Multiparty Handoff Mechanism in Mobile Ipv4 Networks

K. N. Rama Mohan Babu[1*], K. N. Balasubramanya Murthy[2] and Ram P. Rustagi [2]

[1]Dayananda Sagar College of Enggineering
Bengaluru, India
[2]PES Institute of Technology, Bengaluru, India
[*]Correspondent Author: rams_babu@hotmail.com

Abstract

Internet is evolving as a main infrastructure for wireless and mobile communication. Mobile IPv4 (MIPv4) is a protocol designed by Internet Engineering Task Force (IETF) to support seamless roaming of mobile devices. MIPv4 allows mobile devices (or nodes) to have continuity of communication without any interruption while in mobility. MIPv4 handoff is a process by which mobile node (MN) changes its point of attachment from subnet to a new subnet. In this paper, a multiparty handoff scheme is proposed to minimize the handoff latency and packet loss during rapid movement of MN.

Keywords: *Mobile IPv4, fast handoff, multiparty handoff, handoff latency*

1. Introduction

With the increasing growth of users of mobile devices (or nodes), handling the continuity of communication becomes a challenging task. As a solution a mobility protocol called Mobile IP (MIP) has evolved. In MIP, MNs can roam between the IP networks by keeping the same IP address while continuing ongoing communication. Each MN is always identified by its home address assigned in its home network. A router in home network which tunnels packets to MN when it is away from home network is called home agent (HA). When MN attaches to a network other than its home network called foreign network, it obtains a care-of-address (CoA) representing its current location. A router in foreign network which provides routing services to MN while registered is called foreign agent (FA). MN updates the HA regarding its CoA through FA by using binding update messages. HA then updates its binding cache and tunnels the packets arriving from correspondent node (CN) destined to the MN to its CoA [1, 8]. In MIPv4, services such as agent discovery, registration and tunneling are supported. Through agent discovery, HAs and FAs advertise their availability on each link. Registration allows the MN to register its COA with HA when it attaches to foreign network. Tunneling is used to forward packets from home address to CoA [1, 2].

In Mobile IP, when MN moves between two foreign networks it is referred as handoff. The Mobile IP handoff mechanism is shown in Figure 1. Before handoff, MN keeps receiving packets from FA of old network. During handoff MN moves to a new foreign network and obtains its new CoA (NCoA). After handoff MN informs it's HA about its NCoA through new FA (NFA) and HA updates its binding cache. Subsequent packets are forwarded through NFA.

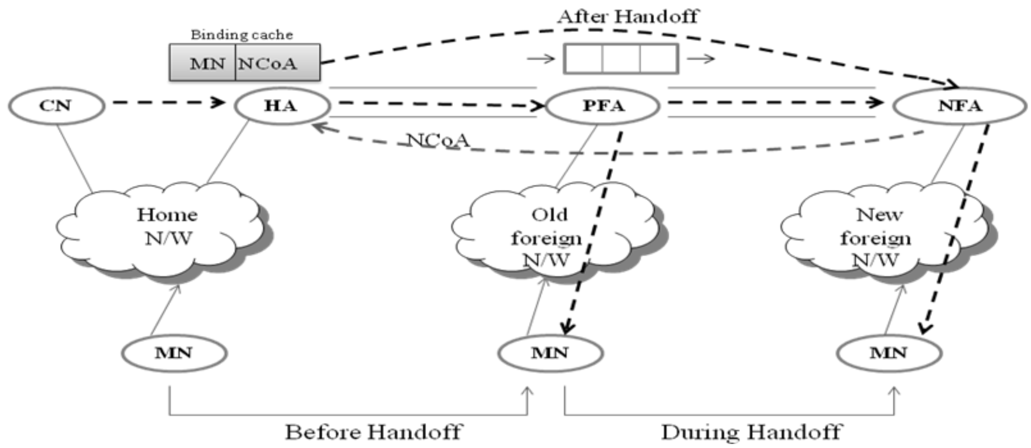

Figure 1. Mobile IP Handoff Mechanism

Handoffs incur significant amount of delay due to movement detection, configuration and binding updates. Such delay is called as handoff latency. During the handoff, packets are not forwarded until communication path with new network is established [3]. The proposed multiparty handoff scheme minimizes handoff latency and takes care of packet loss due to fast movement of MN between several foreign networks using multiparty handoff.

