A novel architecture of Proxy-LMA mobility management scheme for software-based smart factory networking

Chulhee Cho | Tai-Myoung Chung

College of Information and Communications Engineering, Sungkyunkwan University, Suwon 440-745 Kyunggi-do, Republic of Korea

Correspondence
Tai-Myoung Chung, College of Information and Communications Engineering, Sungkyunkwan University, Suwon, Kyunggi-do 440-745, Republic of Korea. Email: tmchung@ece.skku.ac.kr

Funding information
Institute for Information & communications Technology Promotion(IITP), Grant/Award Number: MSIP-0184-15-1003

Summary
Mobile users expect a network service, in which seamless handoff occurs while moving on a next generation wireless network. In addition, in smart factories (SFs), communication is required between factory floor and manufacturing zone, as well as connectivity towards office IT, or remote production facilities that are connected via wide area network or internet. For this purpose, interworking between heterogeneous networks is important, but there has been little research on global mobility support. Therefore, this paper proposes Proxy-LMA technology, a mobile IP-based global internetworking system, to improve global mobility and interoperability in the SFs network environment. The purpose of the proposed Proxy-LMA system is to support global mobility by using mobility management protocols such as PMIPv6 and MIPv6 in heterogeneous network environment. As a result of the performance evaluation, Proxy-LMA system is more efficient than other methods in terms of signaling cost and response delay in heterogeneous network environment. Software-based networking in SFs enables them to easily adapt the communication network to changing requirements. Similar to cloud-based systems, such SFs could be seen as production clusters that could be rented and configured as needed. The SF network uses software-defined networking combined with network functions virtualization, to achieve the required flexibility. Despite the fact that the technology is nowadays not yet ready for deployment in today's manufacturing networks, a novel network architecture for SFs based on software-defined networking and network virtualization is here proposed, to support smart services, especially for Industrie 4.0.

KEYWORDS
global roaming, mobility management, Proxy-LMA, smart factory (SF), software-defined networking (SDN), network functions virtualization (NFV)
1 INTRODUCTION

Internet communication users demand fast and high quality services. Therefore, internet communication service providers are fiercely competing to provide faster network services. To meet the needs of users, it is essential to provide fast and high-quality services in a heterogeneous network environment. During the last few years, the internet has increasingly evolved to become a critical infrastructure of our society. It has found its way into our daily life, be it private or in business, eg, through social networks, IP telephony, video streaming, cloud applications, and the like. Along these lines, it can be envisioned that, in the near future, almost everything will be networked and connected to and by the internet. This way, the Internet of Things will more and more become reality. The digital and analog worlds will become increasingly interwoven, and so-called smart environments will emerge, which promise to improve our quality of life and to enable a more economic usage of our resources. Emerging smart factories (SFs) are envisioned to be seamlessly integrated into such smart environments of the future. Consequently, production, networking, and computing will become tightly integrated. Cooperation among different sites of a factory, or even different factories, will be easily possible. It is envisioned that SFs will be highly modularized and to a large extent driven by IT services. For example, it is expected that they will be largely self-organized, to adapt themselves efficiently to frequently changing requirements. The resulting possibilities for flexible and timely manufacturing have the potential for innovation and may lead to notable changes in manufacturing practices. To achieve the aforementioned objectives, future SFs require, among other things, appropriate IT networks that provide the envisioned flexibility, ie, they should be easily reconfigurable, and it should be possible to treat data streams in the network differently, according to their individual requirements. Further requirements include an easy way to manage the network, as well as high robustness and appropriate IT security considerations. Due to the integration of various wireless terminals, there is an increasing demand for seamless services at anytime, anywhere, and on the move. Thus, various mobility-related protocols have been suggested to enable services not only within a specific network but also among multiple protocols, backbones, different devices, and internet service providers. For this purpose, MIPv6-based mobility-related protocols have been actively discussed for the purpose of ensuring that services can be provided in a heterogeneous network environment. The MIPv6 protocol is a host-based mobility protocol, in which an mobile node (MN) directly exchanges messages with a home agent (HA) when performing a handoff. However, host-based mobility protocols have a problem signaling between the MN and the HA leading to an increased amount of resources used in the wireless section. To solve this problem, Proxy MIPv6 (PMIPv6), a network-based mobility protocol, has been proposed. PMIPv6 is more effective in terms of signal cost than MIPv6 because the proxy mobility agent in the related network domain performs mobility-related signaling instead of MN when the MN performs the handover. However, the local mobility anchor (LMA) defined in PMIPv6 does not efficiently handle external global mobility. Moving within a heterogeneous network requires more network traffic and signaling compared with moving in an existing domain. In addition, PMIPv6, the mobility management protocol recommended by the IETF, delivers only local mobility of the domain, not global mobility.

