

Implementation of Mobile Agent in African Countries

Yassin Abdulkarim Hamdalla¹, Osama Ali Abdelgadir², Makkawi Abdelsalam Mohamedkheir³,
Mohammed Altayeb Ahmed⁴

^{1,2,4}Future University, Department of Electronics & Communication Engineering,
Faculty Telecommunications & Space Technology, Africa Avenue, Khartoum, Sudan

³Future University, Department of Computer Engineering, Faculty of Engineering, Africa Avenue, Khartoum, Sudan

Abstract: *Mobile agents are agents that can physically travel across a network, and perform tasks on machines that provide agent hosting capability. This allows processes to migrate from computer to computer, for processes to split into multiple instances that execute on different machines, and to return to their point of origin. In this project, we propose to implement a mobile agent in networks based on load balance to reducing force in network traffic and overcoming network latency and their ability to operate asynchronously and independently of the process that created them, mobile agent has been an effective choice for parallel and distributed computing, by using a dynamic load-balancing technique. The method used in this research project is MATLAB Simulation method to design and describe the dynamic load balancing for building the network reservation using mobile agent. The results of our mobile agent simulator with the load balance is addresses un-regulated jobs/tasks migration among the servers. Considering distributed several servers running, a lot of bandwidth is wasted due to unnecessary job migration. This work attempts to regulate this movement to minimize bandwidth consumption and network traffic. According to the results and compression, the mobile agent network based on load balance is better than SNMP and give a high success in simulation environment test, simulator has been developed to enhance the SNMP problem. The simulation show the validity of the algorithm to improve the variance of the overhead load in the network and in providing better average response time.*

Keywords: MA, Load Balancing, SNMP, Mobility, Matlab.

1. Introduction

Mobile agents are currently a hot topic in the domain of distributed systems. Reasons are the problems more traditionally designed distributed systems, especially client/server systems, might have to handle work-load, the trend to open large numbers of customers' direct access to services and goods, and user mobility. Mobile agent technology can help to design innovative solutions in this domain by complementing other approaches, simply by adding mobility of code; machine based intelligence, and improved network- and data-management possibilities [1]. Mobile agent can be a new design paradigm in the area of distributed programming and a useful supplement of traditional techniques like the client/server architecture. There are many notions of the term of mobile agent then to understand the meaning; it is any kind of software entity that is able to initiate a migration on its own within a network of heterogeneous computer systems.

In the last year the research and development of mobile agents are very fast growing research area and development. Most projects conducted so far had the goal to proof the possibility to implement Mobile code which is the basis of the mobile agent, in the different programming languages, and show that it is worth using mobile agents for particular application domains [1].

1.1 Understanding of Mobile Agent (MA)

First we talk about the technology of mobile agent. Technology of mobile agent is very new part of computer science research. A mobile agent is not bound to the system

where it begins execution. It has the unique ability to transport them from one system in a network to another. The ability to travel allows a mobile agent to move to a system that contains an object with which the agent wants to interact and then to take advantage of being in the same host or network as the object [2].

A mobile agent is a program that can migrate from host to host in a network of heterogeneous computer systems and fulfill a task specified by its owner. It works autonomously and communicates with other agents and host systems. During the self-initiated migration, the agent carries all its code and the complete execution state with it. Mobile agent systems build the environment in which mobile agents can exist [1]. Although, Mobile agent can be considered as an incremental evolution of the earlier idea of the "process migration". Mobile agents are autonomous programs that can move from host to host in a network, interacting with resources and other agents [3]. Also mobile agent is a running program that can suspend its own execution (the state of the running program is saved) transfer itself to another agent-enabled host in a network and resume its execution at that host. From that host the mobile agent may then decide to transport itself to another in order to carry out more some processing or it may return to its initial host. Mobile agent is a specialization of agent; with the only difference being that it is mobile because the process migration systems in that the agent moves when they chose. [4]. Finally you can define Mobile agents are, intelligent programs, possess the ability to make decision, migrate through a network to acquire services, it is capable of autonomous and asynchronous action, also MA reduces the network load, facilitate robust and fault tolerant applications,

and the platform independent.

