Distributed Identifier-Locator Mapping Management in Mobile ILNP Networks

Moneeb Gohar 1,*, Jin-Ghoo Choi 2,*, Waleed Ahmed 1, Arif Ur Rahman 1, Muhammad Muzammal 1 and Seok-Joo Koh 4

1 Department of Computer Science, Bahria University, Islamabad 44000, Pakistan; badwanpk@gmail.com (A.U.R.); muzammal.muhammad@gmail.com (M.M.)
2 Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Korea
3 Department of Electrical Engineering, University of Poonch, Rawalkot 12350, Pakistan; drwaleedahmad@upr.edu.pk
4 Department of Computer Science and Engineering, Kyungpook National University, Daegu 41566, Korea; sjkoh@knu.ac.kr
* Correspondence: moneebgohar@gmail.com (M.G.); jchoi@yu.ac.kr (J.-G.C.)

Received: 30 October 2019; Accepted: 23 December 2019; Published: 31 December 2019

Abstract: In the Identifier Locator Network Protocol (ILNP) networks, the existing mobility control schemes based on the centralized entity, called the Dynamic Domain Name Service (DDNS) server, such that all the control traffic is processed at the DDNS server. However, the centralized mobility schemes have significant limitations, such as control traffic overhead at the server and large handover delay. In order to resolve these issues, we propose a new mobility control scheme for ILNP networks, which manages the identifier-locators (ID-LOCs) in the fully distributed manner. In our scheme, each domain has a dedicated mobile DDNS (m-DDNS) server at the site border router (SBR). The m-DDNS server maintains two databases; i.e., home host register (HHR) and visiting host register (VHR), to support the roaming of mobile hosts. When a mobile host roams into a domain, the m-DDNS server in the visiting domain registers the host’s ID-LOC in the VHR and requests the update of HHR to the m-DDNS server in the home domain. Since the m-DDNS servers communicate each other directly, the ID-LOC mappings are managed without involvement of any central entities. We analyzed our proposed mobility scheme via numerical analysis and compared its performance with those of existing schemes. Numerical results showed that our scheme outperforms the existing mobility control schemes substantially in terms of control traffic overhead at the servers, total transmission delay and handover delay.

Keywords: ILNP; mobility management; identifier-locator; mobile network; roaming

1. Introduction

With the advent of smart phones and various smart devices, the number of mobile users on the Internet is increasing explosively. This mobile trend has caused the rapid growth of the routing table size in routers [1,2]. To tackle the routing scalability issue, the Internet Engineering Task Force (IETF) has proposed the Identifier Locator Network Protocol (ILNP) protocol [3,4], which operates based on the address rewriting scheme. Under the ILNP, the identifier (ID) of a host is separated from its locator (LOC). Specifically, in the IPv6 address of 128 bits, the upper 64 bits are used for LOC and the lower 64 bits are used for ID. Then the Dynamic Domain Name System (DDNS) [5] server maps the ID onto the LOC for each mobile host.
For the ILNP-based networks, a mobility control scheme was proposed in [3,6–17], where individual mobile hosts conduct the binding and query operations with the DDNS server. We call the mobility scheme “ILNP-Global” in this paper, since it depends on the centralized DDNS server for ID-LOC mappings. Unfortunately, the centralized mobility control scheme has significant limitations in scalability and performance. Indeed, as the number of mobile hosts increases, the DDNS server suffers from huge control overhead. Moreover, all the centralized schemes are subject to large operational costs and service quality degradation due to the single point of failure problem.

An alternative mobility control scheme was proposed in [4] to enhance the ILNP-Global, where a local mapping table was employed in addition to the central DDNS server. We call this scheme “ILNP-Local” here. In the ILNP-Local, each domain maintains the local mapping table at its site border router (SBR). The SBR performs the ID-LOC mapping, where the locator is a local LOC (LLOC), for the hosts in the serving domain. On the other hand, when a host is not in the domain, SBR refers to the central DDNS server for the host’s locator, which is a global LOC (GLOC). This localized approach improves the scalability and performance over the centralized mobility control scheme. However, it still depends on the central DDNS server for inter-domain ID-LOC mappings.

In this paper, we propose a fully distributed ID-LOC management scheme for mobile ILNP networks, which is denoted by “ILNP-Distributed.” In the proposed scheme, we use both the GLOC and LLOC for a mobile host as in the ILNP-Local scheme [4]. We place the dedicated mobile DDNS (m-DDNS) server at each SBR to eliminate the centralized DDNS server. The m-DDNS server manages two databases, called Home Host Register (HHR) and Visiting Host Register (VHR), for ID-LOC mappings. When a host is in the home domain, its ID-LLOC is stored in the HHR of the home domain. On the other hand, when the host moves into another domain, its ID-LLOC is registered in the VHR of the visiting domain and the ID-GLOC is set in the HHR of the home domain. Here, it is the SBR of the visiting domain that requests the ID-GLOC update to the m-DDNS server of the home domain.

The rest of this paper is organized as follows. In Section 2, we review the existing mobility control schemes for ILNP networks. In Section 3, we describe our proposed mobility control scheme. We analyze the considered schemes in Section 4 and compare their performances numerically in Section 5. Finally, we conclude the paper in Section 6.

2. Materials and Methods

In this section, we review two previous mobility control schemes for ILNP networks, which are denoted “ILNP-Global” and “ILNP-Local” respectively.

2.1. ILNP-Global

We consider the mobility scenario shown in Figure 1 to discuss the operation of ILNP-Global. MN (bob.SKT.com) and CN (alice.SKT.com) are located in the same domain initially, but MN moves to another domain then. Further, the MN changes the serving access router (AR) by handover in the visiting domain.
Figure 1. Mobility scenario for Identifier Locator Network Protocol (ILNP)-Global scheme.

In Figure 2, we illustrate the ID-LOC management of the ILNP-Global scheme in cases of non-roaming, roaming and handover [3]. When MN attaches to AR2 in the home domain (non-roaming case), the MN registers its GLOC by sending a LOC binding update message to the centralized DDNS server (Step 1). The DDNS server registers the MN’s ID-GLOC mapping, and returns the LOC binding update ACK to the MN (Step 2). When CN has data packets for the MN, it needs the MN’s GLOC for packet delivery, so the CN sends a LOC query request to the DDNS server (Step 3), and obtains the MN’s GLOC from the associated LOC query response message (Step 4). We recall that the DDNS server keeps the ID-GLOC information of all the hosts in the network. The CN then transmits the packets directly to the MN since GLOC is the MN’s global locator in ILNP-Global, as shown in Figure 1.

When MN moves to another domain and attaches to AR3 (roaming case), the MN updates its GLOC stored in the DDNS server. That is, MN sends a LOC binding update message to the DDNS server (Step 5). The DDNS server updates the MN’s ID-GLOC in its database, and returns the LOC binding update ACK to the MN (Step 6). For data delivery, CN requests the MN’s GLOC to the DDNS server by sending a LOC query request (Step 7). The DDNS server notifies the MN’s GLOC by the LOC query response message (Step 8). The CN can send the data packets to the MN directly.