To solve this problem, we proposed the method that uses the Proxy-LMA gateway to guarantee global mobility between heterogeneous networks. The proposed Proxy-LMA scheme aims to maintain the characteristics of other networks while minimizing the global handover overhead and location management costs incurred when moving between heterogeneous networks. In other words, it minimizes the signal when handover occurs in a heterogeneous network.

However, it is required to have an easy cooperation among sites within factories and other factories recently. In other words, in SFs, communication is required between factory floor and manufacturing zone, as well as connectivity towards office IT, or remote production facilities that are connected via wide area network or internet. As the ideas of SFs are being constructed, which closely integrate production, networking, and computing, the method for supporting a global mobility from the perspectives of intellectualization and SF is reinterpreted. Thus, this paper proposes a novel architecture of Proxy-LMA mobility management scheme for software-based SF networking. The purpose of the Proxy-LMA gateway in the proposed architecture is to minimize the global handoff overhead and location management costs incurred between SF’s software-based networking. The proposed technique of network is an SDN-based factory core network, which comprises 3 blocks (The data plane, the SF control plane, and the network functions virtualization [NFV]). In this regard, the SF performs the role of Proxy-LMA as suggested in the existing technique, and the SDN switch performs the role of LMA in the existing proposed technique. Software-based networking in SFs enables them to easily adapt the communication network to changing requirements. The SF network uses software-defined networking (SDN) combined with NFV to achieve the required flexibility.

Software-defined networks and network virtualization provide a promising basis for flexible requirement-driven network configuration and operation. They are currently heavily discussed trends on the internet, but to some extent, they
are still in their infancy. However, in dedicated exemplary settings, they have already proven their tremendous potential, eg, in the context of cloud computing, and regarding the design of modern data centers, or mobile core networks. Industrial networking and upcoming SFs as outlined above could also profit from the higher flexibility provided by SDN. This could lead to tremendous improvements of manufacturing lines and factory floors and large changes to traditional practices in manufacturing. IT services may very well increasingly shape the design of future factories. It is our vision that software defined networks and virtualization are viable approaches that enable flexibility and cost-efficient resource sharing within and among SFs, leading to a new quality of manufacturing processes. They have the potential to change industrial networking in a manner similar to the changes that virtualization and cloud computing brought to the internet.

This paper is structured as follows: Section 2 discusses the research related to global mobility management. Section 3 explains the proposed technique for global mobility management; and in addition, Sections 3 and 4 evaluate the performance of the proposed technique through a mathematical approach and present the results. Finally, Section 5 draws the conclusion.

2 | BACKGROUND AND RELATED WORK

The mobility management scheme proposed in this paper is divided into location management and handoff management for each network device. In general, handoff management is defined as a technique that enables continuous service at the application layer, when the MN moves between domains or cells. A major advantage of SDN is its increased flexibility compared with traditional network designs. The following 2 subsections introduce the high level concepts behind SDN and network virtualization.

2.1 | Software-defined networking

The main concept behind SDN is a consequent decoupling of the control and data planes, as shown in Figure 1. Following the definition of the Open Networking Foundation, an architecture for SDN can be subdivided into 3 distinct layers—application plane, control plane, and data plane. The application plane consists of software programs that make use of a northbound interface exposed by an SDN controller to explicitly program the behavior of the network. Such programs (SDN applications) are responsible for tasks like routing, access control, load balancing, or network isolation. In fact, any kind of network control functionality can be implemented as a dedicated SDN application. The application developer is only restricted by the functionality provided by the northbound interface, and not, for example, by the innovation cycle of a hardware vendor. The control plane is represented by a logically centralized SDN controller, which can best be described as an operating system for SDN applications. Its main task is to establish and maintain an abstract global view of the current network and provide this view to SDN applications via the northbound interface. It is also responsible for translating the input of SDN applications (ie, commands received via the northbound interface) into instructions that can be understood by switches inside the physical infrastructure. The SDN controller runs on a powerful server and thus provides a higher flexibility for executing complex control tasks,
compared with switches or routers that are typically limited with respect to computational power and overall memory size. Note that while the control plane is logically centralized, the actual deployment inside the physical infrastructure can be distributed to realize scalability and robustness (eg, by using a federation of multiple controllers). Finally, the data plane comprises SDN-enabled switches that expose their capabilities to the SDN controller using a standardized, open, and vendor-neutral control-data-plane interface. The latter is often termed a southbound interface for simplicity.