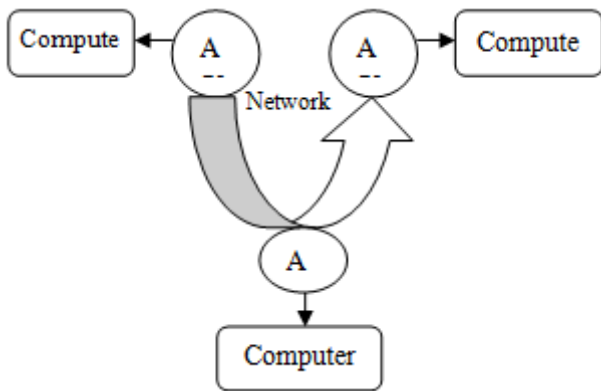


Figure 1: Mobile Agent Travels Computer to Computer

1.2 Mobile Agent and Load balancing Overview

A mobile agent is an executing program that can migrate, at times of its choosing, from machine to machine in a heterogeneous network. On each machine, the agent interacts with stationary service agents and other resources to accomplish its task. Developments in wireless technology liberate network nodes from the constraint of being placed at a fixed physical location and enable the advent of the so-called mobile computing. The proliferation of mobile computing devices, which have the characteristics of low bandwidth and unreliable network connection, has led to the increased use of mobile agent since it supports disconnected operations. Mobile agent paradigm provides a better conservation of bandwidth since only the final result returns back to the client. When the server lacks one of the services, the mobile agent migrates to the server and performs the set of required operations locally. This, in turn, leads to reduction in total completion time. A mobile agent provides an effective means for overcoming network latency; it monitors the network latencies and continually moves to the network location that minimizes the average latency between itself and its clients. [5]

1.3 Load Balancing and Mobile Agents

The mobile agent technology can provide various answers to solve the problems of distributing the load in a set of computational entities. Migration of processes was traditionally a solution to this problem, generally under the supervision of a centralized controller. In a modern perspective, multi agents systems can decentralize the distribution of the computational load. In fact, a complex application is divided into autonomous parts, each of which delegated to one or more mobiles agents. Each mobile agent is in charge of searching for the node of the network which offers the most convenient resources, where to execute its own part of code. During the execution, agents can move to other nodes where more computational resources are available, in order to better distribute the load. [5]

2. Methodology

The method used in this research project is MATLAB Simulation method to design and describe the dynamic load

balancing for building the network reservation using mobile agent. This is because the software simulation is flexible, carries out the analysis and the design of the applications in a simple way. All of these features allow the progress of the work to be smooth and easy to apply.

2.1 The Physical Model

Network and resource information: in a simulated environment physical resources play a major role as well. In this case the characteristics that are of interest include: Consist of 10 node, 1 router, 2 Switches, 1 Mobile Agent Server, and 3 sub servers.

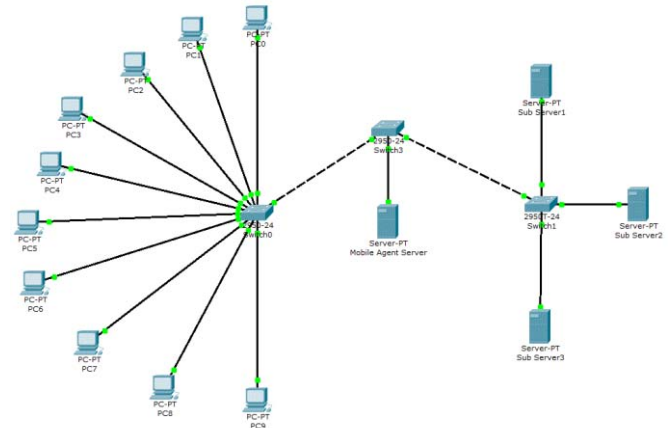


Figure 3.2: Network topology and resource

2.2 Simulation Assumption

The proposed software is MATLAB simulation that reduces the time overhead and traffic we utilized assumption as the following:

- The agent system in mobile agent server used to create agents, and all agents have home server.
- The function randomly selects an agent. The function also provides the ability to exclude certain agents from the random draw. It can also restrict the random draw to agents who have skill of a task.
- The Network Load

We start by analyzing the RMI approach. The client sends a request to invoke a method placed on the server. This method searches locally for the data and replies to the invoker by the found data result or returns a NOT-FOUND reply. The client repeats the search process in the next server until the data item is found. Hence, the network load affected by this approach (BCS) is calculated by: [6]

$$B_{cs} = \sum_{i=1}^n p_i \sum_{j=1}^{i-1} (1-p_j) (iB_{req} + (i-1)B_{NF} + B_{res}) \quad (1)$$

