

Modeling and Simulation of Partially Distributed Mobility Management Scheme

Jang-Geun Ki¹, Kyu-Tae Lee² and Do-Hyeun Kim³

¹*Dept. of Electrical and Electronic and Control Engineering, Konju Nat'l Univ., Korea*

²*Dept. of Information and Communication Engineering, Konju Nat'l Univ., Korea*

³*Dept. of Computing Engineering, Jeju Nat'l Univ., KOREA*

¹*kjg@kongju.ac.kr, ²ktlee@kongju.ac.kr, ³kimdh@jejunu.ac.kr*

Abstract

Most existing mobility management deployments such as MIPv6, PMIPv6 and 3GPP utilize centralized mobility anchors in the hierarchical network architecture. Centralized mobility schemes are subject to restrictions on reliability of SPOF (Single Point Of Failure) and network scalability, etc. As an alternative to the centralized approaches, DMM (Distributed Mobility Management) mechanisms have been actively and widely studied these days. In DMM, functions of the anchor node are efficiently distributed to access routers. This paper suggests the network-based partially distributed mobility management mechanism and performs extensive simulations under various traffic environments for verification and evaluation of the suggested scheme. Performance indexes such as Tx/Rx UDP traffic, end-to-end data transfer delay, binding delay, CPU utilization, WLAN throughput are analyzed and compared among MIPv6, PMIPv6 and the suggested DMM scheme. Simulation results show that the DMM scheme suggested in this paper is useful for IP mobility management research in wireless environment.

Keywords: *Partially Distributed Mobility Management, Simulation*

1. Introduction

Conventional solutions for IP mobility are usually based on the main idea that the specific anchor node maintains the up-to-date binding information between HoA (Home Address) and CoA (Care-of-Address) of the MNs (Mobile Nodes). In these schemes, data traffic is delivered through tunneling between the anchor node and MNs or between the anchor node and ARs (Access Routers). Therefore, the anchor node should not only manage the binding information of all MNs in a centralized fashion but also perform encapsulation and de-capsulation of all data traffic transmitted and received by all MNs. As an example, in MIPv6 (Mobile IPv6) and PMIPv6 (Proxy MIPv6) protocols, all signaling messages and data traffic converge on a HA (Home Agent) or LMA (Local Mobility Anchor) node functioned as an anchor node [1-4]. Centralized mobility solutions are subject to several problems and limitations such as a bottleneck and SPOF (Single Point Of Failure), network scalability, and sub-optimal routing paths [5].

During the recent years, a new trend in mobile network evolution for solving the centralized mobility drawbacks is to flatten mobility architecture by confining mobility support in the access network, named DMM (Distributed Mobility Management), which would keep the rest of the network unaware of the mobility events and their support [6]. Distributed mobility solutions have attracted great interest of researchers and are being actively standardized by the IETF dmm (Distributed Mobility Management) and netext(Network-Based Mobility Extensions) working groups [7-8].

In this paper, the network-based partially distributed mobility management scheme is suggested and evaluated by simulations under various traffic environments. The

simulation results show that the developed models for DMM are useful for mobility management studies in wireless network environment. The rest of this paper is organized as follows. After explaining the background on distributed mobility management in Section 2, this paper describes the operation of the suggested partial DMM scheme in Section 3. Several simulation results for verification and performance evaluation of the partial DMM are stated in Section 4 and final conclusion is discussed in Section 5.

2. Distributed Mobility Management Overview

MIPv6 [1] is a typical host-based IPv6 mobility solution, which requires client functionality in the IPv6 stack of a MN. Unlike MIPv6, where the MN signals its location changes to the HA (Home Agent), PMIPv6 [2], which is an archetypal network-based mobility solution, introduces a MAG (Mobile Access Gateway) node and a LMA (Local Mobility Anchor) node. The MAG acts as a proxy for the MN and the LMA does the same function as the HA. Both MIPv6 and PMIPv6 are based on centralized mobility approach in which a central mobility anchor, *i.e.*, HA or LMA, manages all data packets as well as signaling messages to be sent or received by the MN and the CN. Figure 1 illustrates data paths and tunneling sections in centralized mobility management approaches, MIPv6, PMIPv6 and 3G/LTE.