The Open Networking Foundation defines the following tasks that are associated with the southbound interface: programmatic control of all forwarding operations, reporting of switch capabilities and statistics, and event notification. Control of forwarding operations in the data plane is realized by rules that are remotely installed into SDN switches by the SDN controller on behalf of SDN applications. Every rule consists of a set of match fields and a set of actions. The match fields can be seen as a filter function identifying all packets that belong to a specific flow, and the corresponding actions determine which commands are executed for all packets of that flow. Well-known southbound interfaces like OpenFlow20 provide packet header matching capabilities for many common internet protocols (IP/MAC source and destination addresses, TTL, DSCP, VLAN tag, and so on) and various actions, like sending a packet towards a specific egress port, performing header modifications, or enabling differentiated packet treatment to support Quality of Service. If an SDN switch has no match fields for an incoming packet, that packet is dropped or forwarded towards the SDN controller to decide on further treatment of the associated packet flow (eg, its individual route through the network). Within the traditional internet-based networking architectures, the routers are responsible for both control plane (ie, executing routing protocols) and data plane operations (ie, performing destination-based packet forwarding).21 They use distributed protocols in the control plane that are implemented by the vendor directly within the router hardware. A drawback is the standardized and rigid functionality, eg, always routing along shortest paths, thereby not allowing more flexible routing policies. Moreover, the distributed approach makes system behavior sometimes less predictable and deterministic. On the other hand, with SDN, all high level network control operations like routing are handled by the SDN controller and the SDN applications, while the hardware inside the data plane is only responsible for the actual packet treatment (eg, on switch x forward packet p to switch y using interface z). A main advantage is that this approach can use more sophisticated and individual forwarding strategies (eg, time-dependent routing) implemented in software. In addition, SDN provides network operators with easier and more efficient automated network management, due to open interfaces for remote programmability and fine-grained centralized network control. Because new functionality can be introduced by simply changing the applications, it also enables rapid prototyping and high flexibility, eg, by deploying applications that provide traffic engineering, monitoring, or support for certain aspects of security and dependability.22 A potential drawback of SDN is its limited scalability with respect to the number of individual flows, which might result in a high number of control messages between SDN switches and controller, or the limited number of flow forwarding entries supported by hardware switches. However, the expected number of flows within an SF networking environment seems not to be critical with respect to scalability limits.23

### 2.2 Network virtualization

In addition to SDN, the trend toward network virtualization also bears great potential to revolutionize the provisioning of networked applications in general. In the context of this paper, we focus on NFV. The core idea behind NFV is the virtualization of network services, eg, firewalls, gateways, and related functionality, which are normally realized using dedicated hardware systems inside the data path (also called middle boxes). With NFV, such services can be realized software based inside a virtual machine and provisioned on demand, thereby enabling high flexibility, similar to the resource pooling principle known from cloud computing. To use this flexibility, 2 additional components are required inside the software-defined network. First, an NFV infrastructure is needed where the virtualized network services can be deployed. This might be an edge cloud or a set of dedicated servers (commodity hardware) directly colocated with the physical networking infrastructure. Second, the control plane must be extended to support NFV orchestration, to handle placement and life cycle management of the virtual machines. Network functions virtualization provides advantages with respect to resource scalability and resilience, since the number of NFV instances can be scaled up and down, as required by the current situation. In particular, the functionality provided by virtualized services is no longer tied to a fixed position within the network topology, which greatly enhances the configurability of software-defined networks.

Figure 2 shows an architectural sketch of a network with explicit support for SDN and NFV. A joint control plane that consists of an SDN controller and an NFV orchestrator is responsible for controlling the network, configuring network services, and managing NFV instances. If a specific network service is requested, eg, NS1 in Figure 2, a new virtual machine with the required functionality is spawned on a compute node inside the NFV infrastructure. Note that the
logical placement of the service and the physical placement of the virtual machine can be controlled separately from each other. An SDN application inside the control plane then subsequently directs some or all of the flows in the network so that they have to pass the new virtual machine, before they can reach their original target (NFV redirection). If multiple network services are associated with one flow, service chaining in conjunction with packet attached meta-data (so-called network service headers) may be used to enforce all packets of a flow having to pass through a specific set of network services.

2.3 | Network-based mobility protocol

PMIPv6 is a network-based mobility support protocol that supports MN’s mobility, by allowing network side equipment to perform signaling for MN’s mobility support. The basic components of PMIPv6 consist of LMA, mobile access gateway (MAG), policy server, and MN. PMIPv6 is more effective in signal cost than MIPv6 because proxy mobility agent in the network domain performs mobility signaling. When the MN is turned on, or enters the PMIPv6 domain, the MAG recognizes that the MN is connected to the access link based on the information of the L2 layer and registers the location information in the MAG and the LMA in the local domain, to inform the current location of the MN.24 Through this process, the MAG obtains the MN-ID, which is the identifier of the MN. After the authentication process of the MN to the policy server based on the MN-ID, the MN-Profile, and the LMA Address. The MAG then sends a proxy binding update message with the MN-ID to the LMA to register the MN’s location on behalf of the MN. After receiving this information, the LMA generates the MN’s unique home network prefix (MN-HNP) and sends the proxy binding acknowledgment (PBA) to the MAG, together with the MN-HNP generated by the LMA, in response to the proxy binding update. The MAG receiving the PBA from the LMA transmits the MN-HNP included in the PBA to the MN in the router advertisement. All packets of the MN are then forwarded through the bidirectional tunnel created between the LMA and the MAG.