We now look over the handover of MN from AR3 to AR4. Once MN attaches to the target router AR4, the MN reports its GLOC to the DDNS server. Specifically, MN sends a LOC binding update message with its GLOC to the DDNS server (Step 9). The DDNS server updates the ID-GLOC mapping and returns the LOC binding update ACK to the MN (Step 10). Then, the MN transmits a LOC Update message to CN also, to notify its new GLOC. The CN updates the MN’s GLOC, and replies to the MN by the LOC Update ACK (Step 11 and 12). CN can send the data packets to the MN directly with the updated GLOC.
2.2. ILNP-Local

The ILNP-Global scheme depends on the centralized DDNS server to manage the ID-LOC mappings, which incurs significant control message overheads at the server in the global scale. To deal with this problem, the ILNP-Local scheme was proposed in [4]. Differently from ILNP-Global, the ILNP-Local employs a local mapping table (LMT) at the SBR, in conjunction with the DDNS server, to provide the localized mobility control.

For ILNP-Local, we consider the same mobility scenario with that in the ILNP-Global, as shown in Figure 3. MN and CN are located in the same domain but MN moves to another domain by roaming. After some time, the MN makes a handover to another AR in the visiting domain.

In Figure 4, we illustrate the ID-LOC management operations of the ILNP-Local. We first consider the non-roaming case, where MN attaches to AR2 in the home domain. The MN configures its LLOC and registers it to SBR1 by sending a LOC binding update message (Step 1). SBR1 stores the MN’s ID-LLOC mapping in its LMT, and responds with the LOC binding update ACK to the MN (Step 2). Subsequently, SBR1 sends a LOC binding update message to the central DDNS server to
register the MN’s ID-GLOC (Step 3). The DDNS server replies with the LOC binding update ACK to SBR 1 (Step 4) after storing the ID-GLOC in its database. We notice here that the GLOC is the SBR’s locator in ILNP-Local, as shown in Figure 3, whereas a GLOC is the MN’s locator in ILNP-Global. When CN has data packets for the MN, the packets are delivered to the SBR1 first. Then, SBR1 looks up its LMT to find the MN’s LLOC. Since the LLOC is in the LMT in non-roaming case, SBR1 knows the MN’s current location and forwards the data packet successfully.

We turn to the roaming case now, where MN attaches to AR3 in the visiting domain. Once the MN is connected to AR3, it configures its LLOC and registers it to SBR2 by sending a LOC binding update message (Step 5). SBR2 stores the MN’s ID-LLOC in its LMT and returns the LOC binding update ACK to the MN (Step 6). Right after, SBR2 sends a LOC binding update message to the centralized DDNS server (Step 7) to update the MN’s ID-GLOC mapping. The DDNS server then responds with the LOC binding update ACK to SBR2 (Step 8). For data delivery to the MN, CN sends data packets to SBR1 in the home domain. SBR1 searches the LMT but fails to find the MN’s LLOC. In that case, SBR1 refers to the DDNS server for the MN’s GLOC by sending the LOC query request message (Step 9). When the DDNS server returns the associated LOC query response (Step 10), SBR1 knows the MN’s GLOC and forwards the data packets to SBR2 in the visiting domain. The SBR2 delivers the packets to the MN since it has the MN’s LLOC in its LMT.

During the handover from AR3 to AR4, MN sends a LOC binding update message to SBR2 to update its ID-LLOC in the LMT (Step 11). The SBR2 replies with the LOC binding update ACK to the MN (Step 12). In succession, the MN transmits a LOC Update message to CN, to notify its new LOC. CN updates the MN’s LOC and responds with the LOC Update ACK message to the MN (Steps 13 and 14). CN can deliver the data packets to the MN via AR1, SBR1, SBR2 and AR4 in sequence.

We have looked over the existing mobility control schemes for ILNP-based mobile networks; i.e., ILNP-Global and ILNP-Local. We observe both the mobility schemes rely on the centralized DDNS server in some degrees. This results in poor scalability and performance in mobile environments.

Accordingly, we propose a new mobility control scheme in this paper, which manages the ID-LOC information in the fully distributed fashion.

**Figure 4.** ID-LOC management of ILNP-Local scheme.

We have looked over the existing mobility control schemes for ILNP-based mobile networks; i.e., ILNP-Global and ILNP-Local. We observe both the mobility schemes rely on the centralized DDNS server in some degrees. This results in poor scalability and performance in mobile environments.

Accordingly, we propose a new mobility control scheme in this paper, which manages the ID-LOC information in the fully distributed fashion.
3. Proposed Scheme

In this section, we describe our distributed mobility control scheme, named ILNP-Distributed, in detail.

3.1. Overview

We consider the mobility scenario of Figure 5 to discuss the operation of our proposed ILNP-Distributed scheme. In the scenario, MN (bob.SKT.com) and CN (alice.SKT.com) are in the same domain initially, and MN moves to another domain by roaming. Further, the MN changes the serving AR by handover in the visiting domain.

Figure 5. Mobility scenario for ILNP-Distributed scheme.

In the proposed scheme, each SBR has the dedicated mobile DDNS (m-DDNS) server and the server maintains two databases, called HHR and VHR. HHR keeps track of the ID-LOC (can be LLOC or GLOC) mappings for the home hosts, i.e., the hosts whose home domain is the SBR’s serving domain, as shown in Table 1. VHR stores the ID-LLOC mappings for the hosts visiting the SBR’s serving domain. In Table 2, we provide the example of VHR in LGU.com domain. We can use the host’s IP address as the LLOC and the associated SBR’s IP address as the GLOC. We manage the ID-LOC information similarly to the ILNP-Local, but we do not depend on the centralized DDNS server.

Table 1. Home Host Register (HHR) in SKT.com domain.

| No. | Host Name    | ID | LOC  | Status |
|-----|--------------|----|------|--------|
| 1   | alice.SKT.com | ID1 | LLOC | Home   |
| 2   | bob.SKT.com  | ID2 | GLOC | Roaming|
| 3   | ...          | ... | ...  | ...    |

Table 2. Visiting Host Register (VHR) in LGU.com domain.

| No. | Host Name    | ID | LOC  | Home Domain |
|-----|--------------|----|------|-------------|
| 1   | bob.SKT.com  | ID2 | LLOC | GLOC        |
| 2   | ...          | ... | ...  | ...         |
| 3   | ...          | ... | ...  | ...         |

Before discussing the proposed scheme further, we compare our scheme with the existing mobility control schemes in the architectural perspective, as in Table 3.
Table 3. Comparison of candidate mobility control schemes.

| Schemes             | LOC       | Mapping Architecture | Mobility Agent   |
|---------------------|-----------|----------------------|------------------|
| ILNP-Global         | GLOC      | Centralized          | DDNS             |
| ILNP-Local          | LLOC, GLOC| Centralized          | SBR (LMT), DDNS  |
| ILNP-Distributed    | LLOC, GLOC| Distributed          | SBR (m-DDNS)     |

The ILNP-Global employs only the GLOC as a LOC. A central DDNS server is used for the ID-GLOC mappings. The ID-GLOC binding and query operations are performed between each host and the DDNS server. The ILNP-Local is similar to the ILNP-Global. However, it has two types of LOCs, i.e., GLOC and LLOC, for each host. Each SBR performs the localized ID-LLOC mappings for the hosts in the serving domain, while it depends on the centralized DDNS server for the ID-GLOC information of the hosts in the visiting domains. In the ILNP-Distributed, we use LLOC and GLOC, as in ILNP-Local. Each SBR (or m-DDNS server) manages HHR and VHR databases. The ID-LOC information is managed by m-DDNS servers in the distributed manner without any centralized entities. We describe the detailed operations in the subsequent sections.