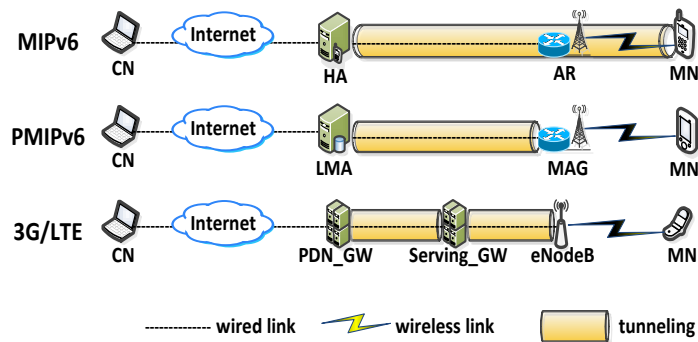


Figure 1. Data Paths and IP Tunnels in Centralized Mobility Management Approaches

As stated in [5, 6] and [9], centralized mobility solutions do not process data and control messages separately and are subject to several problems and limitations: longer routing paths, network scalability, signaling message overhead and longer associated handover delay, more complex network deployment, higher vulnerability of a potential SPOF(Single Point Of Failure), and lack of flow-based mobility management granularity.

To solve the problems and limitations and efficiently support mobile users, the netext (Network-based Mobility Extension) working group and dmm (Distributed Mobility Management) working group in IETF (Internet Engineering Task Force) have studied and proposed various distributed mobility management solutions. They distribute the mobility functions to access routers while bringing them closer to a MN so that the MN is served by the closest network. By topologically distributing mobility anchors, MNs are able to be managed in a decentralized way and mobile data traffic can also be distributed.

Development and deployment of DMM mechanism can be divided into two main categories: i) partially distributed by removing the data path constraint towards the anchor, but maintaining a centralized control plane, and ii) fully distributed by eliminating any centralized role in the architecture and distributing both routing and control functions among access routers.

In general, volume of user data traffic is much higher than that of control signaling traffic. Therefore, by separating the control and data planes and applying a distributed architecture to the data plane only, cost-effective DMM can be achieved. On the other hand, fully distributed mobility needs the introduction of special mechanism in order to identify the anchor node that manages mobility signaling and data forwarding of a particular MN. In most cases, this also requires the absolute distribution of mobility context database (*e.g.*, for binding information) between every element of the distributed anchor system.

In this paper, the network-based partially distributed mobility management solution is suggested and the simulator for model verification and performance evaluation of the suggested mechanism is implemented and extensive simulations under various traffic environments are performed.

3. Network-based Partial DMM Mechanism

The DMM (Distributed Mobility Management) scheme and simulation models developed in this paper have been adapted on the basis of PMIPv6 protocol. The detailed operation of the scheme is explained in this section and performance evaluation and comparison with other protocols are described in the next section.

3.1. Initial Attachment Procedure

Figure 2 shows the initial attachment process that is performed when a MN enters the DMM domain. Once a MN enters a new network, MN's WLAN MAC layer will do a scan procedure to find an AP (Access Point) to be connected. In Figure 2, each MAAR (Mobility Anchor and Access Router) node has an AP function. When a MN finds an AP, it will start the procedure to associate with the new AP. During this association procedure, the AP (MAAR) node gets the MN's MAC (Media Access Control) address as a MN_ID (MN's Identification) value and will become the MN's Home Agent.

MAAR's WLAN data link layer process informs the DMM process in IP layer about MN's association and delivers the MN_ID value. DMM process in MAAR node allocates a HNP(Home Network Prefix) for a MN and sends a PBU(Proxy Binding Update) message that includes {MN_ID, HNP} information to a CMD(Central Mobility Database) node. Address space of the HNP allocated in each MAAR node is the sub-space of the DMM domain controlled by the CMD.

The CMD node creates the new entry {MN_ID : (HNP, MAAR_ip)} and inserts it in the BCE(Binding Cache Entry) table and then replies with a PBA(Proxy Binding Acknowledgement) message to the MAAR. After receiving the PBA message, the MAAR node creates a RA (Router Advertisement) message that includes the allocated HNP and sends it to the MN. The MN configures an IPv6 address by using the received HNP information and starts to use it.