However, the proposed procedure is limited to the location domain of a particular network.18,25 Therefore, the global mobility of the user must be considered so that the various backbones, protocols, and services provided by the service provider can be used in the next generation wireless networks rather than using the internal roaming of a specific network.

2.4 | Support for global mobility using the MIH

Figure 3 shows an MIH-based heterogeneous network topology. The IEEE 802.21 WG was created in March 2004 to support seamless handover between heterogeneous networks and termed the technology MIH.26 The seamless mobility service provided through IEEE 802.21 MIH technology ensures that the quality of the service does not deteriorate, as the user satisfies the service level provided by the user in the previous network, when the user terminal performs handover between different heterogeneous networks.27 The IEEE 802.21 MIH technology defines a structure, service, and protocol.
procedure to optimize the performance of a user application service in a handover between heterogeneous networks, through a close linkage between the mobility management protocol and the physical and link layers.\textsuperscript{28} The MIH has an entity called MIH Function between Layers 2 and 3. This MIH Function entity transfers state information to the Layer 3 protocol independently of Layer 2. Therefore, if MIH is used, the mobility management protocol can be installed in the Layer 3, regardless of the heterogeneous access network.\textsuperscript{26,29}

MIH technology is currently being standardized, and it is not easy to implement authentication procedures and application services. Although much work has been done to provide a way to provide intact services for MN based on MIH, there is no definite definition and proper solution for handover between heterogeneous networks.

2.5 | Support for global mobility between MIPv6 and PMIPv6

PMIPv6 limits the mobility support scope to one non-global area, because the MN only supports mobility within one PMIPv6 domain. Therefore, it is necessary to link with global mobility support protocol such as MIPv6 for global mobility support.\textsuperscript{30} That is, it is possible to support the global mobility by the MIPv6 and to support the local mobility by the PMIPv6, depending on the location.\textsuperscript{31,32} To solve this problem, when Binding Cache Entry (BCE) of PMIPv6 and HA is shared, binding information should be managed regardless of domain. Since PMIPv6 uses network-based mobility management, it can support mobility regardless of MN; however, MIPv6 requires a host-based mobility stack in MN. To solve this problem, when BCE of PMIPv6 and HA are shared, binding information should be managed regardless of domain.

3 | THE PROPOSED PROXY-LMA APPROACH

In this paper, we propose a scheme of using a Proxy-LMA gateway between domains to provide interoperability and global mobility among heterogeneous networks when MN moves. In particular, Proxy-LMA gateways are set up to directly connect with adjacent subsystems, allowing access from both subsystems. Figure 4 shows an example of interconnection between the 3 subsystem networks.

The proposed mobility management scheme can be divided into location management and handoff management.\textsuperscript{16,17} In the case of PMIPv6, an entity on the network, rather than the MN, is responsible for location management, allowing the session to be transferred to the MN. The current location of the MN is stored in a database form in a network entity such as MAG and HA/LMA, and the location information is stored.\textsuperscript{24} Handoff management refers to a technology that enables the application layer service to be seamlessly performed when the MN moves between domains. The proposed Proxy-LMA gateway acts as a gateway to perform signaling message conversion between cells, to facilitate protocol conversion. To do this, it supports conversion to 2 interfaces. Initially, the Proxy-LMA gateway determines the MN’s ID information (MN-ID) and requests registration if it moves between different domains with the expected movement path.

The conversion function of the Proxy-LMA gateway is as follows:
Maintains message conversion and signal format between heterogeneous networks.

Maintains the location information of the mobile users (MUs) after the location registration or update and provides the function of searching the location information at any time.

Transmits the session and transmits the signaling information securely.

To ensure the transmission of secure data, MUs perform authentication.