3.2. Procedures for Non-Roaming Scenario

When MN establishes the network connection with an AR in the home domain, it performs the ID-LOC binding and query operations as in Figure 6.

![Figure 6. Binding and query operations for non-roaming.](image)

MN registers the LLOC to SBR1 by sending a LOC binding update message (Step 1). The SBR1 stores the MN’s ID-LLOC in the HHR, which maintains the ID-LOC (can be LLOC or GLOC) mappings of all the home hosts. Then the SBR1 returns the LOC binding update ACK message to the MN without further action (Step 2). When CN has data packets for the MN, CN queries the MN’s LLOC to SBR1 by sending a LOC query request message (Step 3). Since the MN is in the home domain, SBR1 finds the LLOC in the HHR and returns it to the CN by the LOC query reply message (Step 4). CN then sends the data packets to the MN directly.

3.3. Procedures for Roaming Scenario

We illustrate the ID-LOC binding and query operations in Figure 7. When MN visits another domain (roaming case). When MN attaches to AR2 in the visiting domain, the MN sends a LOC binding update message to the SBR2 (Step 1), to register its LLOC. SBR2 then checks whether the MN belongs to its serving domain (home host) or not (roaming host). In the proposed scheme, we assume that each host has a name with its home domain information. For example, if a host has a name of alice.SKT.com, we can see its home domain is SKT.com, so the SBR2 recognizes that the MN is a roaming host, and stores the MN’s ID-LLOC in VHR. In succession, SBR2 sends a LOC binding update message to the SBR1 in the MN’s home domain (Step 2). SBR2 locates the SBR1 by using the m-DDNS server and the MN’s home domain information. Then, SBR1 updates the MN’s ID-GLOC mapping in the HHR and responds with the LOC binding update ACK to SBR2 (Step 3). The ACK message is eventually delivered to the roaming MN also by SBR2 (Step 4). When CN has data packets for the MN, the CN requests the MN’s LOC to the SBR1 by sending a LOC query request message.
(Step 5). SBR1 finds the MN’s GLOC in the HHR since the host is in the visiting domain, and returns it to the CN by the LOC query reply message (Step 6). With the GLOC, CN can send the data packets to the MN via AR1, SBR1, SBR2 and AR3 in sequence.

**Figure 7.** Binding and query operations for roaming.

### 3.4. Procedures for the Handover Scenario

We assume the proposed scheme exploits the link-layer information for handover, as defined in the IEEE 802.21 standard [18]. With the help of link-layer triggers, e.g. link-detected (LD) and link-up (LU), MN can tell if it has moved to a new network or not, while connected to the old network.

We show the ID-LOC management operation during the handover in Figure 8, under our ILNP-Distributed scheme. When MN detects a new network, it sends a LOC binding update message to SBR2 (Step 1), in order to register the LLOC in the VHR therein. After updating the ID-LLOC mapping, SBR2 replies to the MN with the LOC binding update ACK message (Step 2). We notice that the MN’s GLOC does not change during handover. Hence the packets from CN arrive at SBR2 still. SBR2 looks up VHR and finds the LLOC of the MN. Then the packets are delivered to the MN via AR4 from SBR2.

**Figure 8.** Handover operations.

### 4. Performance Analysis

In this section, we analyze the three candidate mobility control schemes for ILNP networks; i.e., ILNP-Global, ILNP-Local and INLP-Distributed. Especially, we evaluate their performances in terms of control traffic overhead, total transmission delay and handover delay.

#### 4.1. Analysis Model

We consider the network model of Figure 9 to analyze the mobility control schemes. We first summarize the notations used for the analysis in Table 4 [15–17,19–21].
We denote the transmission delay of a message with size $S$ as $T_{x-y}(s)$ \cite{15,17,19–21}, where $x$ and $y$ are the sending and receiving nodes connected by a wireless link. We note that $T_{x-y}(s)$ can be written as

$$T_{x-y}(s) = \frac{1+q}{1-q} \times \left[ \frac{S}{B_{wl}} + L_{wl} \right].$$

Similarly, we let $T_{x-y}(S,H_{x-y})$ \cite{15,17,19–21} denote the transmission delay of a message with size $S$ in the wired network, where $x$ and $y$ are the sending and receiving nodes connected by one or more wired links and $H_{x-y}$ denotes the hop count between $x$ and $y$. It is straightforward to write $T_{x-y}(S,H_{x-y})$ as

$$T_{x-y}(S,H_{x-y}) = \frac{1+q}{1-q} \times \left[ \frac{S}{B_w} + L_w + T_q \right].$$

### 4.2. Analysis of Control Traffic Overhead

We evaluate the control traffic overhead (CTO) for ID-LOC mapping managements at the SBRs and/or the DDNS server.
4.2.1. ILNP-Global

In the ILNP-Global scheme, we evaluate the CTO by the number of ID-LOC mapping messages the DDNS server handles. We consider the non-roaming case first. There, MN attaches to the serving AR, and registers its LLOC to the DDNS server by sending a LOC binding update message. Therefore, the DDNS server processes $S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$ bytes of the control messages. Furthermore, each host sends a LOC query request message to the DDNS server for data delivery, which incurs additional $S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$ bytes of control messages to the server. Therefore, the CTO for non-roaming case can be written as

$$CTO - N_{RLNP-Global} = S_C \times N_{Host/AR} \times N_{AR} + S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$$

(1)

In ILNP-Global, we note that the mobility control messages do not change for roaming and non-roaming cases. We, hence, obtain the CTO for roaming case as

$$CTO - R_{RLNP-Global} = CTO - N_{RLNP-Global}$$

(2)

4.2.2. ILNP-Local

In ILNP-Local, we first consider the non-roaming case, where we measure the CTO as the number of ID-LOC mapping messages processed by SBR1 and the DDNS server. When MN attaches to the AR in the home domain, the MN registers its LLOC to SBR1 by sending a LOC binding update message. That is, the SBR1 should handle $S_C \times N_{Host/AR} \times N_{AR}$ bytes of control messages. Subsequently, SBR1 sends a LOC binding update message to the DDNS server to register the MN’s GLOC, so the DDNS server handles $S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$ bytes of control messages from the SBRs. Accordingly, the CTO for non-roaming case is given as

$$CTO - N_{RLNP-local} = S_C \times N_{Host/AR} \times N_{AR} + S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$$

(3)

We now take into account the roaming case, where we evaluate the CTO by the amount of ID-LOC mapping messages at SBR2 and the DDNS server. When MN attaches to the AR in the visiting domain, the MN registers its LLOC to SBR2 by sending a LOC binding update message. Hence, the SBR2 handles $S_C \times N_{Host/AR} \times N_{AR}$ bytes of control messages. Then, the SBR2 sends a LOC binding update message to the DDNS server to update the MN’s ID-GLOC information. For this, the DDNS server processes $S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$ bytes of control messages. Further, when CN has packets for the MN, CN passes the packets to the SBR1 first. SBR1 sends the LOC query request message to the DDNS server to obtain the MN’s GLOC, which incurs additional $S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$ bytes of control messages. Thus, the CTO in roaming case can be written as

$$CTO - R_{RLNP-local} = S_C \times N_{Host/AR} \times N_{AR} + S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR} + S_C \times N_{Host/AR} \times N_{AR} \times N_{SBR}$$