The rest of the paper is organized as follows: Section II discusses the related work and section III presents the proposed scheme. Simulation environment and results are presented in Section IV while Section V provides the conclusion and the scope for future work.

2. Related Work

In [4], Tan et.al. describes Mobile IP by introducing a new object called Domain Foreign Agent (DFA). This approach is called Hierarchical Mobile IP. DFA is connected as a router of the foreign domain under which many FAs of sub domain are connected. Now the location of the MN's CoA is at DFA and CoA changes only when MN moves to different domain. Hence this scheme can avoid handoffs with in the same domain. It involves the use of DFA to handle all MNs within the domain. This DFA has certain capacity with respect to the amount of MNs and binding messages it can handle. Hence the Hierarchical Mobile IP is less scalable. Whereas in the original Mobile IP [1], FAs also have a limit to their capacity but they are lower in architecture, so they have to handle small number of MNs.

Handoffs can occur either in layer 2 (link layer) or in layer 3 (network layer). In [5], Yokota et.al. gives a link layer assisted fast handoff method over wireless LAN networks. This handoff scheme makes use of access points used in wireless LAN environment and dedicated MAC Bridge together to reduce the packet loss. This scheme mainly depends on layer 2 triggers and uses more network bandwidth to carry out handoff.

In [6], Malki. Ed. explains low latency handoffs in Mobile IPv4, which aims to reduce handoff latency using layer 2 triggers. Two techniques called pre-registration and post-registration are described. Pre-registration allows MN to pre-build its registration state at NFA in order to perform layer 3 handoff before it completes layer 2 handoff. Post-registration establishes a bidirectional edge tunnel or unidirectional tunnels to perform low-latency change in the L2 point of attachment for the MN without requiring any involvement by the MN. It operates for two party handoff and three party handoff. Two-party handoff occurs when the MN moves from previous FA (PFA) to NFA. Three party handoff occurs when MN moves from NFA to another FA without performing MIPv4 registration at NFA. Three party handoff is described using Handoff To Third (HTT) request and reply messages. These strategies mainly dependent on layer 2 triggers as an indication and to respond to the upcoming handoff. This increases handoff latency.

In [7], Koodli et.al. presents Mobile IPv4 Fast Handovers which isolates link layer triggers being needed for handoff operation, thereby reducing the handoff latency. Handoff mechanism is described in two modes: Predictive and Reactive. In predictive mode, MN sends Fast Binding Update (FBU) from its previous link and in reactive mode MN sends FBU from its new link. The timeline for predictive handoff is shown in Figure 2.

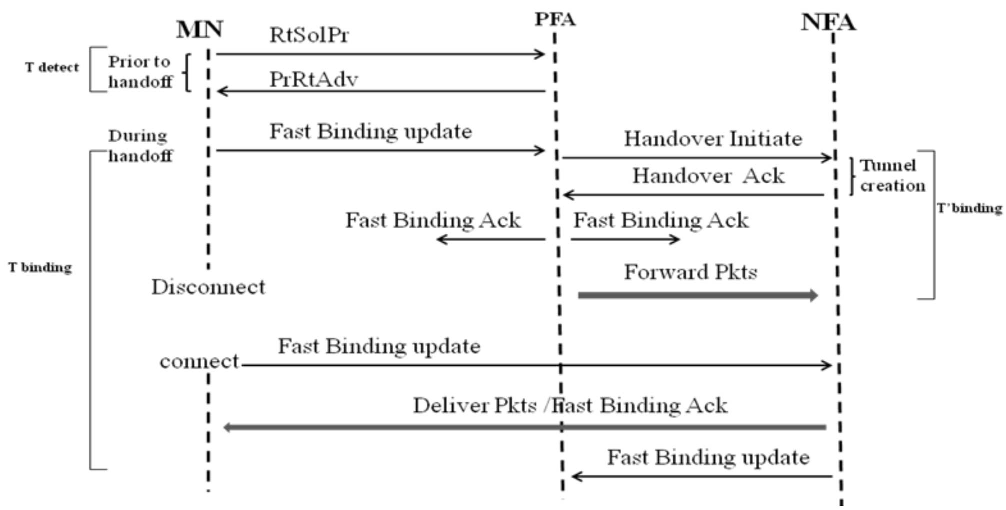

Figure 2. Timeline for Predictive Handoff [7]

It makes use of Proxy Router Solicitation (RtSolPr) and Proxy Router Advertisement (PrRtAdv) to resolve access point (AP) Identifier and subnet information of new AP. When MN decides to handoff, it sends FBU from previous link or new link. FBU enables creation of binding between MN's previous CoA and NCoA. The coordination between the access routers is done using Handover Initiate (HInit) and Handover Acknowledge (HAck). After HInit and HAck are exchanged, data arriving at PFA will be tunneled to NFA which is then delivered to MN. HA will then update NCoA of MN in its binding cache.

