Seamless Mobility Management in IP-based Wireless/Mobile Networks with Fast Handover

Byungjoo Park[1], Eunsang Hwang[2] and Gil-Cheol Park[1]
[1] School of Multimedia, Hannam University
Ojung-Dong, Daeduk-Gu, Daejeon 306-791, Korea
[e-mail: {bjpark, gcpark}@hnu.kr]
[2] LG Electronics Mobile Handset R&D Center Research
Gasan-Dong, Seoul, 153-023, Korea
[e-mail: likerich@lge.com]
*Corresponding author : Byungjoo Park

Received March 20, 2009; revised May 9, 2009; accepted May 20, 2009; published June 22, 2009

Abstract

The challenges of rapidly growing numbers of mobile nodes in IPv6-based networks are being faced by mobile computing researchers worldwide. Recently, IETF has standardized Mobile IPv6 and Fast Handover for Mobile IPv6 (FMIPv6) for supporting IPv6 mobility. Even though existing literatures have asserted that FMIPv6 generally improves MIPv6 in terms of handover speed, they did not carefully consider the details of the whole handover procedures. Therefore, in conventional protocols, the handover process reveals numerous problems manifested by a time-consuming network layer based movement detection and latency in configuring a new care of address with confirmation. In this article, we study the impact of the address configuration and confirmation procedure on the IP handover latency. To mitigate such effects, we propose a new scheme which can reduce the latency taken by the movement detection, address configuration and confirmation from the whole handover latency. Furthermore, a mathematical analysis is provided to show the benefits of our scheme. In the analysis, various parameters are used to compare our scheme with the current procedures, while our approach is focused on the reduction of handover latency. Finally, we demonstrate total handover scenarios for the proposed techniques and discussed the major factors which contribute to the handover latency.

Keywords: Movement detection, IPv6, mobile IPv6, fast handover, duplicate address detection, fast handover for mobile IPv6

This paper has been supported by the 2009 Hannam University Research Fund.

DOI: 10.3837/tiis.2009.03.004

1. Introduction

Mobile IPv6 is designed to manage the movement of mobile nodes (MNs) between wireless IPv6 networks [1][2]. The protocol provides seamless connectivity to MNs when they move from one wireless point of attachment to another in a different subnet. Mobile IPv6 notifies the correspondent(s) of an MN about its new location by binding the MN addresses. Nevertheless, the MN cannot receive IP packets on its new point of attachment until the handover finishes. With regards to mobility support in IPv6 (MIPv6) [1], an MN can determine its network layer movement by using router discovery and Neighbor Unreachability Detection [3]. After an MN makes a new care of address (NCoA), it must check its uniqueness by duplicate address detection (DAD).

The delay of network layer-based movement detection, non-optimized time sequencing of handover procedures and latency in configuring a new care of address are inevitable in Mobile IPv6. These delays will cause packet disruption and increase network load.

Several performance issues have been identified with handover latency in MIPv6. One of the performance issues is the duration which is taken to automatically configuration new care-of address (NCoA) and confirms its uniqueness. According to the current RFC 2462 [4] DAD algorithm, it takes at least 1000ms to detect that there is no duplicate address in the link. Obviously, DAD is a time consuming process, particularly when MN in need of seamless handover runs it. In IPv6 network, the probability of the address conflict is usually very low independent of whether they are obtained via stateless or stateful auto-configuration. In case a fixed node is just booted, there is no address conflict since the node can be re-configured with another new address and if a node moves between networks and tries to maintain its current session. However, there is a problem when address conflict occurs. The session between the node and its correspondent node may be broken and the rightful owner of the address may receive the misdirected packets.

Considering the desire to offer seamless communication services over IP-based wireless networks such as, MIPv6 which in turn requires the resolution of the handover latency issue for the MIPv6 networks. This article proposed a new enhanced fast handover scheme which remarkably reduces DAD processing time during handover procedure. This can be achieved through configuration optimization of new CoA, which will be used without any concern on address collision after an MN moves to a new link. In particular, we also focus on the delay optimization of movement detection and DAD for fast handover in Mobile IPv6 networks. In movement detection [3], router solicitation (RS) delays the transmission for a random amount of time (Random delay for RS, RD_RS). This time serves to alleviate congestion when many hosts start up on a link at the same time, which might happen after recovery from a power failure. Also, router advertisement (RA) must be delayed by a random amount time (Random delay for RA, RD_RA). This time is required to prevent multiple routers from transmitting at exactly the same time, and to prevent long-range periodic transmissions from synchronizing with each other. These random delays are the second largest delay after DAD in MIPv6 Network-Layer handovers.

This article is organized as follows. Section 2 surveys the previous works and problems in existing protocols. Section 3 Introduce the enhanced fast address configuration and confirmation scheme called ‘EFACC’. The performance evaluations and comparisons in MIPv6, FMIPv6 and EFACC scheme are shown in section 4. Section 5 and 6 provides the discussion and conclusions, respectively.

2. Related Works

MIPv6 handover procedure is consists of movement detection, new CoA configuration, DAD and binding update. To process Movement Detection, the MN detects that it was moved to a new subnet by analyzing the router advertisement periodically sent by the access router (AR) (see **Fig. 1**). The MN can also request to AR to send a router advertisement by sending a router solicitation. To initiate CoA configuration and DAD, the information contained in the router advertisement will allow the MN to create a new CoA. As specified in IPv6 [3], the MN needs to verify first the uniqueness of its link-local address on the new link. The MN performs DAD on its link-local address. Then, it may use either stateless or stateful address auto-configuration [4] to form its new CoA.