(4)

4.2.3. ILNP-Distributed

For non-roaming case, we measure the CTO by the amount of ID-LOC mapping messages at SBR1. When MN attaches to the serving AR, it sends the LOC binding update messages to SBR1 and registers its LLOC. So SBR1 processes $S_C \times N_{Host/AR} \times N_{AR}$ bytes of control messages. Moreover, if CN has data packets for the MN, it sends a LOC query request message to the SBR1. Then the SBR1 should handle additional $S_C \times N_{Host/AR} \times N_{AR}$ bytes of control messages. Thus, we can write the CTO as

$$CTO - N_{RLNP-Distributed} = S_C \times N_{Host/AR} \times N_{AR} + S_C \times N_{Host/AR} \times N_{AR}$$

$$CTO - N_{RLNP-Distributed} = 2 \times S_C \times N_{Host/AR} \times N_{AR}$$

(5)
When MN visits a foreign domain (roaming case), we measure the CTO as the amount of ID-LOC mapping messages processed by SBR1 and SBR2. MN attaches to AR3 and registers its LLOC to SBR2 by sending a LOC binding update messages. Hence, the SBR2 processes \( S_C \times N_{\text{Host}/AR} \times N_{AR} \) bytes of control messages. Successively, SBR2 sends a LOC binding update message to SBR1 to update the HHR entry. So the SBR1 handles \( S_C \times N_{\text{Host}/AR} \times N_{AR} \) bytes of the control messages. Additionally, for data delivery, CN inquires the MN’s LOC of SBR1 by sending a LOC query request message. Then, SBR1 processes \( S_C \times N_{\text{Host}/AR} \times N_{AR} \) bytes of control messages. In total, the CTO for the roaming case is given as

\[
CTO = RO_{\text{ILNP–distributed}} = S_C \times N_{\text{Host}/AR} \times N_{AR} + S_C \times N_{\text{Host}/AR} \times N_{AR} + S_C \times N_{\text{Host}/AR} \times N_{AR} 
\]

\[
CTO = RO_{\text{ILNP–distributed}} = 3 \times S_C \times N_{\text{Host}/AR} \times N_{AR} \tag{6}
\]

4.3. Analysis of Total Transmission Delay

In this section, we derive the total transmission delay (TTD) for the ID-LOC management and data delivery. We notice that the TTD can be written as the sum of binding update delay (BUD), binding query delay (BQD) and data delivery delays (DDD), respectively. Hence, for each mobility scheme, we calculate the BUD, BQD and DDD separately and sum up them to obtain the TTD.

4.3.1. ILNP-Global

When MN attaches to the AR1 in the home domain (non-roaming case), it conducts the binding update with the DDNS server by exchanging the LOC binding update and ACK messages. This operation takes \( 2 \times (T_{\text{MN–AR}}(S_C) + T_{\text{AR–SBR}}(S_C,H_{\text{AR–SBR}}) + T_{SBR–DDNS}(S_C,H_{\text{SBR–DDNS}})) + P_{\text{DDNS}}, \) where \( P_{\text{DDNS}} = \beta \log(N_{AR} \times N_{\text{Host}/AR} \times N_{SBR}) \) represents the processing time in the DDNS server for the ID-GLOC update. We assume the processing time is proportional to the logarithm of the number of active hosts \( (N_{AR} \times N_{\text{Host}/AR} \times N_{SBR}), \) which can be attained with tree-based data structures. So, the BUD is written as

\[
BUD = 2 \times (T_{\text{MN–AR}}(S_C) \times T_{\text{AR–SBR}}(S_C,H_{\text{AR–SBR}}) + T_{SBR–DDNS}(S_C,H_{\text{SBR–DDNS}})) + P_{\text{DDNS}}
\]

\[
BUD = NR_{\text{ILNP–Global}} = 2 \times (T_{\text{MN–AR}}(S_C) \times T_{\text{AR–SBR}}(S_C,H_{\text{AR–SBR}}) + T_{SBR–DDNS}(S_C,H_{\text{SBR–DDNS}})) + \beta \log(N_{\text{Host}/AR} \times N_{AR} \times N_{SBR}).
\]

In ILNP-Global, the binding update operation does not change regardless of the roaming and non-roaming cases. Hence, we obtain

\[
BUD = RO_{\text{ILNP–Global}} = BUD = NR_{\text{ILNP–Global}}.
\]

We next calculate the time for binding query operations. When CN has the data packets for MN in the home domain (non-roaming case), it sends a LOC query request message to the DDNS server to obtain the MN’s GLOC. The DDNS server searches its database, which takes \( P_{\text{DDNS}} = \beta \log(N_{\text{Host}/AR} \times N_{\text{SBR}}), \) and responds to the CN by the LOC query reply message. This operation takes \( 2 \times (T_{\text{CN–AR}}(S_C) + T_{\text{AR–SBR}}(S_C,H_{\text{AR–SBR}}) + T_{SBR–DDNS}(S_C,H_{\text{SBR–DDNS}})) \). Thus, we have the BQD for non-roaming case as

\[
BQD = NR_{\text{ILNP–Global}} = 2 \times (T_{\text{CN–AR}}(S_C) \times T_{\text{AR–SBR}}(S_C,H_{\text{AR–SBR}}) + T_{SBR–DDNS}(S_C,H_{\text{SBR–DDNS}})) + \beta \log(N_{\text{Host}/AR} \times N_{AR} \times N_{SBR}).
\]

When the MN is visiting another domain (roaming case), CN obtains the MN’s GLOC from the DDNS server, following the same binding query operation. So, the BQD for roaming case is given as
\[
BQD - RO_{ILNP-Global} = BQD - NR_{ILNP-Global}
\]

We now investigate the data delivery of the ILNP-Global. If MN is in the home domain (non-roaming case), CN delivers data packets to the MN via AR1, SBR1 and AR2 in sequence. Hence the DDD is written as

\[
DDD - NR_{ILNP-Global} = T_{CN-AR}(S_d) \times T_{AR-SBR}(S_d, H_{AR-SBR}) + T_{MN-AR}(S_d).
\]

When MN is visiting another domain (roaming case), CN delivers data packets to the MN via AR1, SBR1, SBR2 and AR3 in sequence. So, the DDD can be written as

\[
DDD - RO_{ILNP-Global} = T_{CN-AR}(S_d) \times T_{AR-SBR}(S_d, H_{AR-SBR}) + T_{SBR-SBR}(S_d, S_d, H_{SBR-SBR}) + T_{AR-SBR}(S_d, H_{AR-SBR}).
\]

In total, we derive the TTD for non-roaming case and the TTD for roaming case as

\[
TTD - NR_{ILNP-Global} = BUD - NR_{ILNP-Global} + BQD - NR_{ILNP-Global} + DDD - NR_{ILNP-Global}
\]

\[
TTD - RO_{ILNP-Global} = BUD - RO_{ILNP-Global} + BQD - RO_{ILNP-Global} + DDD - RO_{ILNP-Global}
\]