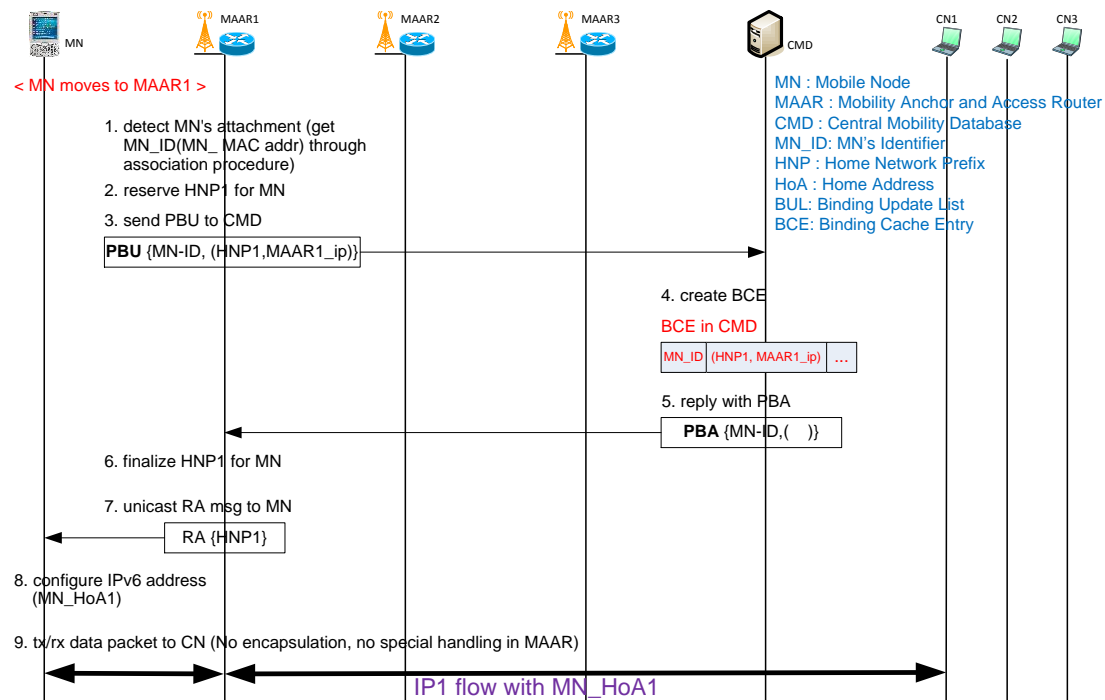


Figure 2. Initial Attachment Process of the Partial DMM

3.2. Handover Procedure

When a MN is moving away from the area covered by one MAAR node and entering a new area covered by another MAAR, the handover process is performed as shown in Figure 3 and the on-going session is transferred to the second MAAR in order to avoid session termination when the MN gets out of the range of the first MAAR.

When the MN enters the new MAAR domain, initial attachment process is performed as described in the previous section and the MN gets the new IPv6 address which will be used for new communication sessions to be started from now on. In the meantime, the on-going session keeps using the old IPv6 address that was gotten from the previous MAAR node that the MN visited when the session started.

After detecting the approach of the MN, the new MAAR allocates a new HNP and creates a PBU message that includes the MN_ID and the allocated HNP and sends it to the CMD. When the CMD receives the PBU, it uses the MN_ID as a key to search the BCE table. The matched entry is updated and becomes the form of {MN_ID : (old_HNP, old_MAAAR_ip) : (new_HNP, new_MAAAR_ip)}. After updating the BCE entry, the CMD replies to the new MAAR with the PBA message that contains the information about old MAARs, {MN_ID : (old_HNP, old_MAAAR_ip)}. At the same time, The CMD sends to the old MAAR the PBU message with the information of {MN_ID : (old_HNP, new_MAAAR_ip)}.

When receiving the PBA message, the new MAAR node inserts the information of {MN_ID : (old_HNP, old_MAAAR_ip)} into the BU(Binding Update) list and establishes a tunnel to the old MAAR. The entry in the BU list is removed when the MN goes far away from the MAAR node. In the meantime, the old MAAR node that is receiving the PBU message from the CMD node updates the BCE table entry by using the information of {MN_ID : (old_HNP, new_MAAAR_ip)} and establishes a tunnel to the new MAAR and replies to the CMD with the PBA message.

