobile Hosts,* http://mosquitonet.stanford.edu/mosquitonet.html.
8. Stuart Cheshire and Mary Baker, *Internet Mobility 4X4*, http://mosquitonet.stanford.edu/mosquitonet.html.
9. Fredrick Broman and Fredrick Tarberg, *Implementation and Analysis of the Mobile-IP Protocol,* Department of TeleInformatics, Royal Institute of Technology, Stockholm, Sweden.
10. David B. Johnson and David A. Maltz, Protocols for Adaptive Wireless and Mobile Networking, http://monarch.cs.cmu.edu.

11. Alex Hills and David B. Johnson, A Wireless Data Network Infrastructure at Carnegie Mellon University, http://monarch.cs.cmu.edu.

12. Noble B., Nguyen G. and Satyanarayanan M., Katz R., *Mobile Network Tracing*, Internet RFC 2041, Oct. 1996.

13. Narsingh Deo, Graph Theory with Applications, Prentice Hall, Inc., 1974.

14. David B. Johnson and Charles E. Perkins, *Route Optimization in Mobile IP*, draft-ietf-mobileip-optim-07.txt-work in progress, Nov. 1997.

15. Charles E. Perkins, *IP Mobility Support,* draft-ietf-mobileip-protocol-16.txt - work in progress, Apr. 1996.

16. Tomasz Imielinski & Henry F. Korth, *Mobile Computing*, Kluwer Academic Publishers, 1996.

17. Victor S. Frost & Benjamin Melamed, *Traffic Modeling for Telecommunications Networks*, IEEE Communications Magazine, March 1994.

18. Kevin Watkins, *Discrete Event Simulation in C*, McGraw Hill, 1993.

19. M. Weiser, *Some Computer Science Issues in Ubiquitous Computing*, Communications of the ACM, Volume 36, Number 7, Pages 74-83, July 1993.

20. W. Tuttlebee, *Cordless Personal Communications*, IEEE Communications Magazine, Volume 30, Number 12, Pages 42-54, December 1992.

21. Raymond Steele, *The Cellular Environment of Lightweight Handheld Portables*, IEEE Communications Magazine, Volume 27, Number 7, Pages 20-29, July 1989.

22. M. Spreitzer and M. Theimer, *Scalable, Secure, Mobile Computing with Location Information*, Communications of the ACM, Volume 36, Number 7, 1993.

23. M. Satyanarayanan, *Mobile Computing*, IEEE Computer, Volume 26, Number 9, 1993, Pages 81—82.

24. Subhashini Rajgopalan and B. R. Badrinath, *Adaptive Location Management for Mobile-IP*, First ACM International Conference on Mobile Computing and Networking - Mobicom'95, November 1995.

25. R. Molva and D. Samfat and G. Tsudik, *Authentication of Mobile Users*, IEEE Network, Volume 8, Number 2, 1994, Pages 26-34.

26. Pravin Bhagwat and Charles Perkins: *A Mobile Networking System based on Internet Protocol*, IEEE Personal Communication Magazine, February 1994.

27. D. E. Comer and J. C. Lin and V. F. Russo: *An Architecture For A Campus-Scale Wireless Mobile Internet*, ftp://gwen.cs.purdue.edu/pub/lin/xpoint.ps.

28. Georges Forman and John Zahorjan: *The Challenges of Mobile Computing*, IEEE Computer, April 1994, Pages 39-47.

29. Randy H. Katz, ed., *Bay Area Research Wireless Access Network (BARWAN)*, http://http.cs.berkeley.edu/~randy/Daedalus/BARWAN/BARWAN_over.html.

30. Fumio Teraoka, Keisuke Uehara, Hideki Sunahara and Jun Murai, *VIP: A Protocol Providing Host Mobility*, Communications of the ACM, Aug. 1994, Pages 59-75.