4.3.2. ILNP-Local

In ILNP-Local, the ID-LOC binding updates are performed as follows. When MN attaches to AR1 in the home domain (non-roaming case), it registers the LLOC to SBR1 by exchanging the LOC binding update and ACK messages. This operation takes \(2 \times (T_{MN-AR}(S_c) + T_{AR-SBR}(S_c, H_{AR-SBR}) + P_{SBR})\), where \(P_{SBR} = a \log(N_{AR} \times N_{Host/AR} \times N_{SBR})\) is the LMT update time at SBR1. Subsequently, SBR1 updates the MN’s GLOC in the DDNS server by exchanging the LOC binding update and ACK messages. Considering the database update time PDDNS at the DDNS server, this operation takes \(T_{SBR-DDNS}(S_c, H_{SBR-DDNS}) + P_{DDNS}\), where \(P_{DDNS} = a \log(N_{AR} \times N_{Host/AR} \times N_{SBR})\). So, the BUD for non-roaming case can be written as

\[
BUD - NR_{ILNP-Local} = 2 \times (T_{MN-AR}(S_c) \times T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-DDNS}(S_c, H_{SBR-DDNS}) + P_{SBR} + P_{DDNS})
\]

The binding update process is not affected by the roaming and non-roaming cases. Hence, the BUD for roaming case is given as

\[
BUD - RO_{ILNP-Local} = BUD - NR_{ILNP-Local}
\]

We then turn to the binding query delay. In ILNP-Local, the binding query is resolved in the SBR locally for the non-roaming case. Thus, the BQD is zero.

\[
BQD - NR_{ILNP-Local} = 0.
\]

When MN is in the foreign domain (roaming case), the binding query is processed as follows. First, CN passes the data packets to SBR1. SBR1 searches the LMT in vain and requests the MN’s GLOC to the DDNS server by sending a LOC query request message. The DDNS server looks for the GLOC in its database, which takes \(P_{DDNS} = \beta \log(N_{AR} \times N_{Host/AR} \times N_{SBR})\), and responds to SBR1 by the LOC query reply message. This operation takes \(2 \times T_{SBR-DDNS}(S_c, H_{SBR-DDNS})\). So, the BQD for roaming case can be written as

\[
BQD - RO_{ILNP-Local} = 2 \times T_{SBR-DDNS}(S_c, H_{SBR-DDNS}) + \beta \log(N_{Host/AR} \times N_{AR} \times N_{SBR})
\]
Once the binding query is resolved, data packets are transmitted from CN to MN. When the MN is in the home domain (non-roaming case), data packets are delivered from CN to MN via AR1, SBR1, and AR2 in sequence, such that

\[
\text{DDD} - NR_{\text{ILNP-Local}} = T_{\text{CN-AR}}(S_d) \times T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + T_{\text{MN-AR}}(S_d).
\]

If the MN is visiting another domain (roaming case), data packets from CN are delivered to MN via AR1, SBR1, SBR2, and AR3 in sequence. So the DDD is given as

\[
\text{DDD} - RO_{\text{ILNP-Local}} = T_{\text{CN-AR}}(S_d) \times T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + T_{\text{SBR-SBR}}(S_d, H_{\text{SBR-SBR}}) + T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + T_{\text{MN-AR}}(S_d).
\]

Therefore, we obtain the TTD for non-roaming case and the TTD for roaming case as

\[
\text{TTD} - NR_{\text{ILNP-Local}} = BUD - NR_{\text{ILNP-Local}} + BQD - NR_{\text{ILNP-Local}} + DDD - NR_{\text{ILNP-Local}}
\]

(9)

\[
\text{TTD} - RO_{\text{ILNP-Local}} = BUD - RO_{\text{ILNP-Local}} + BQD - RO_{\text{ILNP-Local}} + DDD - RO_{\text{ILNP-Local}}
\]

(10)

4.3.3. ILNP-Distributed

In the ILNP-Distributed, when MN attaches to AR1 in the home domain (non-roaming case), it performs the ID-LOC binding operation with SBR1 by exchanging the LOC binding update and ACK messages. Then, SBR1 updates the HHR entry. This operation takes \(2 \times (T_{\text{MN-AR}}(S_c) + T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})) + P_{\text{SBR}}\) where \(P_{\text{SBR}} = a\log(N_{\text{AR}} \times N_{\text{HHR/AR}})\). Hence, the BUD for non-roaming case can be written as

\[
BUD - NR_{\text{ILNP-Distributed}} = 2 \times (T_{\text{MN-AR}}(S_c) \times T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})) + P_{\text{SBR}}
\]

When MN is not in the home domain (roaming case), the MN conducts the binding update operation with SBR2 by exchanging the LOC binding update and ACK messages. Incorporating the VHR update time at SBR2, this operation takes \(2 \times (T_{\text{MN-AR}}(S_c) + T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})) + P_{\text{SBR}}\), where \(P_{\text{SBR}} = a\log(N_{\text{AR}} \times N_{\text{HHR/AR}})\). In succession, SBR2 performs the binding update operation with SBR1 by exchanging the LOC binding update and ACK messages. Therefore, the SBR1 updates the ID-GLOC information in the HHR. This operation takes \(T_{\text{SBR-SBR}}(S_c, H_{\text{SBR-SBR}})\) and \(P_{\text{SBR}}\), where \(P_{\text{SBR}} = a\log(N_{\text{AR}} \times N_{\text{HHR/AR}})\). So, the BUD for roaming case is given as

\[
BUD - RO_{\text{ILNP-Distributed}} = 2 \times (T_{\text{MN-AR}}(S_c) \times T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}}) + T_{\text{SBR-SBR}}(S_c, H_{\text{SBR-SBR}})) + P_{\text{SBR}}
\]

We now calculate the BQD in the non-roaming case. First, CN sends a LOC query request message to SBR1 to find the MN’s LLOC. SBR1 looks for the LLOC in the HHR, which takes \(P_{\text{SBR}} = b\log(N_{\text{AR}} \times N_{\text{HHR/AR}})\), and responds to the CN by the LOC query reply. This takes \(2 \times T_{\text{CN-AR}}(S_c) + 2 \times T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})\). Hence, we can write the BQD for non-roaming case as

\[
BQD - NR_{\text{ILNP-Distributed}} = 2 \times T_{\text{CN-AR}}(S_c) + 2 \times T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}}) + b\log(N_{\text{Host/AR}} \times N_{\text{AR}})
\]

(11)

(12)

In the ILNP-Distributed, the binding query process is the same for roaming and non-roaming cases. So, we have

\[
BQD - RO_{\text{ILNP-Distributed}} = BQD - NR_{\text{ILNP-Distributed}}
\]

When MN is in the home domain (non-roaming case), CN can send data packets to the MN via AR1 and AR2 in sequence, once it acquires the MN’s LOC. Hence, the DDD is given as
$$DDD - NR_{ILNP-Distributed} = T_{CN-AR} (S_d) \times T_{AR-ACK} (S_d, H_{AR-AR}) + T_{MN-AR} (S_d).$$

If MN is in the foreign domain (roaming case), the data packets from CN are delivered to the MN via AR1, SBR1, SBR2 and AR3 in sequence, such that DDD is written as

$$DDD - RO_{ILNP-Distributed} = T_{CN-AR} (S_d) \times T_{AR-SBR} (S_d, H_{AR-SBR}) + T_{SBR-DDNS} (S_d, H_{SBR-DDNS}) + T_{AR-SBR} (S_d, H_{AR-SBR}) + T_{MN-AR} (S_d).$$

Conclusively, we obtain the TTD for non-roaming case and the TTD for roaming case as

$$TTD - NR_{ILNP-Distributed} = BUD - NR_{ILNP-Distributed} + BQD - NR_{ILNP-Distributed}$$

$$TTD - RO_{ILNP-Distributed} = BUD - RO_{ILNP-Distributed} + BQD - RO_{ILNP-Distributed}$$

4.4. Analysis of Handover Delay

We here analyze the handover delay (HOD) of the three candidate mobility control schemes.