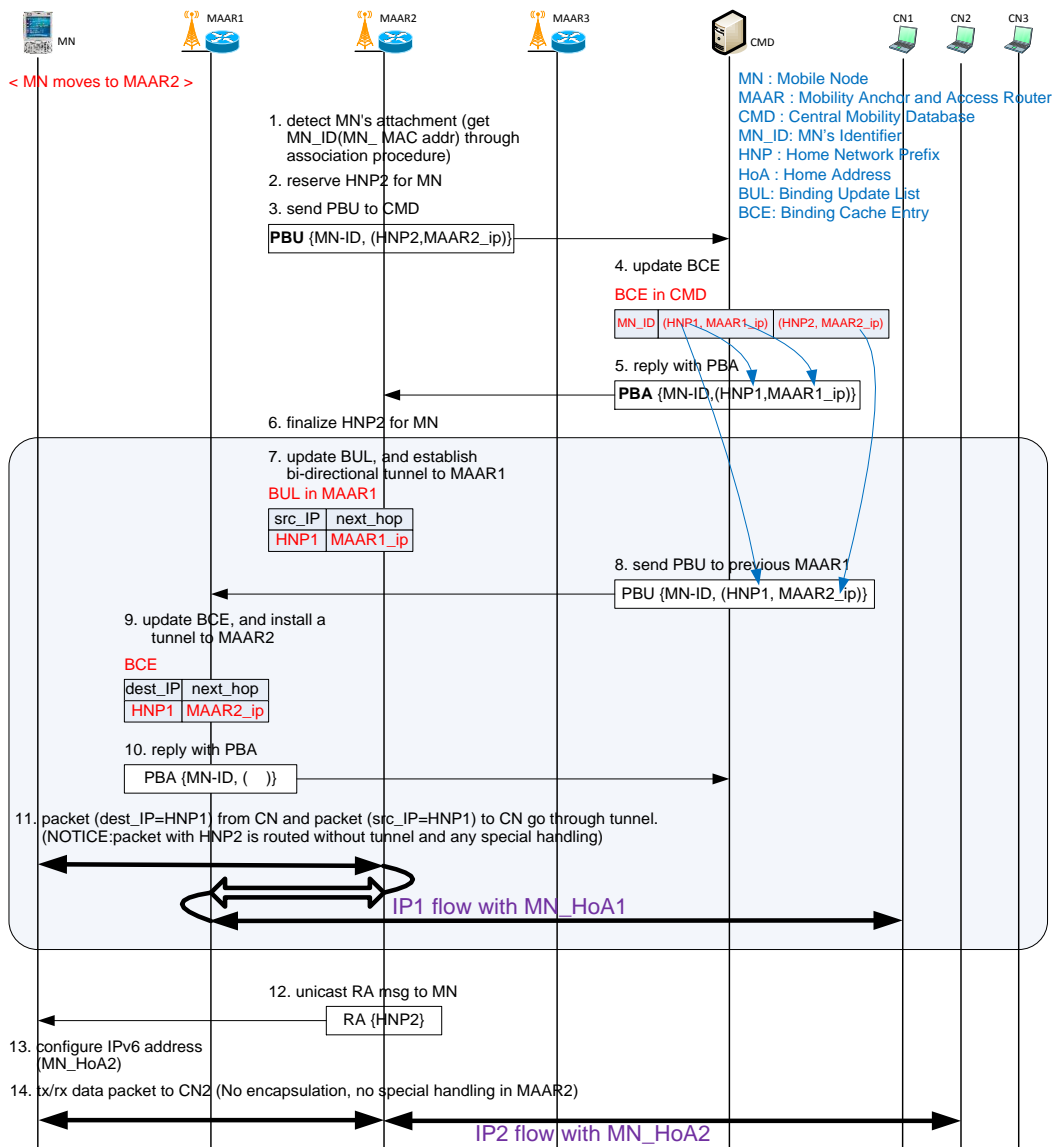


Figure 3. Handover Process of the Partial DMM

Figure 4 shows the IP data flows after handover. On-going IP1 flow continues through the tunnel between old MAAR1 node and new MAAR2 node, whereas new IP2 data flow started with new HNP2 exchanges traffic directly. The rule to forward data packet in a MAAR node is described in Figure 5.

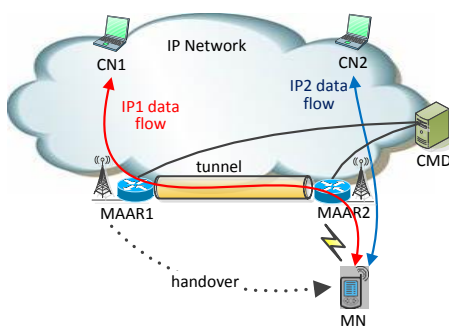


Figure 4. Data Flows after Handover in Partial DMM Scheme

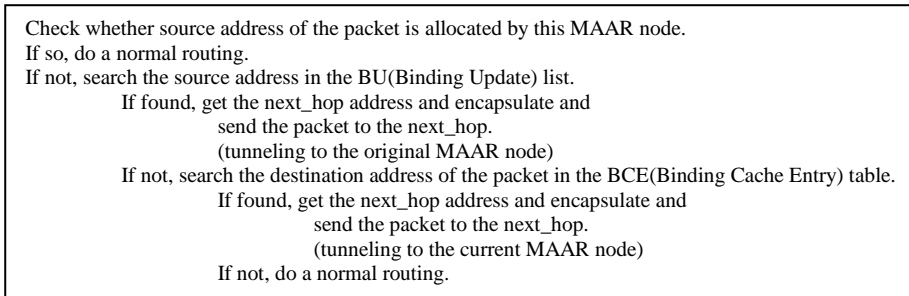


Figure 5. Data Forwarding Rule in a MAAR Node

3.3. Revisiting the Previous MAAR Node

If the MN revisits the MAAR node to which it connected before, information in the BCE table and the BU list becomes entangled in a loop. To solve this recursive problem, when establishing the association relationship between the MN and the MAAR, the MAAR node checks whether the BCE table already has information about the MN and, if found, deletes the information and then performs the handover procedure described in the previous section.