Furthermore, the proposed Proxy-LMA gateway is a high-level entity on an adjacent network and provides interworking with MIP-based services. This service, also based on location area, uses many databases to manage global mobility between heterogeneous networks. In general, HA/LMA constitutes a network with many MAGs. The operation scenario of the Proxy-LMA gateway is as follows: MUs can move to multiple subsystems implemented in heterogeneous network environments with different technologies and protocols. Each subsystem uses its own registration procedure stored in the entity of each network, which stores the information form and manages the mobility. The Proxy-LMA gateway based on such information performs a procedure for message transmission between heterogeneous networks when the MN moves. It also adjusts the format to enable retrieval of user roaming information and also controls errors to ensure secure transmission. In this case, single registration (SR) and multiple registration (MR) procedures may be considered for registration. An SR method is to register with a single system, HA/LMA or MAG, to show the MN’s current location at any time. It is a method of registering the current location of the MN at any time to an entity existing in many subsystems, without each subsystem managing the mobility of the integrated MN. However, this registration procedure can affect the quality of network services, because of the excessive traffic generated by each entity in the network.

Figure 5 shows the procedures associated with the proposed registration process. The Proxy-LMA gateway defines the maximum segment size that can be converted without changing the underlying network characteristics, to flexibly convert the message size.

Figure 6 shows the procedure of global mobility in a heterogeneous network. Unlike the conventional tunneling method, there is no problem due to the constant tunneling; and if the message path is optimized, the transmission cost does not increase greatly, even if the handover continues.
Figure 7 shows the envisioned framework, which relies on an SDN-based factory core network, which implements 3 important building blocks: the data plane, Proxy-LMA acting as the SF control plane, and NFV. The data plane consists of LMAs that act as SDN switches and provides the fundamental functionality of packet treatment within the network. This provides the required agility of SF networks, by enabling a rule-based control of individual data flows.

The network and related services are managed by the (logically) centralized Proxy-LMA. In its role as SDN controller, it is responsible for managing data flows according to changing requirements. The Proxy-LMA also orchestrates the services provided by NFV, in terms of deployment, configuration, and monitoring. Furthermore, it provides an abstract management interface to the manufacturing execution system and implements security policies. The network itself is complemented by NFV. Network functions virtualization enables the Proxy-LMA to deploy virtualized services on demand. These services could be network related, but might also be directly associated with the manufacturing process, eg, in terms of computationally intensive tasks, like image processing. In contrast to a traditional device-bound deployment, virtualized services can be instantiated, replaced, or destroyed dynamically, according to changing requirements.

Figure 5 Location update procedures of Proxy-LMA architecture. BS, base station; HA, home agent; LMA, local mobility anchor; MAG, mobile access gateway; MN, mobile node; WLAN, wireless local area network

Figure 6 Proxy-LMA handover. AP, access point; BS, base station; HA, home agent; LMA, local mobility anchor; MAG, mobile access gateway; MN, mobile node; PBA, proxy binding acknowledgment; PBU, proxy binding update; RA, router advertisement; RS, router solicitation

CHO AND CHUNG
requirements, eg, to scale the number of active instances, or change their location within the network topology. While the software-based core network replaces most of the traditional switches and routers, we envision a separation of concerns regarding hard real-time communication and safety. This means that already established and approved technologies, like Ethernet based fieldbuses, will continue to be used, especially for scenarios that have stringent safety requirements, and depend on deterministic time synchronization. In the long term, such special purpose domains might be complemented or replaced by software-based manufacturing networks that can provide real-time properties (eg, as described in Nayak et al[34]). Both technologies can coexist within our framework (Figure 7). However, other network traffic outside of the special purpose domains is completely under the control of the Proxy-LMA. This includes, for example, the communication between factory floor and manufacturing zone, as well as connectivity towards office IT, or remote production facilities that are connected via wide area network or internet. These traffic flows are either configured according to actual demands, eg, to execute an order scheduled by the manufacturing execution system, or preconfigured, in terms of administrative policies. Thereby, the Proxy-LMA incorporates different Quality of Service requirements, like bandwidth, latency, or jitter, into its routing and scheduling decisions. It also configures and enforces security policies, by permitting or prohibiting individual data flows inside or across virtual network and security domains. If a fine-grained control of network traffic between domains is desired, virtualized firewalls or deep packet inspection instances are deployed and configured by the Proxy-LMA using NFV. These instances screen and filter network traffic originating from possibly untrusted sources. In our scenario, an industrial edge cloud colocated with the production facility provides the resources for virtualized services. Besides the security functionality like firewalls and deep packet inspection mentioned previously, the NFV infrastructure can also be used to perform production-related tasks, eg, data processing and filtering.

4 | PERFORMANCE EVALUATION

We define a mobility model that considers the moving speed of the MU, the direction of movement, and the user’s movement record to perform the query execution and the information update rate.
4.1 Parameter

This section defines the 2 adjacent networks as subsystem \( i \) and subsystem \( j \) for the evaluation of performance and assumes that the Proxy-LMA is located between those heterogeneous networks. Table 1 shows the definitions of the parameters.