4.4.1. ILNP-Global

Completing the handover to AR4, MN updates its GLOC in the DDNS server by exchanging the LOC binding update and ACK messages. Considering the database update time at the DDNS server, it takes $2 \times (T_{MN-AR}(S_c) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-DDNS}(S_c, H_{SBR-DDNS})) + P_{DDNS}$, where $P_{DDNS} = a \log(N_{AR} \times N_{Host(AR)} \times N_{SBR})$. Then, the MN notifies the new GLOC to CN by exchanging the LOC Update and ACK messages, which takes the additional time of $2 \times (T_{MN-AR}(S_c) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-SBR}(S_c, H_{AR-SBR}) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{CN-AR}(S_d))$. At last, CN delivers the data packets to the MN via AR1, SBR1, SBR2 and AR4, which takes $T_{CN-AR}(S_d) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-SBR}(S_c, H_{AR-SBR}) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{AR-MN}(S_d)$.

Hence, in ILNP-Global, we obtain the HOD as

$$HOD_{ILNP-Global} = 2 \times (T_{MN-AR} (S_c) + T_{AR-SBR} (S_c, H_{AR-SBR})$$

$$+ T_{SBR-DDNS} (S_c, H_{SBR-DDNS}) + a \log(N_{Host(AR)} \times N_{AR} \times N_{SBR})$$

$$+ 2 \times (T_{MN-AR} (S_c) + T_{AR-SBR} (S_c, H_{AR-SBR}) + T_{SBR-SBR} (S_c, H_{AR-SBR}) + T_{AR-SBR} (S_c, H_{AR-SBR}) + T_{CN-AR} (S_d))$$

$$+ T_{AR-SBR} (S_c, H_{AR-SBR}) + T_{AR-SBR} (S_d, H_{SBR-SBR}) + (T_{MN-AR} (S_d)).$$

4.4.2. ILNP-Local

MN makes a handover to AR4 and updates its LLOC at SBR2, by exchanging the LOC binding update and ACK messages. With the LMT update time at SBR, i.e., $P_{SBR}$, it takes $2 \times (T_{MN-AR}(S_c) + T_{AR-SBR}(S_c, H_{AR-SBR}) + P_{SBR})$, where $P_{SBR} = a \log(N_{AR} \times N_{Host(AR)}$. The MN then notifies the LOC to CN by exchanging the LOC Update and ACK messages, which takes $2 \times (T_{MN-AR}(S_c) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-SBR}(S_c, H_{SBR-SBR}) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{CN-AR}(S_d))$. The CN delivers the data packets to the MN via AR1, SBR1, SBR2 and AR4 in sequence. The delivery time of each packet can be written as $T_{CN-AR}(S_d) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{SBR-SBR}(S_c, H_{SBR-SBR}) + T_{AR-SBR}(S_c, H_{AR-SBR}) + T_{AR-MN}(S_d)$, we obtain the HOD of ILNP-Local as

$$HOD_{ILNP-Local} = 2 \times (T_{MN-AR} (S_c) \times T_{AR-SBR} (S_c, H_{AR-SBR}))$$

$$+ a \log(N_{Host(AR)} \times N_{AR} \times N_{SBR}) + 2 \times (T_{MN-AR} (S_c))$$

$$+ T_{AR-SBR} (S_c, H_{AR-SBR}) + T_{SBR-SBR} (S_c, H_{SBR-SBR})$$

$$+ T_{AR-SBR} (S_c, H_{AR-SBR}) + T_{CN-AR} (S_d) + T_{AR-SBR} (S_d, H_{AR-SBR})$$

$$+ T_{SBR-SBR} (S_d, H_{SBR-SBR}) + T_{AR-SBR} (S_d, H_{AR-SBR}) + (T_{MN-AR} (S_d))$$

(14)
4.4.3. ILNP-Distributed

In ILNP-Distributed, when MN changes the serving AR, it updates the LLOC in SBR2 by exchanging the LOC binding update and ACK messages. Letting PSBR denote the VHR update time, we express the operation time as

$$\text{operation time} = 2 \times (T_{\text{MN-AR}}(S_c) + T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})) + P_{\text{SBR}} = \alpha \log(N_{\text{AR}} \times N_{\text{Host/AR}}).$$

Then, SBR2 can deliver the data packets from CN to the MN with additional delay of

$$T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + T_{\text{AR-MN}}(S_d).$$

We hence obtain the HOD of ILNP-Distributed as

$$HOD_{\text{ILNP-Distributed}} = 2 \times (T_{\text{MN-AR}}(S_c) + T_{\text{AR-SBR}}(S_c, H_{\text{AR-SBR}})) + \alpha \log(N_{\text{Host/AR}} \times N_{\text{AR}} \times N_{\text{SBR}}) + T_{\text{AR-SBR}}(S_d, H_{\text{AR-SBR}}) + (T_{\text{MN-AR}}(S_d)).$$

(15)

5. Numerical Results

We compare the performances of the considered mobility control schemes, i.e., ILNP-Global, ILNP-Local and ILNP-Distributed, based on the equations in the previous section. For numerical results, we use the parameter values in Table 5, as in [15–17,19–21].

| Parameter | Default | Minimum | Maximum |
|-----------|---------|---------|---------|
| $L_{\text{wl}}$ (ms) | 10 | 1 | 55 |
| $q$ | 0.1 | 0.1 | 0.8 |
| $T_q$ (ms) [15–17,19–21] | 5 | 1 | 55 |
| $H_{\text{AR-SBR}}$ | 10 | 1 | 55 |
| $N_{\text{Host/AR}}$ | 500 | 100 | 10000 |
| $N_{\text{AR}}$ | 30 | 1 | 45 |
| $H_{\text{SBR-DDNS}}$ | 20 | 1 | 55 |
| $N_{\text{SBR}}$ | 100 |
| $H_{\text{AR}}$ | $\sqrt{N_{\text{AR}}}$ |
| $H_{\text{SBR-SBR}}$ | $\sqrt{N_{\text{SBR}}}$ |
| $\alpha$ [17] | 0.1 |
| $\beta$ [17] | 0.2 |
| $L_{\text{wl}}$ | 2 ms |
| $S_c$ | 50 bytes |
| $S_d$ | 200 bytes |
| $B_{\text{wl}}$ | 11 Mbps |
| $B_o$ | 100 Mbps |

5.1. Non-Roaming Case

We discuss the performance results for the non-roaming case.