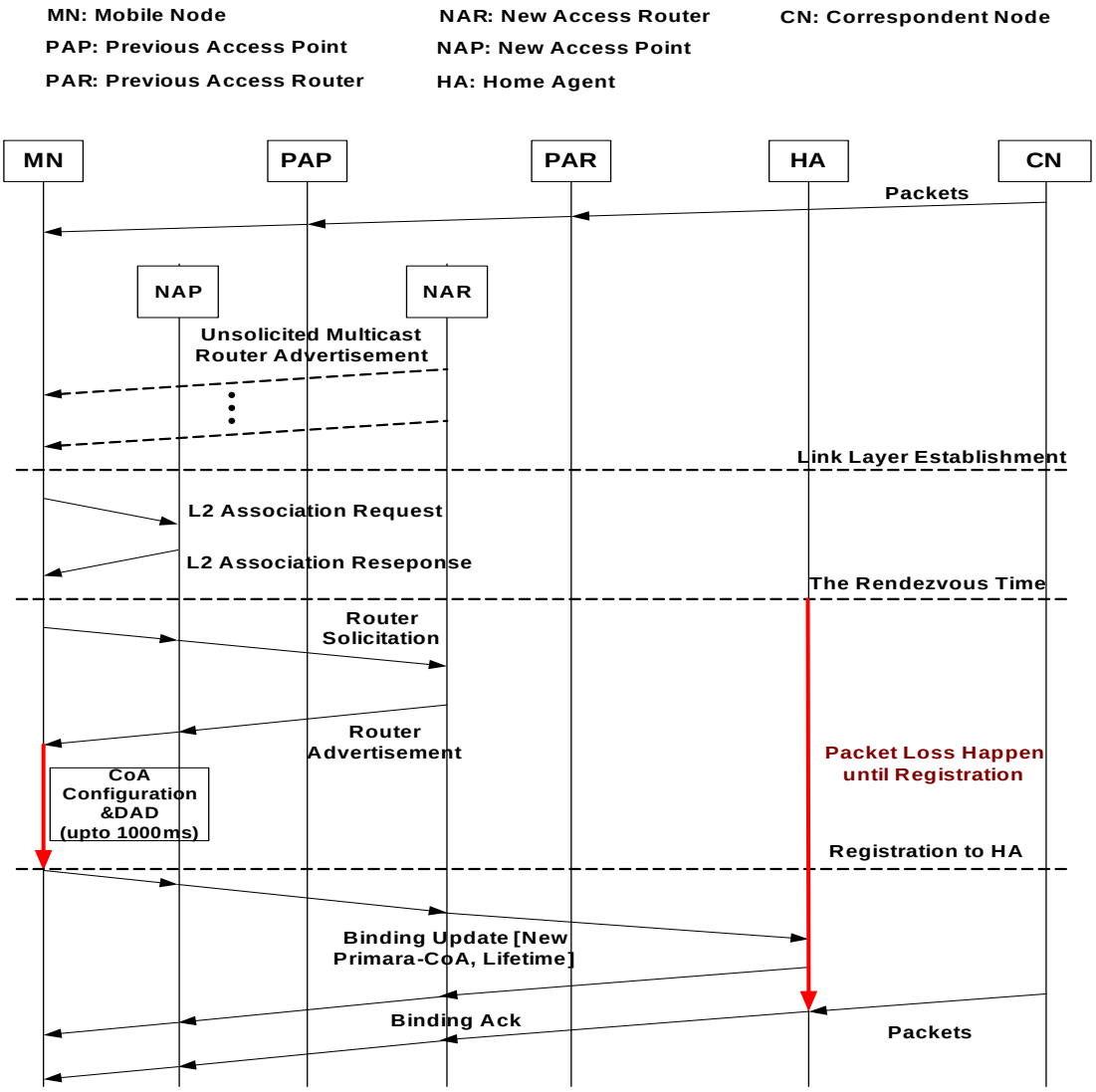


Fig. 1. The IETF MIPv6 handover procedure

2.1. Movement Detection Procedure in MIPv6

The primary aim of movement detection is to identify L3 handovers. In MIPv6, movement detection generally uses Neighbor Unreachability Detection to determine when the default router is no longer bi-directionally reachable, in which an MN must discover a new default router on a new link. However, this detection only occurs when the MN has packets to send, and in the absence of frequent router advertisements or indications from the link-layer, the MN might become unaware of an L3 handover. After a change of link layer connection the MN must detect any changes at the IP layer before it can signal the change to the network. MIPv6 uses RS and RA to detect changes of IP network prefix, this is part of the standard router discovery protocol. The protocol contains a built-in timer, these timers prevent a router from sending immediate responses to RS in order to prevent multiple nodes from transmitting at exactly the same time and to avoid long-range periodic transmissions from synchronizing with each other. These are the significant delays since they interfere with the MIPv6 movement detection algorithm thus preventing mobility signaling for up to 1000ms [1][4].

2.2. Duplicate Address Detection (DAD) in MIPv6

After completing the movement detection, an MN should generate an NCoA using IPv6 stateless address auto-configuration upon moving to the new link [3][4]. After generating the CoA an MN should perform DAD for testing the new CoA's uniqueness within the new link. The DAD must be performed on all new IPv6 addresses to eliminate address collisions. As the problems of DAD delay have become apparent, there are suggestions to skip DAD altogether or to perform DAD "asynchronously" if two nodes do configure the same address [5]. However, there are serious consequences when the colliding nodes will 'fight' for the address and corresponding nodes must choose one arbitrarily between the responses depending on their order of arrival. Not only will packets not be delivered to the correct node, but the colliding node may send negative acknowledgements such as TCP resets or ICMP 'Destination Unreachable' messages, causing existing connections to be terminated.

The current and simplest form of DAD was laid out as part of RFC 2462 [4]. When a node wishes to create a new address on an interface, it combines the network prefix with a suffix generated from its interface identifier. The interface identifier can be either obtained from the interface hardware or generated randomly [6][7]. This address is referred to as the tentative address. The node sends a neighbor solicitation message from the unspecified address to the tentative address. If the address is already in use by another node, that node will reply with a defending Neighbor Advertisement (NA) message.

Once a node has sent the Neighbor Solicitation (NS) message, it waits for RetransTimer milliseconds to see if a defending Neighbor Advertisement (NA) message is forthcoming and the solicit-and-wait process is repeated the Duplicate Address Detection Transmits times. By default the process is only done once and the default value of RetransTimer is 1000ms, resulting in a delay of 1000ms. During this time, the node cannot communicate with another node. Obviously, RFC 2462 DAD is a time consuming process, particularly when an MN in need of seamless handover runs it.

2.3. Fast Handover for Mobile IPv6 (FMIPv6)

IETF Fast Handover for Mobile IPv6 protocol (FMIPv6) was proposed to complement the Mobile IPv6 (MIPv6) by reducing the handover latency for the real-time traffic [10][11]. The FMIPv6's primary goal is to eliminate the factors of delay introduced by the movement detection and the address configuration procedures of MIPv6. It enables MN to quickly detect the movement to a new subnet by providing the new access point (AP) identifier and receiving the associated subnet prefix information. MN formulates a prospective NCoA when still

present on current subnet. Furthermore, in order to make MN allocates NCoA to its interface immediately after attaching to new subnet, FMIPv6 allows the NCoA confirmation procedure to be executed before or while MN switches its subnet. **Fig. 2** shows FMIPv6 handover procedure.

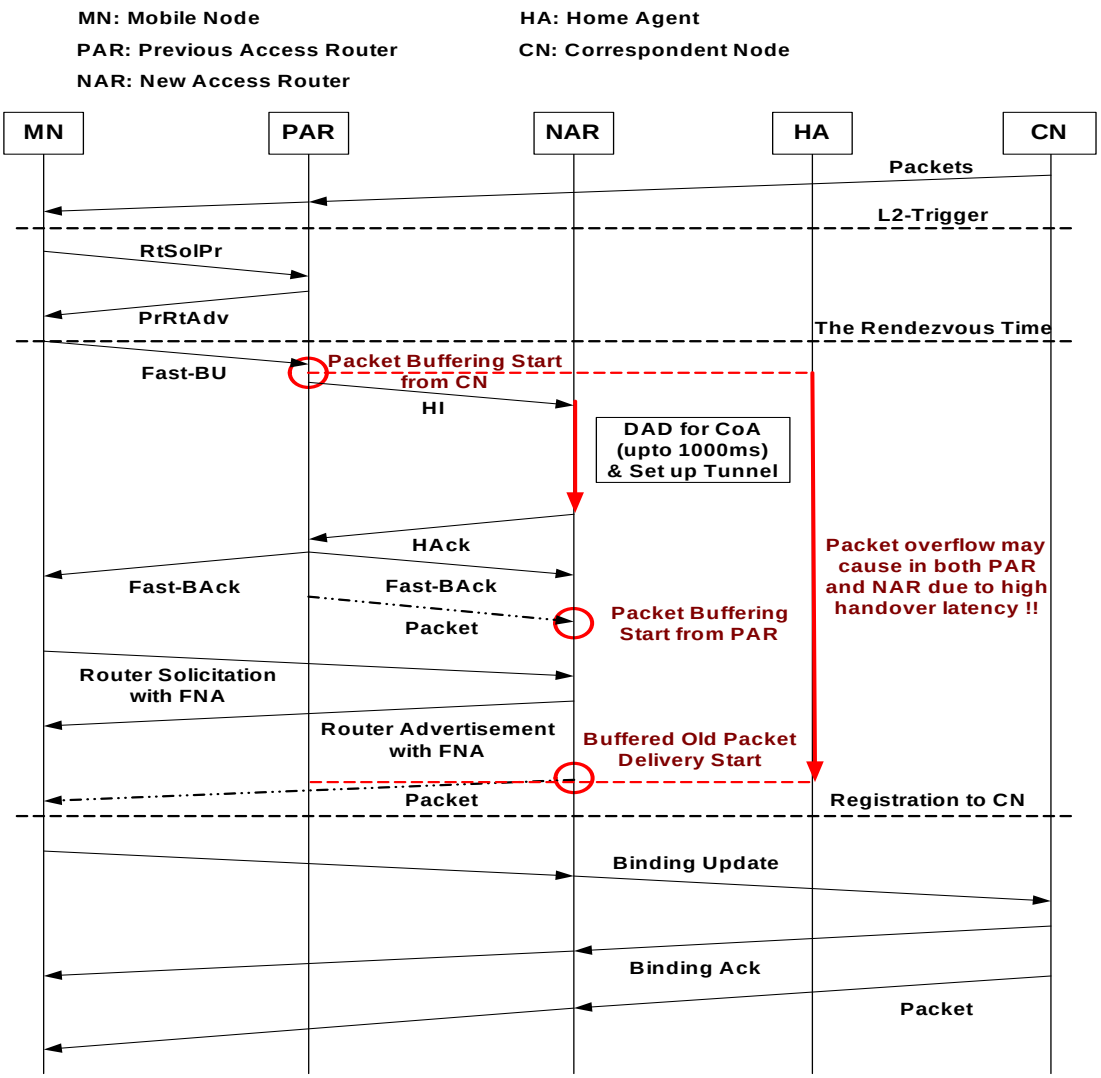


Fig. 2. The IETF FMIPv6 handover procedure

The scenario in which an MN receives the positive result about the confirmation of its prospective NCoA on the current subnet is called predictive mode, in which an MN checks the uniqueness of NCoA after MN attaches to new subnet is called reactive mode. Although MN initiates the NCoA confirmation at early time on the current subnet, FMIPv6 could fall into reactive mode unless MN could receive the confirmation result on the current subnet.