4. Model Verification and Performance Evaluation

In order to evaluate the proposed distributed mobility management scheme and verify the developed simulator, multiple simulations are performed under various traffic environments. DMM's performance is compared to that of MIPv6 and PMIPv6.

DMM network model for simulation is developed by using the OPNET [10] and is illustrated in Figure 6. The current version of the OPNET supports MIPv6 protocol only. The simulation codes and node models for PMIPv6 and partial DMM protocols are developed in this paper. Network models for MIPv6 and PMIPv6 protocols have the exactly same topology with DMM protocol and the CMD and MAAR nodes are replaced with the HA and AR nodes in MIPv6 and with the LMA and MAG nodes in case of PMIPv6.

In the Figure 6, seven MAAR nodes are placed across the network and two MNs have been set to follow the trajectories across MAARs. MN01 is initially attached to the MAAR2 and moves along the MAAR3 through MAAR7 in a clockwise direction and then go back to the original starting point via MAAR2. MN02 is attached to the MAAR5 at startup and then turns clockwise from MAAR5 to MAAR5 again and then its original position. While MNs are moving at a speed of 60Km/h, they continually transmit and receive UDP data traffic to and from CNs at an average rate of 20 packets/sec. Average packet length is 1024 bytes. Since Wi-Fi is the most widely deployed wireless access technology nowadays, IEEE 802.11g WLAN(Wireless Local Area Network) technology with the rate of 11 Mbps is used as a data link layer protocol in our simulation model. Packet latency in the Internet cloud is assumed to be 10 msec.

Figure 7 depicts the WLAN AP connectivity of MN01 and MN02 during simulation. The Y-axis value means the connected AP BSS-ID number and corresponds to the suffix of the MAAR node name. The Y-axis value of -1 means no connection to any AP. From the figure, it is confirmed that each MN establishes a connection to a MAAR node while it is moving around during the simulation period.

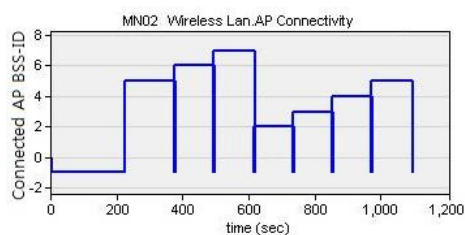
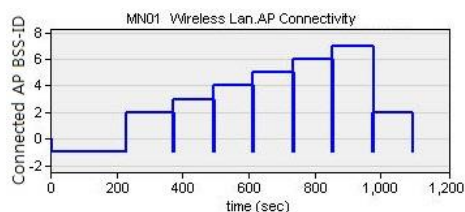
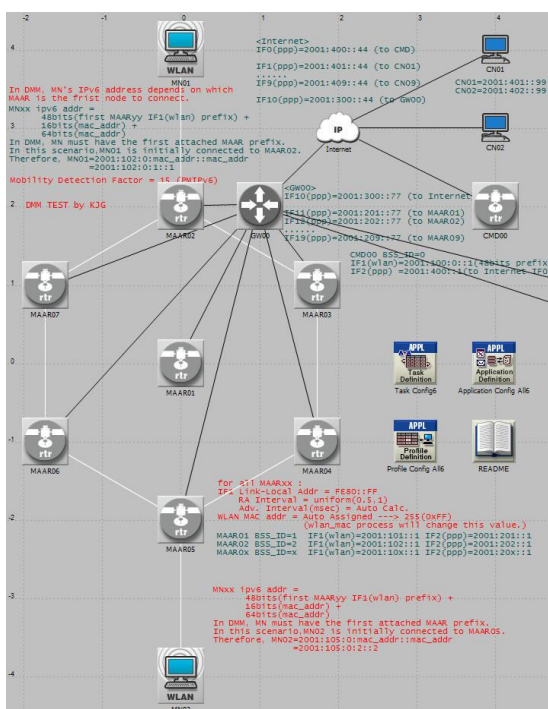


Figure 6. Simulation Network Model

Figure 7. AP Connectivity of the MN

Figure 8 shows the simulation results for the Tx/Rx UDP traffic when MNs and CNs generate 80 packets per second to each other. Almost all transmitted UDP packets are received except the handover periods. Packet loss during handover is caused by WLAN characteristics. The MN moving from the coverage area of one AP to another new AP needs to perform the scanning procedure to connect to the new AP. In other words, roaming between APs is accomplished by scanning for beacons from APs. In our simulation model, it takes a little time to start the new scanning procedure after the MN is disconnected from the old AP. End-to-end packet transfer delay is shown in Figure 9. The average delay in DMM is smaller than that of PMIPv6 and MIPv6 because data packets in DMM go through Internet cloud only one time while data packets in PMIPv6 and MIPv6 have to traverse two times via LMA and HA respectively.