5.1.1. Control Traffic Overhead (CTO)

We show the CTOs, i.e., amount of control messages processed at the SBRs and/or the DDNS server, of the considered mobility control schemes in Figures 10 and 11, for different $N_{\text{Host/AR}}$ and $N_{\text{AR}}$ respectively. Our proposed scheme, ILNP-Distributed, has a significantly smaller CTO, compared to the existing mobility schemes. This is due to the fact that our scheme distributes the ID-LOC management messages over all the SBRs in the network. The performance gaps increase fast as the numbers of hosts and/or ARs increase. We also note that the ILNP-Local has a smaller CTO than the ILNP-Global because the ILNP-Local scheme does not conduct the binding query operations for the hosts in the home domain (non-roaming case).

5.1.2. Total Transmission Delay (TTD)

In Figure 12, we demonstrate the impact of wireless link delay ($L_{\text{wl}}$) on the TTD. We observe that TTD increases steadily with $L_{\text{wl}}$ for every mobility control scheme. However, among them, our
proposed scheme shows the minimal delay. Further, we note that the ILNP-Local has smaller TTDs than the ILNP-Global since the ILNP-Local scheme does not perform the binding query operations for intra-domain data delivery.

Next, we show the impact of wireless link failure probability ($q$) on the TTD in Figure 13. In all the mobility schemes, the TTD increases with $q$. The growth rate is low with small $q$ but becomes high with large $q$. We can see that our ILNP-Distributed scheme has smaller TTDs for any fail probabilities, compared to the two existing schemes. Further, the ILNP-Local outperforms the ILNP-Global since the ILNP-Local scheme does not conduct the binding query operation for intra-domain data delivery.

In Figure 14, we observe the TTD curves while varying the average queuing delay ($T_q$) at each router. TTD increases quickly with larger $T_q$ in all the candidate schemes. However, our proposed scheme shows the smallest value and the smallest growth rate among them. We note that the ILNP-Local is better than the ILNP-Global since the ILNP-Local scheme does not perform the binding query operation for intra-domain data delivery.

![Figure 10. Impact of $N_{host/AR}$ on CTO.](image1)

![Figure 11. Impact of $N_{AR}$ on CTO.](image2)
Figure 12. Impact of $L_{wl}$ on TTD.

Figure 13. Impact of $q$ on TTD.

Figure 14. Impact of $T_q$ on TTD.

Figure 15 shows the impact of the hop count between AR and SBR ($H_{AR,SBR}$) on the TTD. TTD increases with the hop count for all the mobility schemes. Our ILNP-Distributed outperforms the competitors for any hop counts, and the ILNP-Local works better than the ILNP-Global.
We provide the TTD curves for different hop counts between SBR and the DDNS server (H_{SBR-DDNS}), in Figure 16. TTDs of the existing schemes are significantly affected by H_{SBR-DDNS}, since many binding messages are exchanged between SBRs and the DDNS server. On the contrary, our scheme does not employ the DDNS server and its TTD remains constant regardless of H_{SBR-DDNS}.

![Figure 15. Impact of H_{AR-SBR} on TTD.](image1)

![Figure 16. Impact of H_{SBR-DDNS} on TTD.](image2)

5.2. Roaming Case

We here study the roaming case.

5.2.1. Control Traffic Overhead (CTO)

We present the CTOs for different N_{NodiAR} and N_{AR}, in Figures 17 and 18 respectively. We observe that our proposed ILNP-Distributed shows a smaller CTO than the existing schemes, which can be explained as follows. In the previous mobility schemes, all the control messages are processed by the DDNS server. On the contrary, in our scheme, the control messages are distributed to the SBRs in the network. Their performance gaps increase rapidly with the number of hosts and/or ARs. We can see that the ILNP-Local performs poorly in roaming case and its CTO curves are overlapped with those of the ILNP-Global, due to the frequent binding update operations between each SBR and the DDNS server.
5.2.2. Total Transmission Delay (TTD)

We show the impact of wireless link delay ($L_{wl}$) on the TTD in Figure 19. The TTD increases steadily with $L_{wl}$ in all the candidate mobility schemes. However, our ILNP-Distributed has, relatively, a smaller TTD than the existing schemes. Moreover, the ILNP-Local is better than the ILNP-Global. Their performance gap increases with $L_{wl}$.

In Figure 20, we show the impact of wireless link failure probability ($q$) on the TTD. The TTD increases with $q$ slowly for small value, but the growth rate increases for large failure probability. We can see that our ILNP-Distributed has the smallest TTD compared to the competitors. We also note that the ILNP-Local works better than the ILNP-Global since the ILNP-Local scheme does not perform the binding query operations for intra-domain data delivery.
Figure 19. Impact of $L_{wl}$ on TTD.

Figure 20. Impact of $q$ on TTD.

In Figure 21, we compare the TTDs for average queuing delay ($T_q$) at each router. The TTD value increases with $T_q$ for all the mobility control schemes, but our scheme has smaller values than the existing schemes for any $T_q$.

We show the impact of the hop count between AR and SBR ($H_{AR-SBR}$) on the TTD, in Figure 22. We note that the TTD increases with the hop count for all the candidate schemes. Our proposed scheme works well when the hop count is small. However, as the hop count increases, it performs poorer than the ILNP-Local. This is because our ILNP-Distributed scheme conducts the binding query operations for data delivery between hosts and the SBR, for both roaming and non-roaming cases. In ILNP-Distributed the processing time have been done twice on PSBR.

Figure 23 compares the TTDs of the candidate schemes for different hop counts between SBRs and the DDNS server ($H_{SBR-DDNS}$). We can see that $H_{SBR-DDNS}$ has significant impacts on the TTD in the two existing schemes. In those schemes, the ID-LOC management messages are exchanged between SBRs and the DDNS server frequently. So, the two existing schemes depend on the centralized DDNS. The binding update and binding query operations perform with centralized DDNS.
5.3. Handover Delay (HOD)

In Figure 24, we present the impact of wireless link delay ($L_{wl}$) on the HOD. We observe the HOD increases with $L_{wl}$ for all the candidate schemes. Our ILNP-Distributed scheme has a significantly smaller handover delay than the existing schemes. Moreover, the ILNP-Local works better than the ILNP-Global since the ILNP-Local conducts the binding update operations with the SBR only. We recall that in ILNP-Global, the binding updates after handover occurs between the hosts and the centralized DDNS server.

We show the impact of wireless link failure probability ($q$) on the HOD in Figure 25. For every mobility control scheme, the HOD increases with $q$, but our scheme performs much better than the
existing schemes. The ILNP-Local gives smaller handover delay than the ILNP-Global due to the binding update operations within the domain.

**Figure 24.** Impact of \( L_{\text{d}} \) on HOD.

**Figure 25.** Impact of \( q \) on HOD.

We compare the HODs of candidate schemes while varying the average queuing delay \( T_q \) at each router, in Figure 26. The HOD increases with \( T_q \) for every mobility scheme. However, among them, our proposed scheme shows much better performance than any other schemes.