For the ease of modeling, we assume that all servers have the same probability (p) of finding data item. Then, equation 1 can be rewritten as:

$$B_{cs} = np \sum_{i=1}^n (1-p)^{i-1} (iB_{req} + (i-1)B_{NF} + B_{res}) \quad (2)$$

Now, we investigate the mobile agent approach. For sake of simplicity, we additionally assume that when the agent migrates to a new location it carries all its code, data, and all

state information by using the “push all-to-next” migration strategy. However, while the agent migrates back to home, it carries only the data state. BMA is the total network traffic, in bytes, caused by this paradigm. This network load is calculated by:

$$BMA = \sum_{i=1}^n p_i \prod_{j=1}^{i-1} (1-p_j) (i B C + (i+1) B S + i B req + (1-\sigma) B res) \quad (3)$$

Again, if we assume that all the servers have the same probability (p) of finding data item. Then equation 3 can be written as:

$$BMA = np \sum_{i=1}^n (1-p)^{i-1} (i B C + (i+1) B S + i B req + (1-\sigma) B res) \quad (4)$$

Considering the case of handheld devices or as an example a PDA, we are concerned only with the traffic cost from the client side such as GPRS connection to the internet. The traffic by the mobile agent paradigm is calculated by:

$$BMA_{GPRS} = B C + 2 B S + B req + (1-\sigma) B res \quad (5)$$

This load remains constant whatever the number of the visited locations. In addition, the impact of client wireless connection reliability is considered negligible in the case of using mobile agent approach. Therefore, the use of the mobile agent approach reduces the network traffic by the client side to a minimum. Whereas, the network traffic that is caused by the RMI approach remains unchanged from that one calculated above by equation 2.

$$tcs = \sum_{i=1}^n p_i \prod_{j=1}^{i-1} (1-p_j) (t B_0 + (B req + B res) \sum_{j=1}^{i-1} (1-p_j) + t B_0 i) \quad (6)$$

Applying the same approach in simplifying equations 3, 4, and suppose that the network have the same delay per byte over all the links. Then equation 6 can be simplified to be as shown next.

$$tcs = np \sum_{i=1}^n (1-p)^{i-1} (t B (B res + i B req + (i-1) B NF) + i t p) \quad (7)$$

The corresponding response time in the mobile agent approach is calculated by:

$$tMA = \sum_{i=1}^n p_i \prod_{j=1}^{i-1} (1-p_j) (B ma (n t m + \sum_{j=1}^{i-1} (t B (j-1) j) + (B S + (1-\sigma) B res) (t B_0 i + t m) + i t S + i t p)) \quad (8)$$

Simplifying this equation, using the above simplification assumptions leads us to:

$$tMA = np \sum_{i=1}^n (1-p)^{i-1} ((i B C + i B req + (i+1) B S + (1-\sigma) B res) (t B + t m) + i t S + i t p) \quad (9)$$

Using the calculated network load from equation 3 simplifies the response time equation 9 as:

$$M = \frac{np(BMA(tBn(n+1) A + tm) + (tS+tp))}{2} \quad (10)$$

2.3 Constant Parameters

Depend on shape or architecture of network:

- a) Packet size
- b) Compression efficiency
- c) No of server

2.4 Inputs Parameter

Used variable input parameters to check the efficiency of the system

- a) No of agent in network.
- n - Number of agents
- b) No of task ((Task = the task that an agent should be skilled or aware to be included in the draw), Agents_tasks = n x m matrix that contains either a zero or a one to indicate if an agent can solve a task.
- m - Number of tasks.
- c) rev_agent = the previous agent working on the task, the values in prev_agents is assumed to be > 0 and <= n. 'task' is a scalar value
- RETURN: Current agent = returns the agent who can identifies the task and can either solve it or find the person who can solve it.

The table below shows the parameters and the value:

Table 2.1: Parameter value

Parameter's	Value's
Number of servers	4
Packet size	20000 bit per second
No of task	Variable
Number of agents	Variable
Compression efficiency	0.5

3. Results and Discussion

First of all you must give a brief summary about the pervious chapter which is the methodology, we demonstrated the results of our mobile agent simulator with the load balance, and this project addresses un-regulated jobs/tasks migration among the servers. Considering distributed several servers running, a lot of bandwidth is wasted due to unnecessary job migration. It is important to develop a simulation that will address the bandwidth consumption while loads/tasks are being transferred among the servers. This work attempts to regulate this movement to minimize bandwidth consumption and network traffic. The results of this phase of the experiment as illustrated in the graphs Figure 4.8, Figure 4.9, Figure 4.10 and Figure 4.11 show that our simulator performs better.