In addition, if the proposed NCoA is rejected during the NCoA confirmation procedure, MN may configure NCoA by itself after movement so that handover latency becomes long. In this case, NAR can suggest a unique NCoA, but there is no specific method to do it. In order to

achieve more reduction of handover latency, it is required that predictive mode should occur more frequently than reactive mode. So, it is very demanded that the NCoA confirmation should be done promptly and its result should be always successful. However, a proper confirmation method has not been provided.

2.4. Optimistic DAD

Optimistic DAD (oDAD) [8] provides an approach for eliminating DAD completion time. oDAD was proposed based on the premise that DAD is far more likely to succeed than failed. An optimistic MN modifies its IPv6 neighbor discovery protocol [3] and IPv6 address auto-configuration protocol [4] while keeping backward interoperability. For a node to send or receive packets after it moves to a new subnet, it must participate in neighbor discovery protocol. However, sending NSs and NAs with tentative address has risks in stealing service from an existing node in the case of collision. To avoid this, oDAD exploits an existing 'override' flag in NAs, indicating that this NA takes precedence over earlier ones. NAs sent with tentative address, however, it is sent with the 'override' flag cleared; indicating that a neighbor node has already a neighbor cache entry for that address and the NAs should be ignored.

Although this optimistic approach reduces handover latency in non-collision state, the address collision occurs after sending NAs where its 'override' flag is cleared, and it can incur some penalty to both optimistic node and rightful owner of the address. In the collision case, if an optimistic node already sent packets (e.g., BU message, etc.) which identify it as the owner of its tentative address, the connection between the node and its correspondent node may be broken and the rightful owner of the address may receive the misdirected packets. The authors in [9] demonstrated exiting methods to handle the address conflicts in IPv6 network, and discuss their strengths and weakness. It introduces some alternatives of RFC 2462 DAD. Nevertheless, it is noted that they did not say that oDAD is the complete and unique solution. This method does not assure the uniqueness of the address, and when the collisions will occur.

2.5. Advanced DAD

Advanced DAD (aDAD) [10] is based in an address pool maintained by the access router to provide an address to a mobile node that is known to be unique on the link. DAD's performance is the same as the oDAD's in terms of latency. Two schemes completely eliminate the DAD latency from the layer-3 handover latency. However, in terms of the method to handle address conflict, the two schemes are completely different: aDAD prevents it from happening, while oDAD recovers it after it actually occurs.

To provide the addresses for address pool, the access router randomly generates global routable addresses as background process and must perform standard DAD on the address according to RFC 2462. This procedure may increase an additional load to the access router that configures sufficient addresses in advance.

3. New Proposal for Fast Handover in IP-based Wireless/Mobile Networks

The goal of our proposal is to allow a Mobile Node (MN) to obtain a unique NCoA before or while it establishes connectivity with a new access router (NAR). We focus on the delays optimization of address configuration and confirmation to quickly obtain NCoA for fast handover during the connectivity establishment with a NAR or before it. In this section, we

describe our new enhanced fast address configuration and confirmation scheme called 'EFACC' to reduce handover latency. We define the handover procedure in different ways such as movement detection, NCoA configuration, DAD and binding update. To support our new scheme EFACC, we use new movement detection, the NCoA configuration and DAD scheme with look up algorithm in MIPv6. **Fig. 3** shows the EFACC handover procedure.

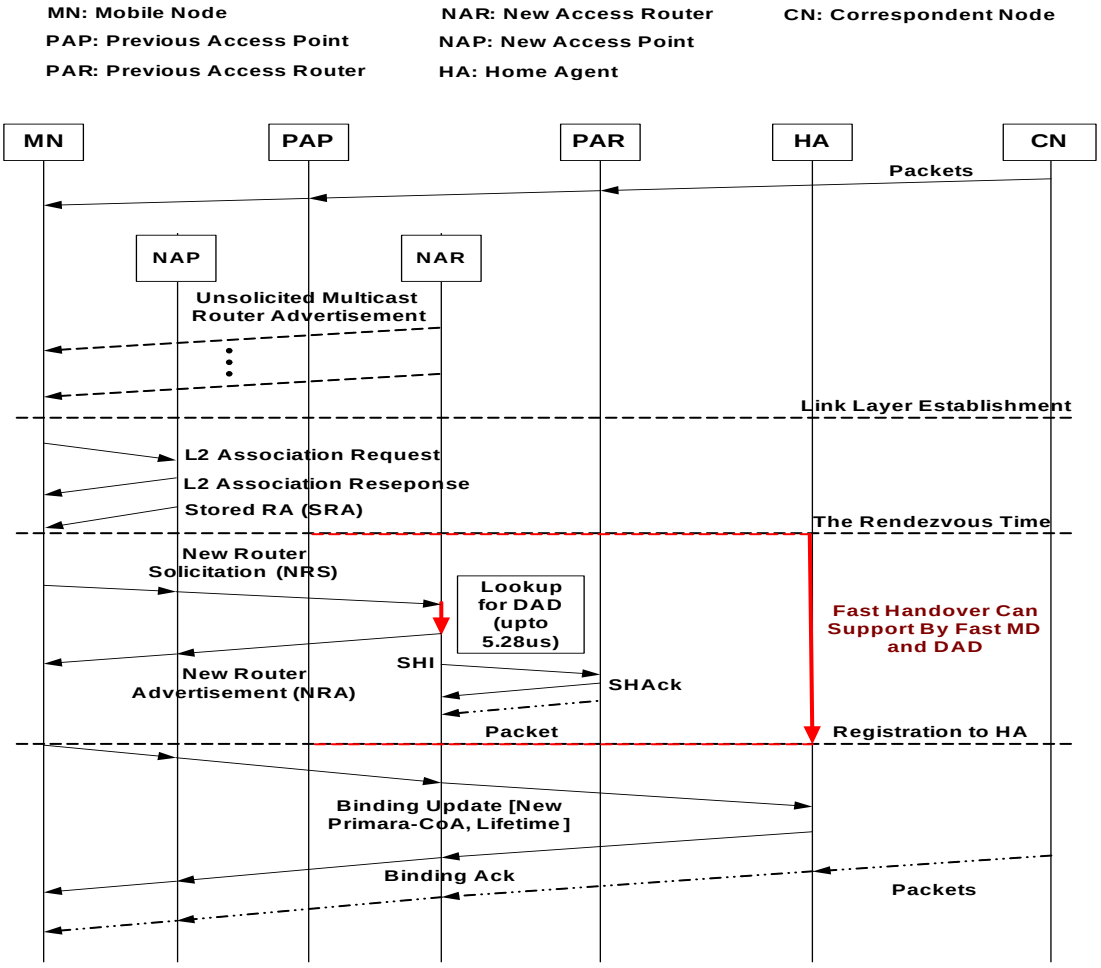


Fig. 3. The proposed EFACC handover procedure

3.1 Fast Movement Detection

Our proposed Movement detection scheme is based on EAP scheme [12]. In movement detection, the MN is aware of performing handover to another base station because of the channel maintenance problem. The MN achieves a scan to observe APs through probes, and the result of the scan is a list of AP's information. When the authentication is completed, then the MN sends the association request message with its MAC address. The new AP grants association by sending the Association Response Message. After association is made the NAP sends the stored RA to an MN in Association Response message.