The protocol operation is limited by how fast MN can move between the access routers. Access routers here refer to HAs and FAs. In order to address the packet loss caused by rapid movement of MN between the access routers, a multiparty handoff scheme is proposed.

3. Proposed Scheme

Multiparty handoff occurs when MN moves rapidly between the access routers. It is applicable when MN sends FBU and creates tunnel with NFA, moves to another FA without performing Mobile IPv4 registration at NFA. The NFA with which tunnel is established acts as an anchor foreign agent (AFA). Some radio L2 protocols allows MN to move so rapidly from one FA to another FA, such that a probability exists that the Mobile IPv4 registration will not complete before the MN moves on. In such scenario, multiparty handoff should be implemented. Certain wireless systems do not allow such rapid movements and may force Mobile IPv4 registration to complete after establishing tunnel with AFA. In such cases, multiparty handoff is not applicable [6].

In this paper we present a multiparty handoff scheme considering three foreign agents as shown in Figure 3.

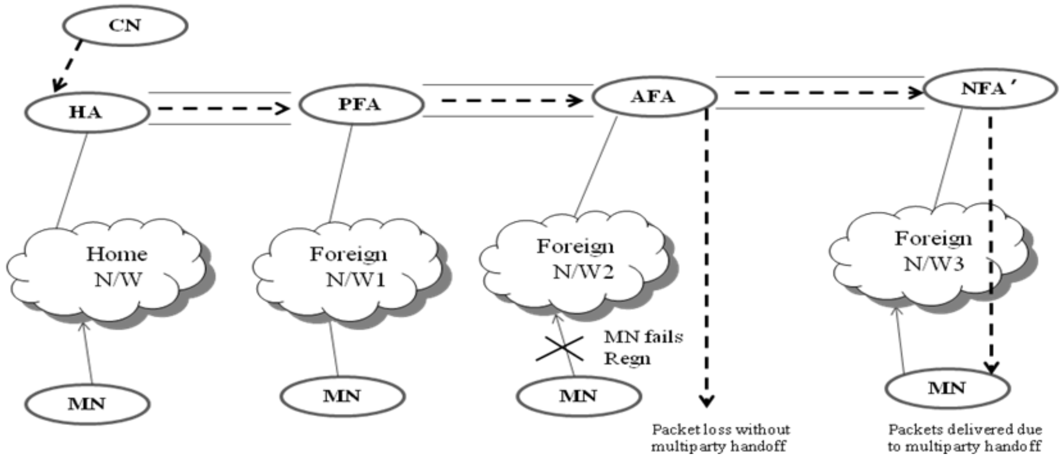

Figure 3. Multiparty Handoff Mechanism

Consider the scenario where MN decides to undergo handoff and sends FBU to PFA, as a result HI and HAck are exchanged between PFA and NFA to create a tunnel. Packets are tunneled to NFA. Further instead of sending FBU to NFA which is needed for Mobile IPv4 registration, MN moves rapidly to another FA (NFA'). This NFA with which tunnel is established now acts as an AFA. This makes Mobile IPv4 registration to be incomplete. Hence the tunneled packets are not delivered to MN until the registration in new network completes. This results in packet loss.

Multiparty handoff aims to reduce the handoff latency in the above scenario by minimizing the time during which MN is unable to send and receive packets due to the delay in Mobile IPv4 registration. This reduces the packet loss caused by rapid movement of MN. Timeline for multiparty handoff is shown in Figure 4.