3.1 Scenario 1

In this first scenario the physical resources are include: 10 node, 1 router, 2 Switches, 1 Mobile Agent Server, and 3 sub servers, and the input parameter: Number of tasks and Number of agents, this two input variable depend on the network design, to determine the efficiency of system.

- Number of tasks = 25 tasks.
- Number of agents = 64 agents.

Our system ask user to enter the number of tasks and number of agents from GUI as shown in Figure (4.1) and then the result shown the compare between efficiency of simple network management protocol (SNMP) with Mobile Agent LAN as shown in figure (4.2 – A and B).

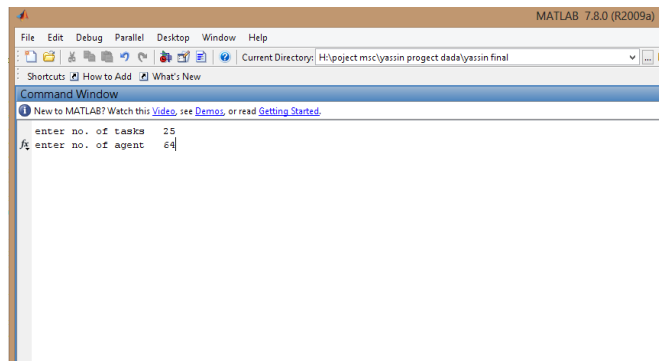
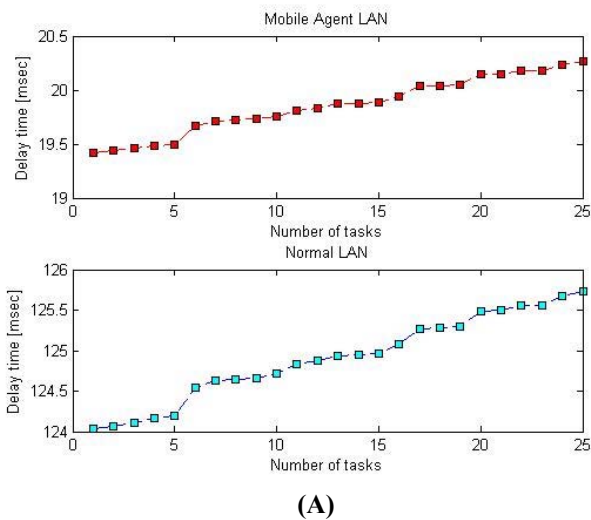
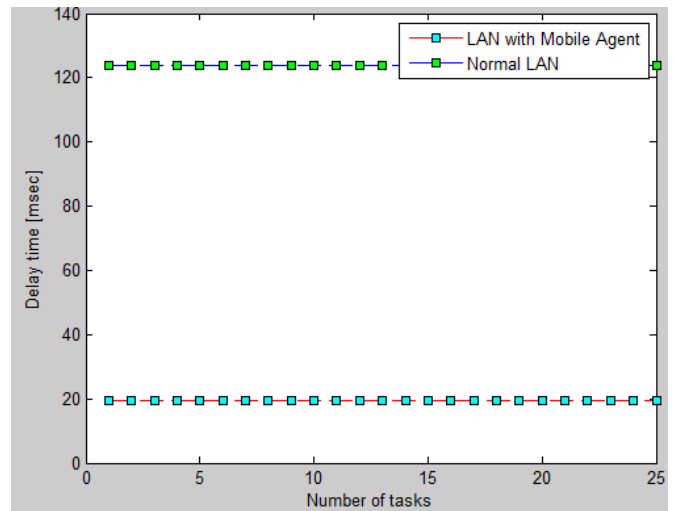


Figure (3.1): GUI for users to enter Number of tasks and Number of agents



(A)

Figure (4.2 – A) show the compression of system efficiency between simple network management protocol (SNMP) with Mobile Agent LAN



(B)

Figure (4.2 –B) show the compression of system efficiency between simple network management protocol (SNMP) with Mobile Agent LAN

3.3 Scenario 2

In this first scenario the physical resources are include: 10 node, 1 router, 2 Switches, 1 Mobile Agent Server, and 3 sub servers, and the input parameter: Number of tasks and Number of agents, this two input variable depend on the network design, to determine the efficiency of system.