In our new fast movement detection procedure, when the MN receives the stored RA (SRA) message from a NAP during L2 handover, the MN compares the prefix of the RA message with existing prefixes in the cache by using SRA message. The MN can receive NAR's Prefix

information which can support fast movement detection. If the prefix is different, the MN is able to generate a link local address using the stateless address auto-configuration scheme and the prefix option allowed in stored RA messages. As soon as it finished the NCoA configuration, the MN sends a new RS message by unicast signaling to the new AR using the received stored RA's source address included in the Association Response Message.

The unicast signaling method is very significant because it can reduce the network load and avoid the random delays attributed to RS and RA. We also add a mechanism to NRS messages in order to process the new DAD method. A MN can help smooth handover by adding the CoA used by MN to the NRS as an option to provide interoperability with normal nodes. This takes place by using a 1-bit D-flag in reserved field and notifying the node that follows the scheme that we offer. We name this bit as the "NCoA DAD Request bit (D bit)" and the two options the "Previous MN's CoA" and "Previous AR's global address".

3.2 New Care-of address Configuration and Confirmation

To reduce DAD processing delay, we propose a new NCoA configuration and DAD scheme using an authoritative address cache. This authoritative address cache supports a new enhanced lookup algorithm which can reduce DAD processing delay from 1000 ms to a very short time. That is, regular DAD in standard MIPv6 needs to 1000ms to check the uniqueness of the new NCoA. However, new scheme can support fast DAD delay by using new lookup algorithm called Patricia Trie searching method. Also, the use of neighbor cache for DAD gives an additional advantage of obtaining alternative addresses because addresses are managed in the neighbor cache. In our article, we use authoritative address cache which similarly acts as neighbor cache to compare MN's states.

Firstly, the MN allocates NCoA in its interface without DAD by NRS. We need to consider how an authoritative address cache and the Neighbor Unreachability Detection procedure perform DAD. As stated in [3] [13], an authoritative address cache contains one entry for each neighbor to which the node has recently sent messages. That is, an authoritative address cache similarly acts as neighbor cache in access router. Each entry of an authoritative address cache may be generated by the NRS, the Neighbor Solicitation or the Unsolicited Neighbor Advertisement in the case of router. The entries generated in this way are maintained in the state of stale until traffic is sent to the neighbor. Since the authoritative address cache has the list of all hosts of the link, which the AR manages, we can compare the entry of the authoritative address cache with an MN's MAC address, which is included in each NRS message transmitted from MN to AR. If the addresses of all nodes are generated on the link by stateless auto-configuration, we do not have to consider DAD. However, if the link local address of a node is generated by some exceptional process the ARs cannot maintain the entry of the node in the authoritative address cache. **Fig. 4** shows the architecture of modified authoritative address cache.

We propose that ARs should be able to receive solicited multicast NS messages for normal DAD of exceptional nodes. Since the solicited NS message is sent by multicast, the ARs can receive this message by modification of its interface. The lookup procedure in the authoritative address cache on NAR is the DAD procedure. The proposed DAD using lookup procedure consumes an extremely short amount of time, typically a few micro second units such as longest prefix matching speeds in routing table. If the new DAD procedure is completed by lookup algorithm in NAR, the NAR can unicast the new RA (NRA) message to the MN's link local address of the destination address. At the same time the NAR immediately sends smooth handover initiation (SHI) message with an NCoA to the previous access router (PAR) to setup the tunnel between the NAR and PAR. When the PAR receives SHI message, the PAR sets up

a host route for the MN's NCoA and responds with a smooth handover acknowledge (SHAck) message. The RA message can also be made like the RS message by adding a 2-bit L-flag to the reserved flag and including the "New MAC address", "New link-local address" and "NCoA DAD Reply" as an options in the option field in case of address duplication. When an MN receives NRA, the MN has to be operated by L-flag.

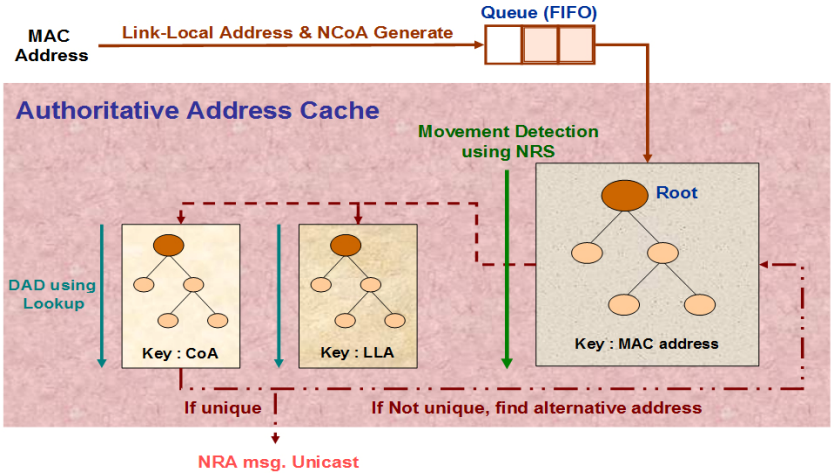


Fig. 4. The authoritative address cache architecture

The L-flag in NRA is defined by four states: {(0, 0), (0, 1), (1, 0), (1, 1)}. Explanations of L-flag in the NRA are as follows. The state (0, 0) denotes that an MN must change MAC address (This is not applicable to IEEE 802 case). In the state (0, 1), an MN can allocate a link-local address and an NCoA. The state (1, 0) means that an MN has to change the link-local address allocated into the alternative address. The state (1, 1) means that an MN can not use NCoA.

3.3. Lookup Procedure for Fast NCoA Confirmation

We denote t_{LD} as the address lookup delay, which is the time required to check an MN' MAC address for movement detection and DAD in the Patricia Trie search. The address lookup delay is accordingly given as:

$$t_{LD} = t_{DAC} \times N \tag{1}$$

Where t_{DAC} is the delay for access and the comparison operations in RAM and N is the number of lookups in the Patricia Trie. This Patricia Trie has the worst performance in line per minute. We use this algorithm in order to show the lookup time of the worst performance. Under the present circumstance, since a memory access requires from 60 to 100 nsec [14] and a comparison requires 10nsec in DRAM [15], we can use the value of t_{DAC} as 70 and 110nsec. In Patricia Trie case, the lookups require accessing memory of 48 times in the worst case, the N value is 48. Hence, t_{LD} is 3.36 μsec and 5.28 μsec and the calculated lookup delay is very small.

The traffic intensity φ is the quantity that governs the stability of the system. Let us introduce LD as the lookup delay, which is defined as the time duration from when an NRS

packet arrives at the AR to when an NRA message is forwarded to the output link. By applying the M/G/1 queuing model, the mean lookup processing delay is derived by

$$E[LD] = E[W] + E[t_{LD}] \tag{2}$$

Where $E[W]$ is the expected mean waiting time of a packet in the queue. Using the Pollaczek-Khinchin (P-K) formula, the mean waiting time is derived by