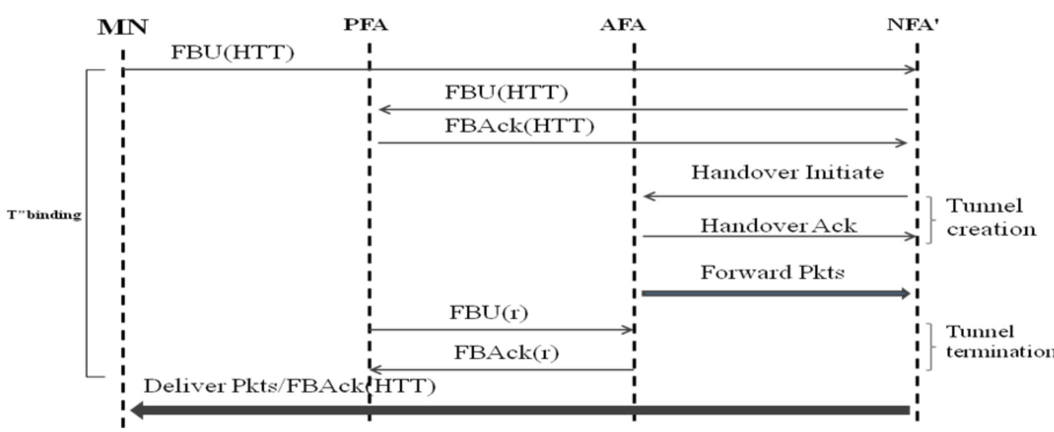

Figure 4: Timeline for Multiparty Handoff

During multiparty handoff MN sends FBU(HTT) to NFA' informing it that multiparty handoff has to take place. NFA' passes on FBU (HTT) to PFA and receives FBAck(HTT). Then HInit and HAck are exchanged between AFA and NFA' to create a tunnel and forward packets to NFA'. In order to cancel the previous tunnel between PFA and AFA, FBU(r) and

FBAck(r) are exchanged. Finally packets tunneled to NFA', are delivered to MN. The message formats remain same as per [7]. For multiparty handoff, few message formats are modified as per the requirement and are discussed here.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |x|x|D|M|G|r|T|x|  reserved   |     Lifetime      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Home Address                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Home Agent                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Care-of Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                                 |
+                         Identification                          +
|                                                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Extensions ...
+-+-+-+-+-+-+-+-
```

Figure 5. Message Format of FBU [7]

In Figure 5, FBU message format is shown. The 16^{th} bit of FBU is a reserved bit and is currently not used. We this bit to differentiate between fast handoff and multiparty handoff. During multiparty handoff, FBU(HTT) is sent with its 16^{th} bit set, so that the receiver performs operation of multiparty handoff. The tunnel flag bits such as T must be set 0 since the tunnel is negotiated between AFA and NFA', not between PFA and NFA. The FBAck(HTT) is identified by matching its 64 bit identification field with that of FBU(HTT). For tunnel cancellation, FBU(r) and FBAck(r) are exchanged between PFA and AFA which are same as FBU and FBAck except the lifetime being set to 0.

In [7], Koodli et.al. describes fast handoff mechanism which aims to reduce handoff latency and packet loss. It addresses movement detection, IP address configuration and binding update latencies during handoff. NCoA is always made to be the new access routers IP address in order to eliminate CoA configuration latency.

The latency resulting during fast handoff [7] is given by

$$T_{handoff} = T_{detect} + T_{CoAconfig} + T_{binding} \quad (1)$$

T handoff = MIPv4 Handoff Latency

T detect = Movement Detection Latency

T CoA config = CoA Configuration Latency

T binding= Binding Update Latency.

Movement Detection Latency is evaluated when MN moves to a new subnet by exchanging RtSolPr and PrRtAdv between MN and PFA. CoA Configuration Latency is the time taken by MN to configure new CoA in foreign network. It depends on whether IP CoA or co-located CoA being used. For eliminating the CoA configuration latency NCoA is always made to be the new access router’s IP address. Binding Update Latency is the time to send binding update message for registration with NFA and acknowledging its reply to MN.

Binding update latency comes from sending binding update message, tunnel creation message and respective acknowledgements as shown in Figure 2.

The latency resulting during multiparty handoff is given by

$$T_{handoff} = T_{detect} + T_{CoA\ config} + T'_{binding} + T''_{binding} \quad (2)$$

$T'_{binding}$ = binding update latency with AFA

$T''_{binding}$= binding update latency with NFA'

Binding update latency with AFA is evaluated by exchanging binding update message, tunnel creation message and their replies among MN, PFA and NFA as shown in Figure 2. NFA in Figure 2 becomes AFA for multiparty handoff as shown in Figure 4. Binding update latency with NFA' is evaluated by exchanging binding update (HTT) message, tunnel creation message, tunnel termination message and their replies among MN, PFA, AFA and NFA' as shown in Figure 4.

Algorithm for MN