4.2 Evaluation of query and update rates

In global roaming, signals such as registration, cancellation, transmission request, and transmission request delivery are generated. To process this signal, query execution and updates of information are performed on the database. We define a mobility model that considers the moving speed of the MU, the direction of movement, and the user’s movement record; performs query execution and evaluation of the data update rate. Assuming that the MU of the subsystem \( i \) is uniformly distributed in the moving direction \([0, 2\pi]\), the average speed is represented as \( \nu_i \), and the ratio of the MU moving from subsystem \( i \) to subsystem \( j \) is expressed as follows, where \( \rho_i \) is the user density in subsystem \( i \) and \( L_i \) is the boundary of subsystem \( i \).

\[
rij = \frac{\rho_i \nu_i L_i}{\pi} \quad (1)
\]

By modeling each network entity (ie, MAG, HA/LMA, or Proxy-LMA) as an M/G/1 queuing system, we can determine the average query and update rates at each database level. In particular, for the SR method, we use the sequence of operations executed during intersystem roaming and reported in Vivaldi et al to evaluate for each subsystem the average arrival rates at each MAG, HA/LMA, or Proxy-LMA, as follows:

\[
\tau_{SR\_req\_MAGx} = N_{MAGi} [r_{ji} + (\lambda_{outi} + \lambda_{ini}) \rho_i A_i] \quad (2)
\]

\[
\tau_{SR\_update\_MAGx} = N_{MAGi} (r_{ji} + r_{ji}) \quad (3)
\]

\[
\tau_{SR\_req\_LMAx} = N_{MAGi} [r_{ji} + (\lambda_{outi} + \lambda_{ini}) \rho_i A_i] \quad (4)
\]

\[
\tau_{update\_MAGx} = N_{MAGi} (r_{ji} + r_{ji}) + 2N_{MAGi} r_{ji} \quad (5)
\]

\[
\tau_{SR\_update\_LMAx} = N_{MAGi} [r_{ji} + \lambda_{ini} \rho_i A_i] \quad (6)
\]

| TABLE 1 | Parameter definitions for each subsystem \( i \) |
|----------|------------------------------------------------|
| Parameter | Definition |
| \( N_{MAGi} \) | Number of \( MAG_i \) in subsystem \( i \) |
| \( \rho_i \) | User density in subsystem \( i \) |
| \( \nu_i \) | Average speed in \( MAG_{ix} \) \( \forall x=1,..,N_{MAGi} \) |
| \( A_i \) | Total area of subsystem \( i \) |
| \( \lambda_{outi} \) | Average traffic generated by each MT of subsystem \( i \) (calls/second/terminal) |
| \( \lambda_{ini} \) | Average incoming traffic to each MT of subsystem \( i \) (calls/second/terminal) |
| \( \tau_{M\_req\_e} \) | Query rate at the functional entity \( e \) using method \( M \) |
| \( \tau_{M\_update\_e} \) | Update rate at the functional entity \( e \) using method \( M \) |
\( \tau_{SR}^{\text{update}_LMA_i} = N_{MAG}(r_{ij} + r_{ji}) \)  

(7)

\( \tau_{\text{req}_LMA_i}^{SR} = N_{MAG}r_{ij} + N_{MAG}\left[r_{ij} + \lambda_{inj}A_j\right] \)  

(8)

\( \tau_{\text{update}_LMA_i}^{SR} = N_{MAG}(r_{ij} + r_{ji}) + N_{MAG}r_{ji} \)  

(9)

\( \tau_{\text{req}_G}^{SR} = N_{MAG}\left(r_{ij} + \lambda_{inj}A_j\right) + N_{MAG}\left(r_{ji} + \lambda_{inj}A_i\right) \)  

(10)

\( \tau_{\text{update}_G}^{SR} = N_{MAG}(r_{ij} + r_{ji}) + N_{MAG}\rho_jA_j(\lambda_{outj} + \lambda_{inj}) \),  

(11)

where \( A_i \) is the total area of subsystem \( i \) and \( N_{MAG} \) and \( N_{MAG_j} \) represent the number of MAG in subsystems \( i \) and \( j \), respectively. In addition, the query and update rates in the database for the MR method can be obtained through the algorithm presented in Weniger and Velev.30 The average arrival rate of query/update at each network entity is calculated as follows:

\( \tau_{\text{req}_MAG_i}^{MR} = N_{MAG}\left[r_{ij} + (\lambda_{outi} + \lambda_{ini})\rho_iA_i\right] + N_{MAG}\lambda_{inj}\rho_jA_j \)  

(12)

\( \tau_{\text{update}_MAG_i}^{MR} = N_{MAG}r_{ji} \)  

(13)