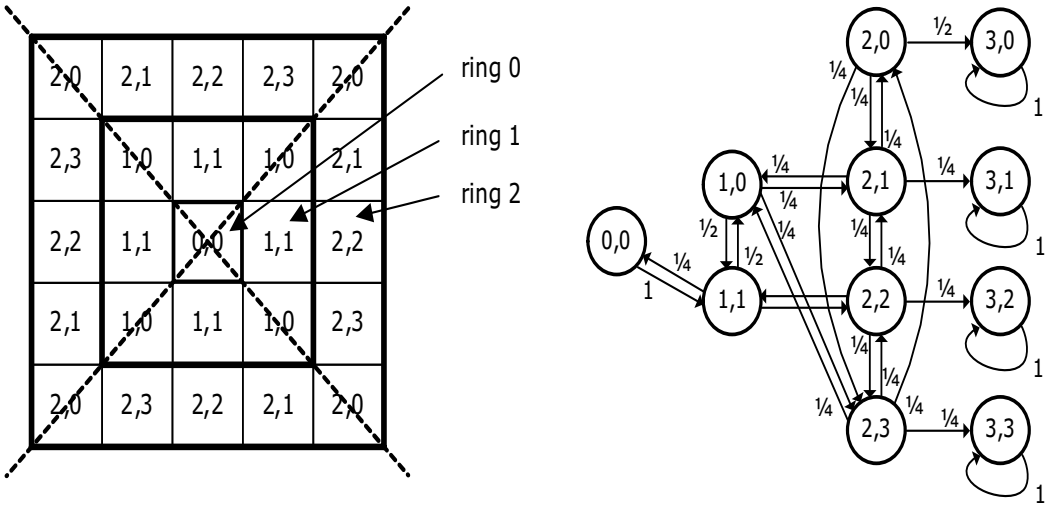
$$E[W] = \frac{\lambda_R \cdot E[t_{LD}]^2}{2(1-\varphi)} = E[t_{LD}]\left(\frac{\varphi(1+C_B{}^2)}{2(1-\varphi)}\right) \tag{3}$$

Where $C_B{}^2$ denotes the squared coefficient of variation of the processing time. An important observation is that, clearly, the mean waiting time only depends upon the first two moments of the lookup processing time. For exponential processing times we have $C_B{}^2 = 1$. So, in this case the expression for the mean waiting time measures simplifies to

$$E[W] = E[t_{LD}]\left(\frac{\varphi}{1-\varphi}\right) \tag{4}$$

4. Performance Evaluation and Comparisons

In this section, we will analyze the handover latency per movement for each protocol. Handover latency is defined for a receiving MN as the time elapses between the disconnection with the previous attachment of point and the arrival of the first packet after the MN moves to NAR.



(a) 3-layer subnet structure

(b) State diagram of two-dimensional random walk model in 3-layer subnet structure

Fig. 5. 3-Layer Subnet Architecture

4.1. Packet Level Traffic Model

We apply a simple model for the data packet traffic, even though there is similar nature that has been noticed. Our packet traffic model consists of two layers namely: session and packet. During a session, several packets are generated by a CN at an arbitrary rate and they reach the MN at a rate. Our traffic behavior can be modeled as follows:

- Session duration (t_0): It follows the exponential distribution with mean $E[t_0] = 1/\lambda_0$.
- Packet generation and arrival: It is done by an arbitrary rate.

We will evaluate the handover latency per movement for each protocol. The Handover latency is defined for a receiving MN as the time that elapses between the disconnection with the previous attachment of point and the arrival of the first packet after the MN moves to NAR.

4.2. Network System and Mobility Model

We assume that the homogeneous network which all wireless AP areas have the same shape and size in a subnet domain. First, we define some parameters that we will use for performance analysis. Let t_s and t_p be i.i.d. random variables representing the subnet domain residence time and the AP area residence time, respectively. Let $f_s(t)$ and $f_p(t)$ be the density function of t_s and t_p, respectively. In our paper, we intend the MN visit k AP areas in a subnet domain for a period t_s^k during t_s^k the MN resides at AP area i for a period t_i . Then, $t_s^k = t_1 + t_2 + t_3 + \cdots + t_{k-1} + t_k$ has the following density function

$$f_s^{(k)}(t) = \int_{t_1=0}^{t} \int_{t_2=0}^{t-t_1} \cdots \int_{t_{k-1}=0}^{t-t_1-\cdots-t_{k-2}} f_p(t_1) f_p(t_2) f_p(t_3) \cdots f_p(t_{k-1}) f_p(t-t_1-\cdots-t_{k-1}) dt_{k-1} \cdots dt_2 dt_1. \quad (5)$$

Using the Laplace transform convolution, we can determine the Laplace transform for $f_s^{(k)}(t)$ as follows [16]:

$$f_s^{(k)*}(s) = [f_p^*(s)]^k \quad (6)$$

Where $f_p^*(s)$ is the Laplace transform of $f_p(t)$.

We describe a two-dimensional random walk model for mesh planes in order to compute the subnet domain residence time density function. Our model is similar to reference [12], however, we consider a regular AP area/subnet domain overlay structure. We assume that an MN resides in an access point (AP) area for a period and moves to one of its four neighbors with the same probability, i.e. with probability 1/4. A subnet is referred to as a n-layer subnet domain if it overlays with $N = 4n^2 - 4n + 1$ AP areas.

Fig. 5 shows the 3-layer subnet domain in which each of the 25 small squares and the entire square represents each of the AP areas and one subnet domain area, respectively. The layer 0 AP area is called the AP area at the center of the subnet. The AP areas that surround layer x-1 AP areas are called layer x AP areas.There are 8x AP areas in layer x and exactly one AP area which is in layer 0. The n-layer subnet overlays AP areas from layer 0 to layer n-1. Particularly the AP areas that surround the layer n-1 AP areas are referred to as boundary neighbors, which

are outside of the subnet. According to the equal moving probability assumption, we classify the AP areas in a subnet domain into several AP area types. An AP area type is of the form <x, y>, where x indicates that the AP area is in layer x} and y represents the y+1st type in layer x. AP areas of the same type have the same traffic flow pattern because they are at the symmetrical positions on the mesh domain. For example, in **Fig. 5 (a)**, the AP type <1, 1>, <2, 1> represent that this AP is in ring 1 and ring 2 and it is the AP of 2nd type in ring 1 and ring 2, respectively.

In the random walk model, a state (x, y) represents that the MN is in one of the AP areas of type <x, y>. The absorbing state (n, j) represents that an MN moves out of the subnet from state (n-1, j), where $0 \le j \le 2n-3$. The state diagram of the random walk for 3-layer subnet is shown in **Fig. 5 (b)**. We assume that the AP area residence time of an MN has a Gamma distribution with mean $1/\lambda_p (= E[t_p])$ and variance υ. The Gamma distribution is selected for its flexibility and generality. The Laplace transform of a Gamma distribution is

$$f_p^*(s) = \left(\frac{\gamma\lambda_p}{s+\gamma\lambda_p} \right)^{\gamma}, \ where \ \gamma = 1/(\upsilon\lambda_p^2) \tag{7}$$

For a MN, in the end, the probabilities $\prod_p(i)$ and $\prod_s(j)$ that the MN moves across i AP areas and j subnets during session duration, can be derived as follows [16]:

$$\prod\nolimits_p(i) = \left\{ \begin{array}{ll} 1 - \frac{E[t_0]}{E[t_p]}(1 - f_p^*(\frac{1}{E[t_0]})) & , i = 0 \\ \frac{E[t_0]}{E[t_p]}(1 - f_p^*(\frac{1}{E[t_0]}))^2 (f_p^*(\frac{1}{E[t_0]})^{i-1}) & , i > 0 \end{array} \right\} \tag{8}$$

$$\prod\nolimits_s(j) = \left\{ \begin{array}{ll} 1 - \frac{E[t_0]}{E[t_s]}(1 - f_s^*(\frac{1}{E[t_0]})) & , j = 0 \\ \frac{E[t_0]}{E[t_s]}(1 - f_s^*(\frac{1}{E[t_0]}))^2 (f_s^*(\frac{1}{E[t_0]})^{j-1}) & , j > 0 \end{array} \right\} \tag{9}$$

Accordingly, between the two packet arrivals, an MN moves across the following average numbers of AP areas and subnets, respectively:

$$m_p = \sum_{i=0}^{\infty} i \prod\nolimits_p(i) \tag{10}$$

$$m_s = \sum_{j=0}^{\infty} j \prod\nolimits_s(j) \tag{11}$$

4.3. Network System and Mobility Model

At first, we introduce the distance parameters used for handover latency functions. (See **Fig. 6**)

a: #hops between HA and Border Gateway
b: #hops between CN and Border Gateway
c: #hops between Border Gateway and AR
d: #hops between AR and AP;

e: #hops between two ARs;

t_{WD} is the wireless component of the delay for a new AP re-association and authentication latency (MN's switching delay between APs). Let us assume that η is the packet delivery delay in wireless path between AP and MN, and ε is the packet delivery delay per one hop in wired path. t_{SRA} is the wireless delay for SRA latency between the MN and the new AP. In this article, the SRA message and L2 information are triggered together with an association response message. $t_{NRS/RS}$ and $t_{NRS/RA}$ are the transmission delays for the new RS/RA messages in EFACC and regular RS/RA messages in standard MIPv6, respectively $(t_{NRS/RS} + t_{NRA/RA} = 2t_{NR/R})$. ${}^{*}t_{RD}$ is the random delay for RS, RA defined as the RFC 3775 $({}^{*}t_{RD} = t_{RD\_RS} + t_{RD\_RA})$.