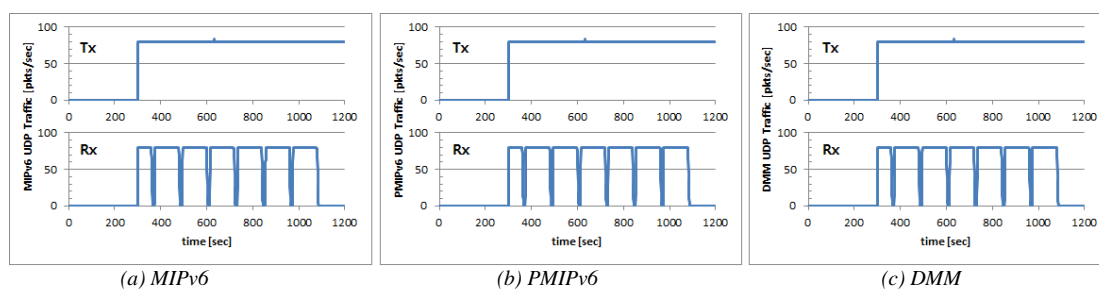


Figure 8. Tx/Rx UDP Traffic

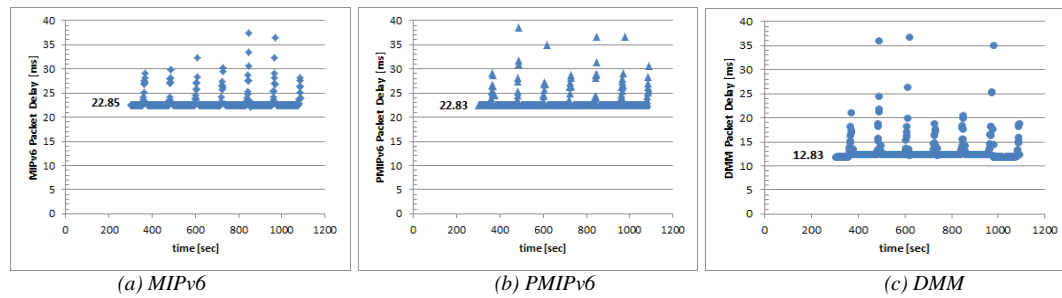


Figure 9. End-to-end Packet Transfer Delay

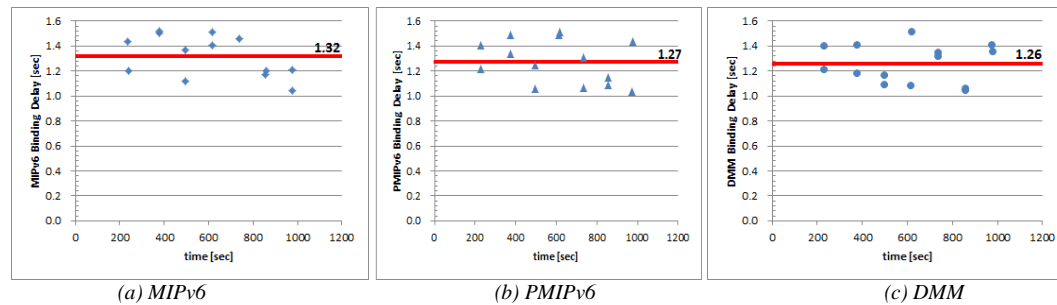


Figure 10. Binding Information Registration Delay

Figure 10 shows binding (registration) delay performance in each protocol. Binding delay is the period of time from the moment a BU or PBU message is sent to the moment a BA or PBA message is received in response by the originator of the BU or PBU message. In MIPv6, a BU message is sent by a MN and goes to a HA through the WLAN network. On the other hand, both PBU messages sent by a MAG router to a LMA in PMIPv6 and sent by a MAAR router to a CMD node in DMM do not visit the wireless network to reach their destinations. Therefore, binding delay in MIPv6 is usually higher than that of PMIPv6 or DMM if all other conditions are the same even though it is very hard to notice the difference in our simulation results because packet latency in the Internet cloud is much bigger than the propagation and transmission delay in wireless network. The main cause of binding delay except the latency in the Internet is the delay at the HA to perform the DAD (Duplicate Address Detection) procedure. In the HA, the delivery of the BA message is delayed by $(\text{DupAddrDetectTransmits} * \text{Neighbor solicitation interval})$ to model the time it would take to perform DAD as stated in the RFC4862 [11]. In this expression, DupAddrDetectTransmits (default value = 1) means the number of neighbor solicitation messages to be sent while performing DAD. Neighbor solicitation interval (default value = uniform[1000..1500] ms) is the time, in milliseconds, between retransmitted Neighbor Solicitation messages and is used by Address Resolution and Neighbor Unreachability Detection algorithms.