\( \tau_{\text{req}_MAG_i}^{MR} = N_{MAG}\left[2r_{ij} + (\lambda_{outj} + \lambda_{ini})\rho_jA_j\right] \)  

(14)

\( \tau_{\text{update}_MAG_i}^{MR} = N_{MAG}r_{ji} + 2N_{MAG}r_{ji} \)  

(15)

\( \tau_{\text{req}_LMA_i}^{MR} = N_{MAG}\left[r_{ji} + \lambda_{inj}\rho_iA_i\right] + N_{MAG}\lambda_{inj}\rho_jA_j \)  

(16)

\( \tau_{\text{update}_MAG_i}^{MR} = N_{MAG}r_{ji} \)  

(17)

\( \tau_{\text{req}_LMA_i}^{MR} = N_{MAG}\left[r_{ji} + \lambda_{ini}\rho_jA_j\right] \)  

(18)

\( \tau_{\text{update}_LMA_i}^{MR} = N_{MAG}r_{ji} + N_{MAG}r_{ji} \)  

(19)

\( \tau_{\text{req}_G}^{MR} = N_{MAG}\lambda_{inj}\rho_jA_j + N_{MAG}\left(r_{ji} + \lambda_{inj}\rho_iA_i\right) \)  

(20)

\( \tau_{\text{update}_G}^{MR} = N_{MAG}r_{ji} + N_{MAG}\rho_jA_j(\lambda_{outj} + \lambda_{inj}) \)  

(21)

For the proposed approach, query and update arrivals at each database also form a Poisson process. According to the operation shown in Figure 7, the average arrival rate of each network entity is calculated as follows:

\( \tau_{\text{req}_MAG_i}^{NM} = r_{ji} + (\lambda_{outi} + \lambda_{ini})\rho_iA_i \)  

(22)

\( \tau_{\text{update}_MAG_i}^{NM} = r_{ij} + r_{ji} \)  

(23)
\[ \tau_{\text{req}, \text{MAG}_j} = r_{ij} + (\lambda_{out_j} + \lambda_{in_j}) \rho_j A_j \] (24)
\[ \tau_{\text{update}, \text{MAG}_j} = r_{ji} + r_{ji} \] (25)
\[ \tau_{\text{req}, \text{LMA}_i} = r_{ji} + \lambda_{in_i} \rho_i A_i \] (26)
\[ \tau_{\text{update}, \text{LMA}_i} = r_{ji} + r_{ji} \] (27)
\[ \tau_{\text{req}, \text{LMA}_j} = r_{ij} + \lambda_{in_j} \rho_j A_j \] (28)
\[ \tau_{\text{update}, \text{LMA}_j} = r_{ij} + r_{ji} \] (29)
\[ \tau_{\text{req}, G} = r_{ij} + r_{ji} + \lambda_{in_i} \rho_i A_i + \lambda_{in_j} \rho_j A_j \] (30)
\[ \tau_{\text{update}, G} = r_{ij} + r_{ji}. \] (31)

### 4.3 Measuring of the response time

Through the above expressions (2) to (31), it is possible to calculate the average arrival ratio \( \lambda_e \) of messages in each network entity \( e \). Suppose that the probability that the message, which arrived at network entity \( e \), will be the information renewal message is \( P_{\text{update}, e} \) and that the probability that this message will be the query execution message is \( P_{\text{req}, e} \). Then the probability \( P_{\text{req}, e} \) can be expressed as below, when the M method is used.

\[ P_{\text{req}, e}^M = 1 - P_{\text{update}, e}^M = \frac{\tau_{\text{req}, e}^M}{\tau_{\text{req}, e}^M + \tau_{\text{update}, e}^M} \] (32)

Here, M shows the proposed method (ie, SR, MR, or NM), and \( e \) shows the functional entity (ie, MAG, HA/LMA, or Proxy-LMA). As each entity was modeled through the M/G/1 queuing system, the response time for the query execution or information renewal can be checked. To obtain better information, HA/LMA is assumed to be caching the data replicated from MAG in the environment with the probability of \( P. \) As a result, the effect on global roaming can be analyzed when various probability \( P \) is applied. Assuming that the average delay time for the query execution or information renewal processing in each network entity is known and assuming that the average delay time for query or update processing in each network entity is \( T_{\text{pr}, e} \) or \( T_{\text{pr}, \text{update}, e} \), respectively, the average delay time for processing the message in an entity \( e \) can be expressed as follows:

\[ E(T_{\text{pr}, e}) = P_{\text{req}, e} T_{\text{pr}, e} + P_{\text{update}, e} T_{\text{pr}, \text{update}, e} \] (33)