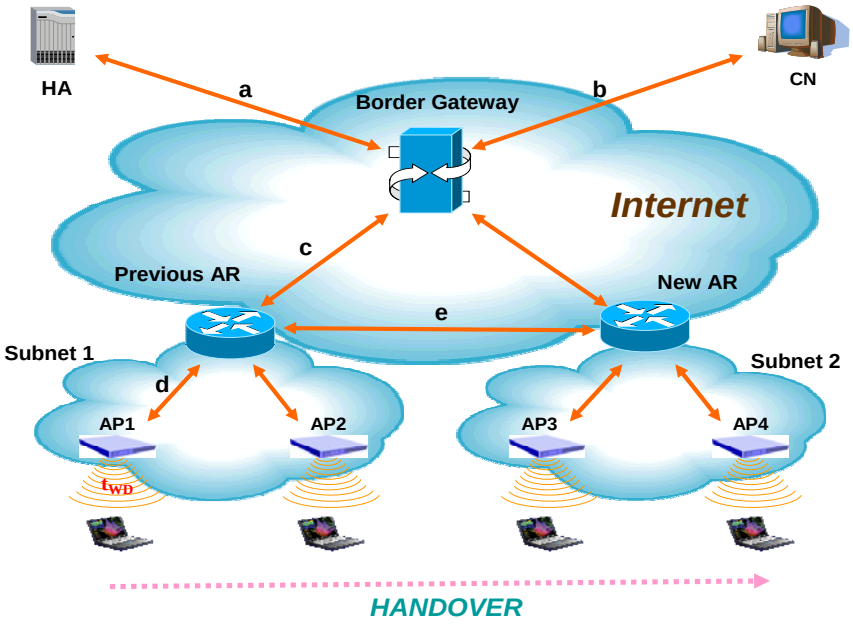


Fig. 6. Network Architecture

t_{BU} and t_{BAck} are the transmission delays for BU/Back messages respectively $(t_{BU} + t_{BAck} = 2t_B)$. t_{packet} is the packet transmission delay from CN to MN. t_{DAD} is the DAD processing delay defined as the RFC 2462. t_{LD} is the lookup delay for DAD in EFACC. ζ is the weighting factor of packet tunneling. ψ is the total delay between the time to exchange FBU/FBAck and the time of disconnection (Link-Down) with the current AP. t_{RS_FNA} and t_{RA_NAAck} are the transmission delays for RS with Fast Neighbor Advertisement and RA with Neighbor Advertisement Acknowledgment. $t_{SHI/HI}$ and $t_{SHAck/HAck}$ are the transmission delays for the new smooth HI/Hack messages in the EFACC and regular HI/HAck messages in the standard FMIPv6, respectively to setup the tunnel between PAR and NAR $(t_{SHI} + t_{SHAck} = 2t_{SH})$. t_{Packet_PN} is

the buffered packets forwarding delay from PAR to NAR. t_{Packet_MN} is the packet forwarding delay between MN and NAR. In FMIPv6, MN sends FBU to PAR prior to disconnection with PAR. At this time, the handover procedure of FMIPv6 is divided into two independent procedures: H_I the procedure to be executed by MN itself with PAR and NAR, and H_{II} is the procedure to be executed by only both PAR and NAR to establish the bidirectional tunnel. The two separated procedures will combine into one when NAR receives FNA from MN after MN's subnet movement. We assume that NAR has already received at least HI from PAR, when it receives FNA from MN. Before the two procedures H_I and H_{II} combine into one, the completion times of each procedure are defined as follows:

$$T_{H_I} = \psi + t_{WD} + t_{RS\_FNA} + t_{RA\_NAACk} \tag{12}$$

$$T_{H_I I} = (t_{HI} + t_{HAck} + \zeta \cdot t_{Packet\_PN}) + t_{DAD} \tag{13}$$

If H_{II} finishes before the completion of H_I (that is, $T_{H_I} > T_{H_{II}}$), NAR has buffered the packets tunneled from PAR and forwards them to MN when it receives FNA. If not, NAR waits for the packets which will be tunneled from PAR when it receives FNA. At the latter case, NAR have to wait for the completion of the address confirmation procedure. After announcing its attachment to NAR and receiving the tunneled packets, MN sends binding update messages with its new CoA to HA, and to CNs consecutively. In FMIPv6, the average handover latency per session time is defined in Eq.13. In our paper, the SRA message and L2 information are triggered together with an association response message. We assume that $t_{NRS/RS}$, $t_{NRA/RA}$, t_{RS_FNA} and t_{RA_NAAck} have the same value in transmission time. Also, t_{BU} and t_{BAAck} have the same value in transmission time. In our proposed scheme, t_{LD} is the most important factor determining the performance, using such parameters for the standard MIPv6, FMIPv6 and EFACC. The average handover latency per session duration is defined as follows:

$$\begin{aligned} T_{MIPv6} &= \frac{(m_p - m_s)t_{WD} + m_s(t_{WD} + 2t_R + {}^{*}t_{RD} + t_{DAD} + 2t_B + t_{Packet})}{m_p} \\ &= t_{WD} + \frac{m_s(2t_R + {}^{*}t_{RD} + t_{DAD} + 2t_B + t_{Packet})}{m_p} \end{aligned} \tag{14}$$

$$\begin{aligned} T_{FMIPv6} &= \frac{(m_p - m_s)t_{WD} + m_s(t_{WD} + MAX\{T_{H_I}, T_{H_{II}}\} + \zeta \cdot t_{Packet\_MN} - t_{WD} - \psi)}{m_p} \\ &= t_{WD} + \frac{m_s(MAX\{T_{H_I}, T_{H_{II}}\} + \zeta \cdot t_{Packet\_MN} - t_{WD} - \psi)}{m_p} \end{aligned} \tag{15}$$

$$\begin{aligned} T_{EFACC} &= \frac{(m_p - m_s)t_{WD} + m_s(t_{WD} + 2t_{NR} + t_{LD} + 2t_{SH} + 2t_B + \zeta \cdot t_{Packet})}{m_p} \\ &= t_{WD} + \frac{m_s(2t_{NR} + t_{LD} + 2t_{SH} + 2t_B + \zeta \cdot t_{Packet})}{m_p} \end{aligned} \tag{16}$$