For comparison of CPU utilizations of CMD, LMA and HA in DMM, PMIPv6, and MIPv6 respectively, simulation results are shown in Figure 11. CPU utilization models the IP packet forwarding delays and application processing delays in the node. The more packets to be forwarded, the higher CPU utilization to be achieved. In the result graphs, CPU utilization of CMD node in DMM protocol is much lesser than those of HA in MIPv6 and LMA in PMIPv6 because data packets in DMM are distributed across all MAARs while all data traffics in PMIPv6 and MIPv6 are transmitted by way of a LMA and a HA respectively.

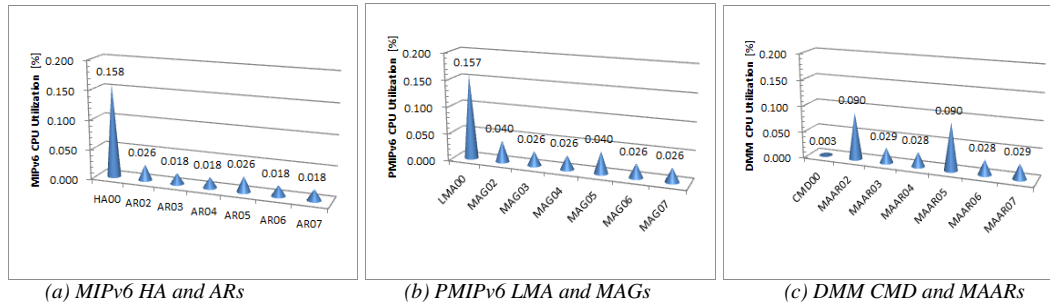


Figure 11. CPU utilizations in MIPv6, PMIPv6 and partial DMM

WLAN throughput in the whole network and WLAN load in a MN is depicted in Figure 12 and Figure 13 respectively. WLAN throughput represents the total number of bits (in bits/sec) forwarded from wireless LAN layers to higher layers in all WLAN nodes of the network. MN's WLAN load means the traffic volume submitted to the wireless LAN MAC by its higher layers in a MN node. The throughput and load for MIPv6 is a little higher than that of PMIPv6 and DMM because, in case of MIPv6, tunneled data packets that have one more outer IPv6 header are transmitted through the wireless network. In MIPv6, tunneling is established between MN and HA including WLAN area, while the tunneling path in PMIPv6 is set up between LMA and MAG only in the wired network. In DMM, tunneling is performed between the first MAAR to which the MN was initially attached and the present MAAR to which the MN is currently connected. In general, the tunnel in DMM is the shortest and the next is the tunnel in PMIPv6 and the longest is the MIPv6 tunnel

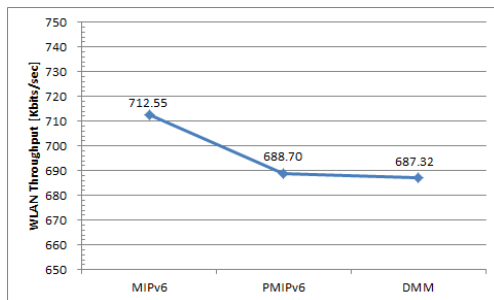


Figure 12. WLAN throughput in the Whole Network

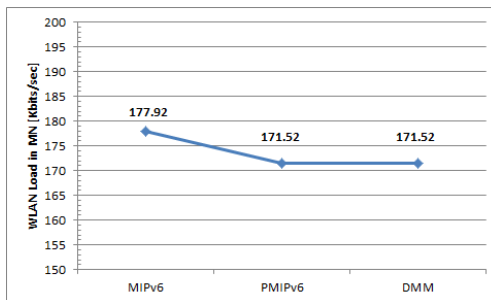


Figure 13. WLAN Load in a MN

5. Conclusion

In most conventional centralized mobility management mechanisms including MIPv6 and PMIPv6, signaling messages and data traffic are concentrated on the anchor node such as a HA in MIPv6 and a LMA in PMIPv6. To solve these bottleneck and SPOF (Single Point Of Failure) problems, distributed mobility management mechanisms have been actively and widely studied in and around the IETF dmm (Distributed Mobility Management) and netext (Network-Based Mobility Extensions) working groups

After extensive and comprehensive studying about centralized and distributed mobility managements, the network-based partial DMM scheme is suggested in this paper and simulation models and programs have been developed on the basis of PMIPv6 protocol and verified by simulations under various traffic environments.