In Figure 27, we demonstrate the impact of the hop count between AR and SBR \( (H_{\text{AR-SBR}}) \) on the HOD. We can see that the HOD increases with the hop count for all the mobility schemes, but the rate is relatively slow in our proposed scheme. The ILNP-Global and the ILNP-Local have a similar tendency while keeping a constant gap.
5.4. Implementation Perspective

For applicability validation of the proposed scheme, we performed the testbed experimentation. For the testbed the locator was implemented by using the IPv4 address. The device ID was implemented by using the IPv6 address, which operates as 3.5 layer over IPv4.

The proposed scheme operations at GW can be summarized in Figure 28, which were implemented by using the Click Router software. When a packet arrives at the GW, it checks whether it is a control or data packet. If it is a control packet, the GW identifies the home GW for the device, based on the home domain information. Note that the home GW of device can be identified from the home domain information. In case that the packet is a data packet, if the GW already knows the LOC of the corresponding device, it will forward the packet directly. Otherwise, GW should first get the LOC of the device by contacting with the home GW of the device.
6. Conclusions

In this paper, we propose a new distributed mobility control scheme for mobile ILNP networks. Each SBR has the dedicated m-DDNS server, which maintains a Home Host Register (HHR) and a Visiting Host Register (VHR) to support the roaming. We let the m-DDNS trace all the ID-LOCs of the home hosts and the visiting hosts in the distributed manner. Hence, we obtain the LOC of any host of interest from the SBR (or m-DDNS server) of its home domain. By numerical analysis, we showed that our proposed scheme performs much better than the existing mobility control schemes in terms of control traffic overhead, total transmission delay and handover delay.

Author Contributions: M.G. wrote the initial manuscript; J.-G.C. and S.-J.K. revised the manuscript. M.M. and A.U.R. proofread the manuscript. W.A. conducted the performance analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2018R1D1A1B07048948), and in part by the 2019 Yeungnam University Research Grant.

Conflicts of Interest: The authors declare no conflict of interest

List of Acronyms

| Acronym | Description |
|---------|-------------|

Figure 28. Algorithm at GW for mapping management and data transmission.
| Acronym     | Description                                      |
|-------------|--------------------------------------------------|
| ILNP        | Identifier Locator Network Protocol              |
| DDNS        | Dynamic Domain Name Service                      |
| ID-LOCs     | Identifier-locators                              |
| m-DDNS      | mobile DDNS m-DDNS                               |
| HHR         | Home Host Register                               |
| VHR         | Visiting Host Register                           |
| IETF        | Internet Engineering Task Force                  |
| SBR         | Site Border Router                               |
| LLOC        | Local Locator                                    |
| GLOC        | Global Locator                                   |
| AR          | Access Router                                    |
| LMT         | Local Mapping Table                              |
| LD          | Link Detected                                    |
| CTO         | Control. Traffic Overhead                        |
| TTD         | Total Transmission Delay                         |
| BUD         | binding update Delay                             |
| BQD         | Binding Query Delay                              |
| DDD         | Data Delivery Delay                              |
| HOD         | Handover Delay                                   |

**References**

1. Gohar, M.; Jung, H.; Koh, S.J. Distributed Mapping Management of Identifiers and Locators in Mobile-Oriented Internet Environment. *Int. J. Commun. Syst.* 2014, 27, 95–115.
2. Morgan Stanley Report, Internet Trends. 2010. Available online: http://www.morganstanley.com/ (accessed on 28 September 2019).
3. Meyer, D.L.; Zhang, K. Fall, Report from the IAB Workshop on Routing and Addressing; IETF: Fremont, CA, USA, 2007; RFC 4984.
4. Atkinson, R.J.; Bhatti, S.N.; Andrews, S.T. Identifier-Locator Network Protocol (ILNP) Architecture Description; IETF: Fremont, CA, USA, 2012; RFC 6740.
5. Atkinson, R.J.; Bhatti, S.N.; Andrews, S.T. Optional Advances Deployment Scenarios for the Identifier-Locator Network Protocol (ILNP); IETF: Fremont, CA, USA, 2012; RFC 6748.
6. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. DNS Resource Records for the Identifier-Locator Network Protocol (ILNP); IETF: Fremont, CA, USA, 2012; RFC 6742.
7. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. Mobility, multi-homing, localised addressing and security through naming. *Telecommun. Syst.* 2009, 42, 273–291.
8. Atkinson, R.J. An Introduction to the Identifier-Locator Network Protocol (ILNP). In Proceedings of the IEEE London Communications Symposium (LCS), London, UK, 14-15 September 2006.
9. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. Mobility as an Integrated Service Through the Use of Naming. In Proceedings of the 2nd ACM International Workshop on Mobility in the Evolving Internet (MobiArch), Kyoto, Japan, 27–30 August 2007.
10. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. A Proposal for Unifying Mobility with Multi-Homing, NAT, & Security. In Proceedings of the 5th ACM International Workshop on Mobility Management and Wireless Access (MobiWAC), Chania, Crete, 22 October 2007.
11. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. Mobility Through Naming: Impact on DNS. In Proceedings of the 3rd ACM International Workshop on Mobility in the Evolving Internet (MobiArch), Seattle, WA, USA, 22 August 2008.
12. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. Harmonized Resilience, Security and Mobility Capability for IP. In Proceedings of the 28th IEEE Military Communications Conference (MILCOM): Boston, MA, USA, 18–21 October 2009.
13. Rehunathan, D.; Atkinson, R.J.; Bhatti, S.N. Enabling Mobile Networks Through Secure Naming. In Proceedings of the 27th IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 16–19 November 2008.
14. Makaya, C.; Pierre, S. An analytical framework for performance evaluation of IPv6-based mobility management protocols. *IEEE Trans. Wirel. Commun.* 2008, 7, 972–983.
15. Choi, N.J.; Gohar, M.; Koh, S.J. Domain-based distributed identifier-locator mapping management in...
Internet-of-Things networks. *Int. J. Netw. Manag.* 2018, 28, e2035.

16. Kim, J.I.; Jung, H.; Koh, S.J. Mobile-Oriented Future Internet: Implementation and Experimentations over EU-Korea Testbed. *Electronics* 2019, 8, 338.

17. Gohar, M.; Ahmed, W.; Bashir, F.; Choi, J.G.; Koh, S.J. A hash-based distributed mapping control scheme in mobile locator-identifier separation protocol networks. *Int. J. Netw. Manag.* 2017, 27, e1961.

18. Atkinson, R.J.; Bhatti, S.N.; Stephen, H. Evolving the Internet Architecture Through Naming. *IEEE J. Sel. Areas Commun.* 2010, 28, 1319–1325.

19. IEEE 802.21. *Local and Metropolitan Area Networks: Media Independent Handover (MIH)*; IEEE: Piscataway, NJ, USA, 2006.

20. Jung, H.; Gohar, M.; Kim, J.I.; Koh, S.J. Distributed mobility control in proxy mobile IPv6 networks. *IEICE Trans. Commun.* 2011, 94, 2216–2224.

21. Gohar, M.; Koh, S.J. A distributed mobility control scheme in LISP network. *Wirel. Netw.* 2014, 20, 245–259.

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).