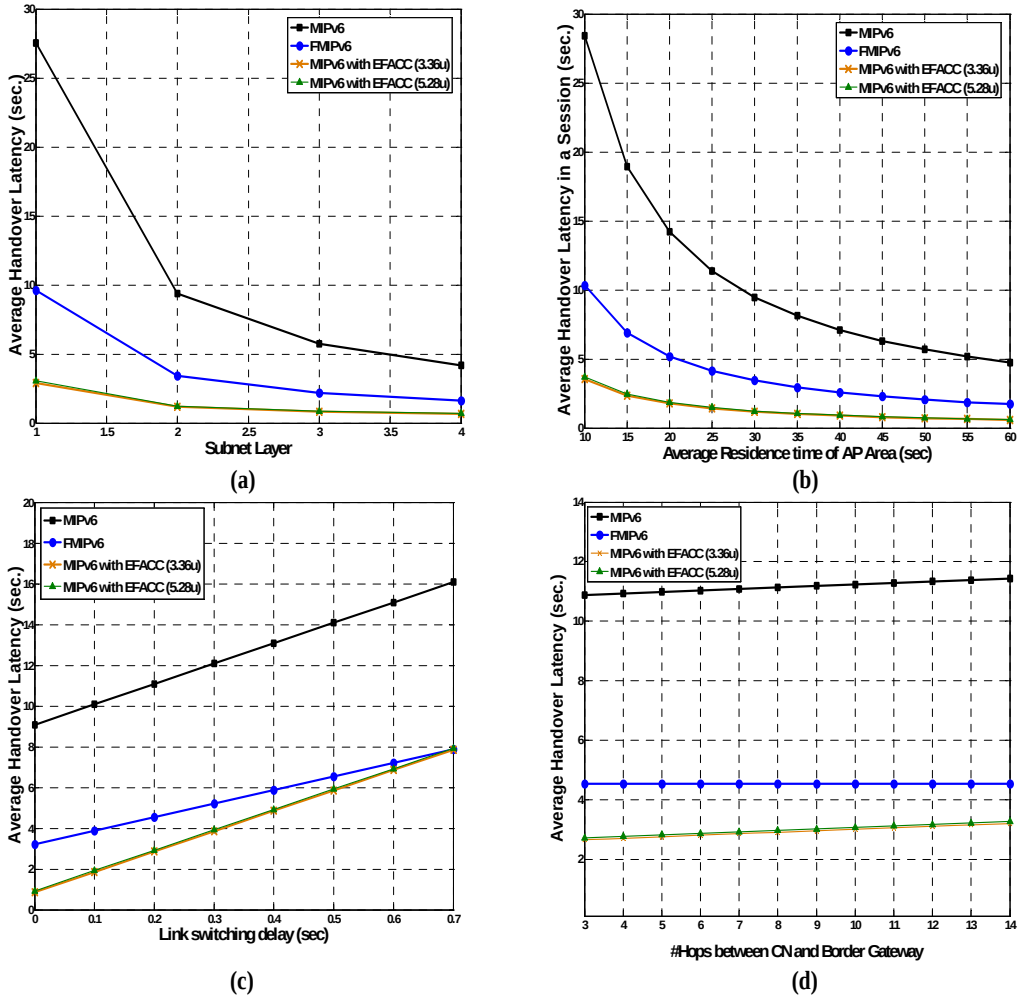


Fig. 7. Average Handover Latency Comparisons I

4.4. Numerical Results

For scrutiny, the following fixed parameters are used: $t_{RS/RA} = \eta + \varepsilon d = 0.015$, ${}^{*}t_{RD} = 0.15$, $t_{DAD} = 1$, $t_{BU/BAck} = \eta + \varepsilon(b + c + d) = 0.065$, $t_{HI/HAck} = \varepsilon e = 0.01$, $\zeta = 1.2$, $\lambda_0 = 0.0033$ (session duration is 300sec), $\varepsilon = 0.005$ (delay of one hop delivery is 5 milliseconds), $\upsilon = 1.0$ (variance of AP area residence time is 1 sec.), a=b=7, c=3, d=1, e=2 and $t_{Packet/Packet\_MN/Packet\_PN} = 0.065/0.015/0.01$. As the target of investigation, we select the following changeable parameters and their default values: $n = 2$ (subnet layer is 2), $\lambda_p = 0.033$ (mean of AP area residence time is 30 sec.), $t_{WD} = 0.03$ sec., and $\psi = 0.1$. As we select one parameter and change its value, the remaining parameters values are set to their default values during the following investigation.

Fig. 7 diversely depicts the average handover latency with respect to each changeable parameter. From the figures, we can know that MIPv6, FMIPv6 handover latencies are

substantially reduced when the existing address configuration and confirmation process is simplified by our proposed EFACC scheme.

Fig. 7 (a) shows the total handover latency of each protocol with respect to the subnet layer. It shows that the reduction of latency becomes high when a subnet contains many AP areas. The total handover latency depends mostly on Layer 2 link switching delay if there are many AP areas in a subnet domain. In contrast, when subnet layer is one, all movements will be always done between different subnets. Consequently, Handover will be always involved at the case. The figure show that proposed EFACC is under little influence of such system deployment, because it remarkably reduces the address confirmation delay, t_{LD} from the handover latency defined in Eq. 17. Since the session duration is set to 300sec., we can expect there will be a number of events that an MN can move to another AP area or subnet in the session duration. Total handover latency is defined as the sum of handover latencies caused by each movement in session duration. **Fig. 7 (b)** shows the average handover latency in session duration. As we seen from the figure, the handover process occupies much time within the whole session duration when MN moves across AP areas and subnets more frequently. Therefore, the less the handover process occupies time within a session duration when the less an MN moves across AP areas and subnets. **Fig.7 (c)** shows the relationship between the handover latency and the delay of link switching in session duration. When the switching delay t_{WD} becomes high, all protocols' handover latency becomes high too. We can also see that the standard FMIPv6's handover latency becomes almost equal to that of the proposed EFACC when the link switching delay is 0.7. Therefore, when the link switching delay is 0.8, the procedure H_1 becomes the dominant factor of handover latency. **Fig. 7 (d)** reveals the relationship between the handover latency and the distance from the MNs' access network to CN. When the distance is long, MIPv6's handover latency becomes a little high. But, in FMIPv6 the location registration by BU is not included in real handover process, and thus the distance does not affect the FMIPv6's handover latency. **Fig. 8** shows the average handover latency comparison with respect to the packet delivery delay in wireless path between AP/subnet and MN. We observed that the delivery delay affects all protocols' handover latency. We also found out that both MIPv6 and HMIPv6 are slightly influenced by FMIPv6.

In fact, in terms of signaling cost, FMIPv6 uses the wireless bandwidth more than MIPv6 because FMIPv6 defines four new messages which are delivered to or from MN.

5. Discussion

5.1. EFACC vs Optimistic DAD

The oDAD handles the address conflict in most cases. However, it is just a key that has an optimistic approach for address confliction where everything is assumed to go well. Essentially, the frequency of the address duplication is very low in IPv6 network. However, the optimistic scheme like oDAD becomes valid only when randomly auto-configured addresses are truly random and they are distributed uniformly and globally. In practice, generating true random numbers is tricky. We can think that two or more nodes use the same seed for random number generation. If the nodes meet in a link, the collision probability may be high. To think another example, let us assume that there is a series of links where a great quantity of IPv6 nodes exists and an optimistic node wanders about within the series of links. If the node generates its CoA randomly whenever moving to one of the series, the probability of address collision may be high too. Therefore, the oDAD cannot be the unique solution for DAD optimization when we take all situations into account, while we generally accept that it is

the one of optimized solutions when an MN uses the IEEE 802 identifiers to generate its interface identifier, or when it assures that the probability of collision is exceedingly low.

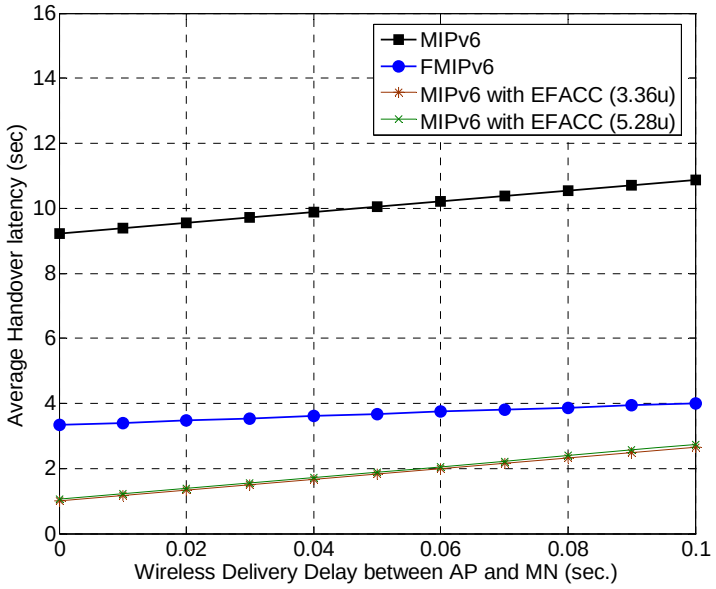


Fig. 8. Average Handover Latency Comparisons II

On the other hand, the proposed EFACC is a solution of pessimistic approach for IPv6 address configuration. Both EFACC and oDAD can remarkably reduce the DAD latency from the layer-3 handover latency. However, in terms of method how to handle the address conflict, these two schemes are definitely different. The EFACC prevents it from happening, while oDAD recovers it after it occurs. In this article, we consider that a node moves to a new network and tries to maintain its current session because the address conflict can be a big problem in such a node. We argue that EFACC would be better than oDAD in a mobile computing system where some possibility of address conflict exists.