- Number of tasks = 25 tasks
- Number of agents = 115 agents

Repeat the same system but enter different number of tasks and number of agents from GUI as shown in Figure (4.3) and then the result shown the compare between efficiency of simple network management protocol (SNMP) with Mobile Agent LAN as shown in figure (4.4 – A and B) compression system efficiency between simple network management protocols (SNMP) with Maximum Mobile Agent LAN

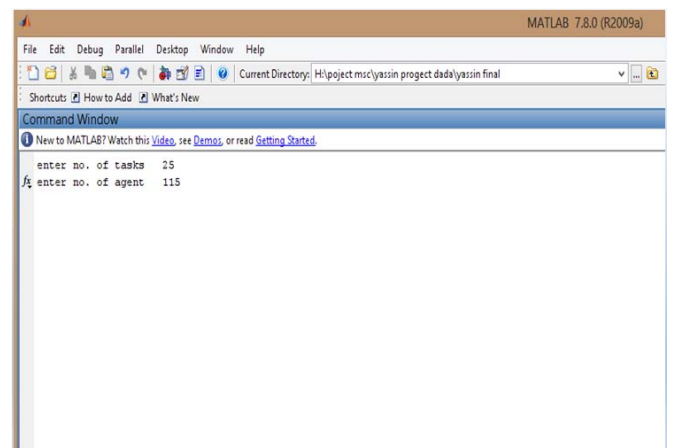


Figure 3.3: GUI for users to enter Number of tasks and Number of agents

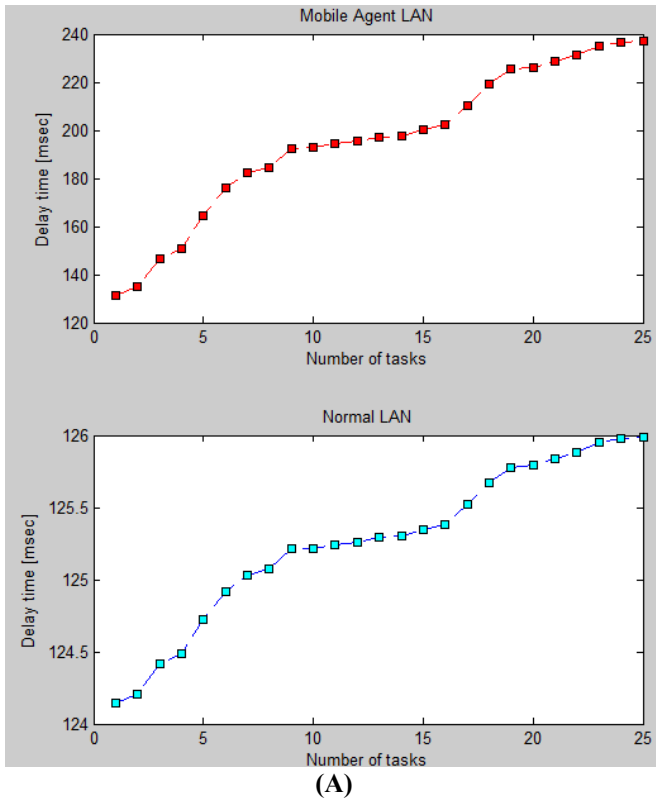


Figure (4.4 – A) compression system efficiency between simple network management protocol (SNMP) with maximum No of Mobile Agent in LAN, this figure approve the number of mobile agent can't be unlimited increase, when the number of mobile agent exceed 115 agent in local network the normal system became better than mobile agent network.