```
// prior to handoff, MN is at PFA

        Step 1: Send RtSolPr to PFA
                Receive PrRtAdv from PFA

//during fast handoff, MN is moving to NFA

        Step 2: Send FBU with 16th bit=0 to PFA

        Step 3: If (FBU with 16th bit=0)

                then

                    Receive FBAck from PFA
                    Disconnect from PFA
                    Connect to NFA
                    Send FBU to NFA

                If (FBAck is not received from PFA)

                  then

         Receive FBAck  from NFA
         Receive  packets from NFA
        Goto Step5

// during multiparty handoff,MN is moving to NFA'
```

Step 4: If (FBU (HTT) with 16th bit=1 to NFA')
Receive packets & FBAck (HTT) from NFA'

// after handoff

Step 5: update HA's binding cache with new CoA

Algorithm for PFA

// prior to handoff, PFA cache the neighboring routers PrRtAdv, MN is at PFA

Step 1: Receive RtSolPr from MN
Send PrRtAdv to MN

Step 2: FBU with 16th bit=0 is received from MN
Send HInit to NFA
Receive HAck from NFA
Send FBAck to MN and NFA
Forward packets to NFA

//when MN moves to NFA'

Step 3: FBU (HTT) with 16th bit=1 is received from NFA'
Send FBAck (HTT) to NFA'
Receive FBU(r) from AFA
Send FBAck(r) to AFA

// tunnel termination occur and no more role to play

Algorithm for NFA

Step 1: Receive HInit from PFA
Send HAck to PFA

Step 2: Receive FBAck from PFA
Receive Packets from PFA
Step 3: Receive FBU from MN
Deliver packets and FBAck to MN
Send FBU to PFA

Algorithm for AFA

Step 1: Receive HInit from NFA'
Send HAck to NFA'

Step 2: Forward packets to NFA'

Algorithm for NFA'

Step 1: FBU (HTT) with 16^{th} bit=1 is received from MN
Send FBU (HTT) to PFA
Receive FBAck (HTT) from PFA
Send HInit to AFA
Receive HAck from AFA
Receive Packets from AFA
Deliver packets and FBAck (HTT) to MN

4. Simulation

The model for multiparty handoff scheme is simulated using widely used NS2 simulator [9]. For the purpose of simulation the test bed created is as shown in Figure 6.

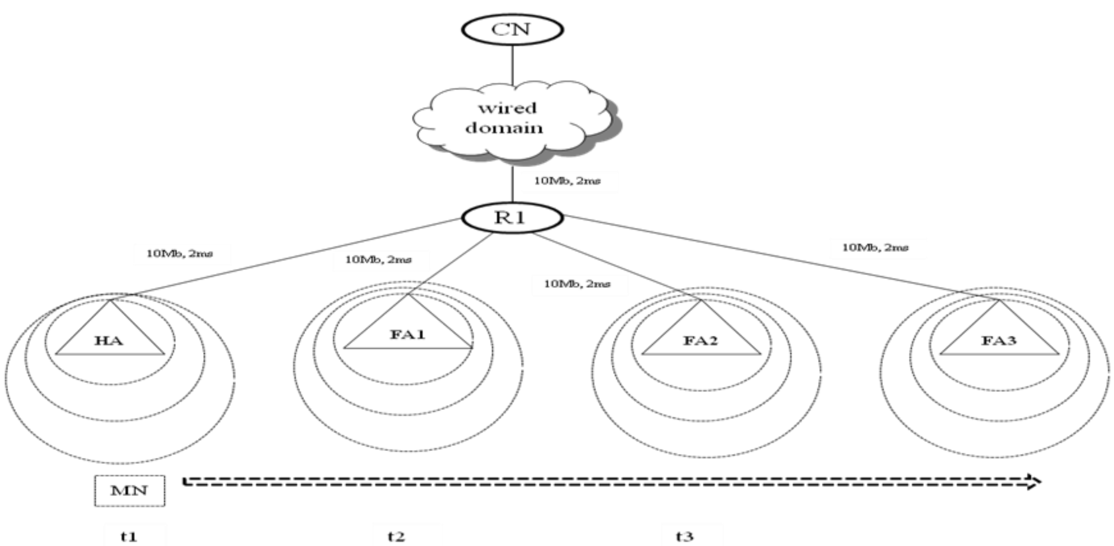

Figure 6: Simulation Test Bed

To illustrate multiparty handoff, 1 HA, 1 MN and 3 FAs namely FA1, FA2 and FA3 are considered. At time t1, MN leaves HA and moves towards FA1. At time t2, MN leaves FA1 and moves towards FA2. At time t3, MN leaves FA2 and moves towards FA3. From the simulation, the handoff latency is evaluated for fast handoff involving 2 FAs [7] i.e. FA1 and FA2. Fast handoff involving 3 FAs is evaluated by using [7], between FA1 and FA2, FA2 and

FA3. Proposed multiparty handoff latency is evaluated for 3 FAs i.e. FA1, FA2 and FA3. Simulation results obtained from simulation trace file are shown in Table 1.

Table 1. Simulation Results

	T detect	T binding		Total handoff latency
Fast handoff between 2 FAs [7]	232.30 ms	267.92 ms		498.22 ms
Fast handoff between 3 FAs [7]	498.22*2 ms			996.44 ms
Multiparty handoff between 3 FAs	232.30ms	T′binding	T″binding	896.58 ms