5.2. Address Recycling

To support our EFACC protocol, AR's authoritative address cache checks the uniqueness of a new care-of-addresses and allocates it into authoritative address cache using new lookup procedure. As long as an MN resides in the same subnet, the allocated address will still utilized by it. When it goes out from the current subnet, all it should do is to throw away the address. This means that MN and AR don't have to concern about address recycling issue. Because of this, the AR can randomly create an alternative address in the authoritative address cache and the thrown address will sometime be made and allocated to a new MN.

5.3. Performance Impacts on AR

There are the performance impacts on AR due to the support of EFACC protocol. The main impacts are as follows. First, there is an actual CPU and IO load to check uniqueness of new care-of-address, and keep them unique in AR's authoritative address cache in accordance with the proposed EFACC protocol. This is unavoidable, but can be countered with additional CPUs and sufficient IO bandwidth. Furthermore, since a bunch of address will be stored into

authoritative address cache at the initial AR boot time, the impact will not be large in the course of AR's execution. Second, there are some impacts due to fetching an address from its authoritative address cache in response with MN's request. The impact largely depends on MNs' movement frequency. The impact will be large with a high rate of MNs' movement, while impact will be small with a relatively low movement frequency.

5. Conclusions

In this article, we presented the enhanced movement detection, address configuration and confirmation scheme which is called "EFACC". The use of authoritative address cache has advantages, such as a faster DAD checking speed which solves the shortcomings of normal DAD when a router has more than two links, and also obtain alternative addresses by managing it in the network. In the numerical analysis, we developed packet traffic, and system mobility models. By using various parameters, we analyzed the handover latencies of IETF IPv6 mobility protocols and the proposed EFACC. Based on the numerical results, we can conclude that the major benefits of our scheme are: to completely eliminate CoA configuration and confirmation latency involved in any seamless handover schemes, and to eliminate packet loss by avoiding the occurrence of address collision. The simulation results show that EFACC has better performance compared to the existing Mobile IPv6 protocols, and it works well in fast handover environments.

References

[1] D. B. Johnson, C. E. Perkins and J. Arkko, "Mobility Support in IPv6," *IETF* RFC 3775, 2004.
[2] S. J. Vaughan-Nichols, "Mobile IPv6 and the Future of Wireless Internet Access," *IEEE Computer*, Vol. 36, N. 2, pp.18-22, Feb 2003.
[3] T. Narten, E. Nordmark and W. Simpson, "Neighbor Discovery for IP version 6," *IETF* RFC 2461, 1998.
[4] S. Thomson and T. Narten, "IPv6 Stateless Address Autoconfiguration," *IETF* RFC 2462, 1998.
[5] N. Montavont, T. Noel, "Handover Management for Mobile Nodes in IPv6 Networks," *IEEE Communications Magazine*, Vol. 40, No. 8, pp38-43, Nov 2002.
[6] R. Hinden, S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture," *IETF* RFC 3513, 2003.
[7] T. Nartan, R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," *IETF* RFC 3041, 2001.
[8] N. Moore, "Optimistic Duplicate Address Detection," *IETF* RFC 4429, 2006.
[9] N. Moore, G. Daley, "Fast address configuration strategies for the mobile internet," in *Proc. of Australian Telecommunications, Networks and Applications Conference*, Dec 8-10, 2003.
[10] E. Rajeev Koodli, "Fast Handovers for Mobile IPv6," *IETF* RFC 4068, 2005.
[11] L. Dimopoulou, G. Leoleis, I. Venieris, "Fast Handover Support in a WLAN Environment: Challenges and Perspectives," *IEEE Network*, Vol. 19, No. 3, pp14-20, June 2005.
[12] B. J. Park, Y. H. Han, H. A. Latchmann, "EAP: New Fast Handover Scheme based on Enhanced Access Point in Mobile IP Networks," *International Journal of Computer Science and Network Security*, Vol. 6, No. 9B, pp.69-75, Sept 2006.
[13] B. Patil, F. Xia, B. Sarikaya, J. H. Choi, S. Madanapalli, "Transmission of IPv6 over 802.16's IPv6 convergence sublayer," *IETF* RFC 5121, 2008.
[14] V. Srinivasan, G. Varghese, "Fast address lookups using controlled prefix expansion," *ACM Transactions on Computer System*, Vol. 17, No. 1, pp.1-40, Feb 1999.
[15] R. Kawabe, S. Ata, M. Murata, "On performance prediction of address lookup algorithms of IP routers through simulation and analysis techniques," in *Proc. of IEEE International Conference on*

Communications, pp.2146-2151, April 28-May 2, 2002.
[16] I. F. Akyildiz, Y. B. Lin, W. R. Lai and R. J. Chen, "A New Random Walk Model for PCS Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 7, pp.1254-1260, July 2000.
[17] J.H. Jo, J.S. Cho, "Cross-Layer Optimized Vertical Handover Schemes between Mobile WiMax and 3G Networks," *KSII Transactions on Internet and Information systems*, Vol. 2, No. 4, pp.171-183, August 2008.

Byungjoo Park received the B.S. degree in electronics engineering from Yonsei University, Seoul, Rep. of Korea in 2002, and the M.S. and Ph.D. degrees (first-class honors) in electrical and computer engineering from University of Florida, Gainesville, USA, in 2004 and 2007, respectively. From June 1, 2007 to February 28, 2009, he was a senior researcher with the IP Network Research Department, KT Network Technology Laboratory, Rep. of Korea. Since March 2, 2009, he has been a Professor in the Department of Multimedia Engineering at Hannam University, Daejeon, Korea. He is a member of the IEEE, IEICE, IEEK, KICS, and KIISE. His primary research interests include theory and application of mobile computing, including protocol design and performance analysis in next generation wireless/mobile networks. He has published approximately 43 research papers on the theory and application of mobile computing, IPTV, Internet Application. Since 2004, his activities have focused on IPv6, IPv6 mobility, media independent handover, and cross-layer optimization for efficient mobility support on IEEE 802 wireless networks. He is an honor society member of Tau Beta Pi and Eta Kappa Nu.

Eunsang Hwang received the M.S. degree in the department of electronics engineering from Yonsei University, Seoul, Rep. of Korea in 2004. He is currently a senior researcher with the LG Electronics Mobile Handset R&D Center Research, Seoul, Rep. of Korea. His research interests include computer networks, routing management algorithms, Internet protocols, traffic control, QoS, performance analysis, and wireless Internet.

Gil-Cheol Park received B.E. and M.E. degrees from Hannam University and Soong-Sil University in 1983 and 1985 respectively, and received Ph.D. degree from Sungkyunkwan University in Rep. of Korea in 1998. From 1985 to 1990, he was a senior researcher in the SW research group of Samsung Advanced Institute of Technology. From 1996 to 1998, he was also an assistant professor, Computer Engineering, Hanseo University, Korea. Now he is a professor and Dean in Department of Multimedia Engineering at Hannam University, Daejeon, Korea. From 2005.3 to 2006.2 He has also conducted research as a visiting faculty member at the Tasmania state University, Australia. Since 2002, his activities have focused on multimdedia application, multimedia contents, IPv6 mobility, media independent handover in IEEE 802 wireless networks. He has published almost 100 technical Journal articles and refereed conference proceedings and given conference presentations in the areas of his research in multimedia application, communications and internetworking and innovative educational technologies. He is also the author of the books Multimedia Communications, published by SciTech and Multimedia for the Ubiquitous Age, published by SciTech. He has also served as Guest Editor for the International Journal of Multimedia and Ubiquitous.