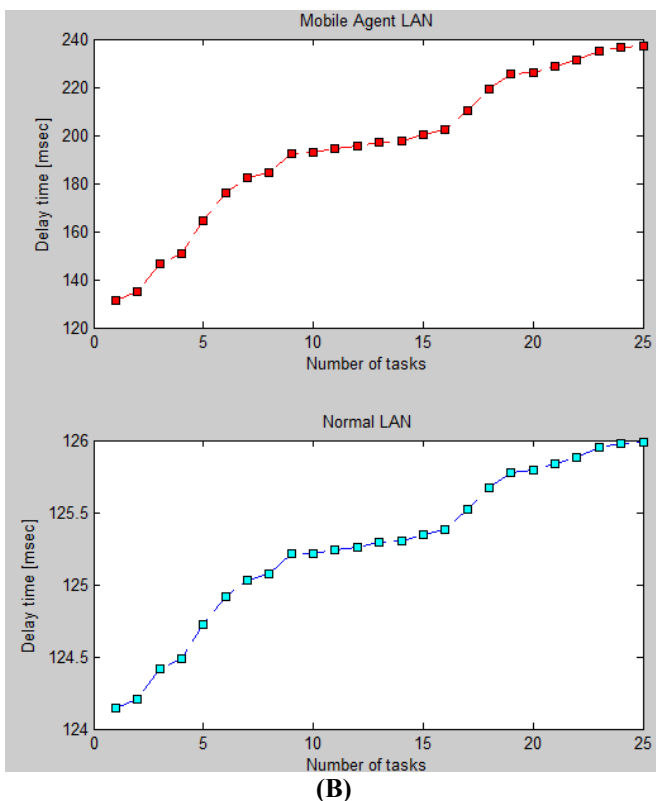


Figure (4.4 – B) compression system efficiency between simple network management protocol (SNMP) with maximum No of Mobile Agent in LAN, this figure approve

the number of mobile agent can't be unlimited increase, when the number of mobile agent exceed 115 agent in local network the normal system became better than mobile agent network.

Figure (3.4 – A and B): compression system efficiency between simple network management protocols (SNMP) with Maximum Mobile Agent LAN

4. Conclusion

According to the results and compression, the mobile agent network based on load balance is better than simple network management protocol (SNMP) and give a high success in simulation environment test, simulator has been developed to enhance the simple network management protocol (SNMP) problem. The simulation show the validity of the algorithm to improve the variance of the overhead load in the network and in providing better average response time, but we found that the maximum number of Mobile Agent used in this study is 110 mobile agents to give a high performance, and if used more than 110 mobile agents the normal LAN system will be better than mobile agent network, also by Increasing the network size the number of mobile agents therefore will increased.

5. Recommendation

The full realization of the proposed system can be better obtained through live and practical implementation of the system, The maximum number of Mobile Agent used in the system is 110 mobile agents to give a high performance, because when used more than 110 mobile agents the simple network management protocol (SNMP) system will be better than mobile agent network LAN.

References

- [1] Peter Braun, Ingo M'uller, Sven Geisenhainer, Volkmar Schau, Wilhelm Rossak," A Service-oriented Software Architecture for Mobile Agent Toolkits", Friedrich Schiller University Jena,2004.
- [2] general magic, inc.sunnyvale, California, U.S.A, Mobile object and Mobile Agent the future of distributing system, <http://www.acm.org/~danny>. 25 May 2006.
- [3] Naveen Ramamurthy dept of computer science and engineering university of Texas at Arlington"Role of mobile agents in mobile computing environment" Department of Computer Science and Engineering University of Texas at Arlington.
- [4] Stephen Preston, What are mobile agents why they are expected.
- [5] MAGDY SAEB, CHERINE FATHY," Modified Diffusion Dynamic Load Yousry El-Gamal, Khalid El-Gazzar, and Magdy Saeb, A Comparative Performance Evaluation Model of Mobile Agent Versus Remote Method Invocation for Information Retrieval, International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering Vol:1, No:3, 2007

- [6] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, L. Salgarelli. (2000). IP micro-mobility support using HAWAII.
Retrieved:<http://www.cse.tkk.fi/fi/opinnot/T110.5190/2000/Micro-mobilitysolutions/micro-mobility.html>
Retrieved from:
<http://comet.columbia.edu/cellularip/overview.htm>
- [7] Lisa Lindqvist, (2000), Micro-mobility for IP. Helsinki University of Technology,
- [8] Han-Chieh Chao. (2001) "An overview and analysis of mobile Internet protocols in cellular environments", Vol. 11(5), pp.435 – 450
- [9] Myung-Kyu Yi, Chong-Sun Hwang. A New IP Paging Protocol for Hierarchical Mobile IPv6 .
- [10] Qing Li (Author), Tatuya Jinmei (Author), Keiichi Shima (Author) Mobile IPv6: Protocols and Implementation Paperback – July 31, 2009