The simulation results for performance comparison of DMM, PMIPv6 and MIPv6 protocol include Tx/Rx data traffic, End-to-end packet transfer delay, binding delay, CPU utilization, WLAN throughput and load, etc. and show that the developed

models for DMM are useful for mobility management studies in wireless network environment.

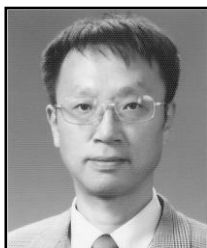
Acknowledgement

This research was supported by the ICT Standardization program of MISP (The Ministry of Science, ICT & Future Planning), Corresponding author; Do-Hyeun Kim (e-mail: kimdh@jejunu.ac.kr).

References

- [1] C. Perkins, D. Johnson and J. Arkko, "Mobility Support in IPv6", IETF RFC6275, (2011) July.
- [2] Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury and B. Patil, "Proxy Mobile IPv6", IETF RFC5213, (2008) August.
- [3] Yokota, K. Chowdhury, R. Koodli, B. Patil and F. Xia, "Fast Handovers for Proxy Mobile IPv6", IETF RFC5949, (2010) September.
- [4] Soliman, C. Castelluccia, K. ElMalki, Athonet, and L. Bellier, "Hierarchical Mobile IPv6 (HMIPv6) Mobility Management", IETF RFC5380, (2008) October.
- [5] Chan, D. Liu, P. Seite, H. Yokota and J. Korhonen, "Requirements for Distributed Mobility Management", IETF, draft-ietf-dmm-requirements-15 (work in progress), (2014) March.
- [6] Giust, A. de la Oliva and C. J. Bernardos, "Flat access and mobility architecture: An IPv6 distributed client mobility management solution", 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), April, (2011), pp. 361-366.
- [7] <http://datatracker.ietf.org/wg/dmm/charter/>, Distributed Mobility Management (dmm) working group, (2014).
- [8] <http://datatracker.ietf.org/wg/netext/charter/>, Network-Based Mobility Extensions (netext) working group, (2014).
- [9] Anthony Chan, H. Yokota, J. Xie, P. Seite and D. Liu, "Distributed and Dynamic Mobility Management in Mobile Internet: Current Approaches and Issues", Journal of Communications, vol. 6, no. 1, pp. 4-15, (2011) February
- [10] Simulator, <http://www.opnet.com>, (2014).
- [11] Thomson, T. Narten and T. Jinmei, "IPv6 Stateless Address Auto-configuration", IETF RFC4862, (2007) September.
- [12] Al-Surmi, M. Othman and B. Mohd Ali, "Mobility management for IP-based next generation mobile networks: Review, challenge and perspective", Journal of Network and Computer Applications, vol. 35, no. 1, January (2012), pp. 295-315.
- [13] G. Ki, K.-T. Lee and D.-H. Kim, "Modeling and Evaluation of Network-based Partial Distributed Mobility Management", 2014 International Conference on Green and Human Information Technology, (2014) February 12-14.

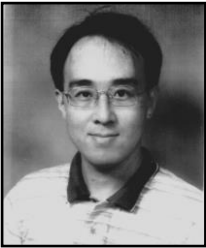
Authors



Jang-Geun Ki is currently a professor in the Department of Electrical & Electronic & Control Engineering, Konju National University in KOREA. He received B.S., and M.S. degrees in Electronics from Korea University, KOREA, in 1986 and 1988, respectively. In 1992, he received Ph.D. degree of Electronics at Korea University. His main research interests include telecommunication protocols and mobile networks.



Kyu-Tae Lee received his B.S. in Electronics Engineering from Korea University, Korea, 1984 and M.S. degree in Electronics Engineering from Korea University, Korea, 1986 and Ph.D. degree in Computer and Electronics Engineering from Korea University in 1991. He joined the faculty of Kongju National University, Korea where he is currently a professor in Division of Information and Communication Engineering. His research interests include DSP programming, Reconfigurable open architecture, Software Defined Radio system and Circuit and System. He is a member of Webcasting, Internet TV and Telecommunication.



Do-Hyeun Kim received the B.S., M.S. and P.D degrees in Electronics Engineering from Kyungpook National University, Taegu, Korea, in 1988 and 1990, 2000 respectively. He joined the Agency of Defense Development (ADD), Korea, in 1990. Since 2004, he is currently a professor at the Department of Computer Engineering at Jeju National University, Korea. His research interests include sensor web, optimization algorithm and context prediction.