		222.97 ms	441.31 ms	

From the simulation results shown in Table 1, fast handoff between 3 FAs [7] takes 996.44ms. With the proposed multiparty handoff between 3 FAs takes 896.58ms. Hence the proposed scheme reduces 99.86 ms (10%) of handoff latency.

5. Conclusion and Future Work

This paper proposes a multiparty handoff mechanism to minimize Mobile IPv4 registration time thereby reducing 10% of handoff latency. The proposed scheme has been designed to take care of packet loss during fast movement of MN between networks. The time minimization has been achieved by reusing registration messages as well as by optimizing the bandwidth usage during handoff mechanism. Further research work is being carried out to use multiparty handoff mechanism by considering more than 3 FAs and also for IPv4 / IPv6 mixed environments.

References

[1] Perkins C, “IP Mobility Support”, RFC 2002, IETF, **(1996)** October.

[2] Perkins C, “IP Mobility Support for IPv4”, RFC 3344, IETF, **(2002)** August.

[3] Yusof AL, Ismail M, Misran N, “Handoff Architecture in Next Generation Wireless Systems”, proceedings of APACE 2007, **(2007)** December, pp. 1-5.

[4] Tan CL, Lye KM, Pink S, “A Fast Handoff Scheme for Wireless Networks”, Proceedings of the 2nd ACM International workshop on Wireless mobile multimedia, **(1999)**, pp. 83-90.

[5] Yokota H, Idoue A, Hasegawa T, Kato T, “Link Layer Assisted Mobile IP Fast Handoff Method over Wireless LAN Networks”, Proceedings of the 8th annual international conference on Mobile computing and networking MobiCom 02 , **(2002)**. pp. 131-139.

[6] El Malki K, “Low-Latency Handoffs in Mobile IPv4”, RFC 4881, IETF, **(2007)** June.

[7] Koodli R, Perkins C, “Mobile IPv4 Fast Handovers”, RFC 4988, IETF, **(2007)** October.

[8] Perkins CE, “Mobile IP”, IEEE Communications Magazine, **(1997)** May, pp. 84-99.

[9] Network simulator tool, http://www.isi.edu/nsnam.

Authors

Mr. K N Rama Mohan Babu obtained his B.Tech degree in Computer Engineering from Mangalore University, India. M.S. from BITS-PILANI India. Currently he is PhD Research scholar, Dr. MGR University, Chennai, India & working as Assit. Professor at Dayanada Sagar College of Engineering Bengaluru, India.

Dr. K N Balasubramanya Murty is presently working as the Principal & Director of PES Institute of Technology, Bengaluru, India. He has been working in the area of parallel computing and computer networks. Since 1990 he has published several research papers in journals and conferences of international repute. He obtained his master degree from IISC, Bengaluru and PhD from IIT, Chennai.

Dr. Ram P.Rustagi is obtained his M.Tech degree from IISC, Bengaluru and PhD from IIT-Delhi and he has been working in the area of Networks. Currently, he is working as a professor in department of Information Science at PES Institute of Technology, Bengaluru India. Previously, he worked in Software industry for 25+ years.