Then the processing delay at the second time point is calculated as follows:

\[ E(T^2_{\text{pr}, e}) = P_{\text{req}, e} T^2_{\text{pr}, e} + P_{\text{update}, e} T^2_{\text{pr}, \text{update}, e} \] (34)

Using the Pollaczek-Kintchine formula, the delay time in processing each network entity can be expressed as follows:

\[ W_e = \frac{\lambda_e E(T^2_{\text{pr}, e})}{2[1 - \lambda_e E(T_{\text{pr}, e})]} \] (35)
where \( \lambda_e \) is the average arrival rate of the message at network entity \( e \) and the response time for the query depends on \( P \), the probability that the requested information is in the MAG. \( T_x \) is represented by assuming the delay in the transmission between \( MAG_{ix} \) and \( LMA_i \) as constant, where \( T_{pr.MAG_{ix}} \) and \( T_{pr.LMA_i} \) represent the average processing delay in \( MAG_{ix} \) and \( LMA_i \), respectively, and \( W_{MAG_{ix}} \) and \( W_{LMA_i} \) represent the average waiting delay in \( MAG_{ix} \) and \( LMA_i \), respectively. The response time for location information utilization and mobility related signals is calculated as a weighted sum of query performance in \( MAG_{ix} \) and \( LMA_i \). Therefore, the response time from subsystem \( i \) is calculated as follows:

\[
T_{res_i} = P(W_{MAG_{ix}} + T_{pr.MAG_{ix}}) + (1-P)(2T_x + W_{LMA_i} + T_{pr.LMA_i}).
\] (36)

Here, \( T_{res_i} \) is the average delay time in responding to the query when the MU is located at subsystem \( i \).

### 4.4 Numerical results and analysis

The analysis evaluates the number of expected signaling messages per second exchanged at the M-LMA/M-HA or Proxy-LMA level and the average response time for queries in subsystem \( i \) to analyze signaling traffic due to global roaming. First, we investigated the effect of user behavior on query/update rates on M-LMA/M-HA or Proxy-LMA. The behavior of the MU in global movement is characterized by global call-to-mobility ratio (GCMR). Table 2 summarizes the parameters of the respective subsystems. In general, for any method, as the GCMR increases, the query and update rate of each entity decreases. In other words, in the case of a fixed call rate, as the number of times the MU performs intersystem roaming increases, more queries/updates are required per unit of time. Figures 8 and 9 show that the proposed scheme performs at a lower rate than the SR and MR schemes in terms of queries/updates generated during global roaming.

| Parameter                        | Subsystem \( i \)          | Subsystem \( j \)          |
|----------------------------------|----------------------------|----------------------------|
| Number of MAGs                   | 6                          | 4                          |
| Cell area (km\(^2\))            | 0.04                       | 36.0                       |
| Average speed (km/h)             | 5.0                        | 20.0                       |
| Number of HA/LMA                 | 1                          | 1                          |
| \( \lambda_{in}(\text{call, second, terminal}) \) | \( 8.333 \times 10^{-4} \) | \( 5.556 \times 10^{-5} \) |
| \( \lambda_{out}(\text{call, second, terminal}) \) | \( 5.556 \times 10^{-4} \) | \( 2.778 \times 10^{-4} \) |

**FIGURE 8** Impact of user behavior on query rate. GCMR, global call-to-mobility ratio; LMA, local mobility anchor; MR, multiple registration; SR, single registration
We also investigated the average query and update rate and system response time for M-LMA/M-HA or Proxy-LMA levels for each method according to the distribution of MUs. Figures 10 and 11 show that as the percentage of MUs in subsystem $i$ increases, the average query and update rate and system response time at each entity decreases. This means that if MUs are concentrated in a subsystem, better services can be provided. In addition, the proposed scheme reduces the query/update rate at the entity as well as the response time, as opposed to the SR and MR methods. Figure 12 shows that the proposed approach significantly reduces the response time, as well as the query execution and information renewal rate, in M-LMA/M-HA or Proxy-LMA.

To investigate the change in response times according to the use of location information, we changed the probability $P$ of using the necessary information in the MAG after the query. As a result, the response time decreased as the probability $P$ increased, as shown in Figure 13. It can be seen that the response time on the network becomes faster as the location information is retrieved at the MAG level after query execution. In conclusion, we have confirmed that the proposed approach for all values of $P(0<P<1)$ can shorten the response time.

4.5 | Discussion on interoperability for software-based SF networking

We believe that the concepts of SDN and network virtualization outlined above can help to solve the challenges of future factory networking. As an actively managed infrastructure, software-defined networks can be adopted to the

![Figure 9](image1.png) Impact of user behavior on update rate. GCMR, global call-to-mobility ratio; LMA, local mobility anchor; MR, multiple registration; SR, single registration

![Figure 10](image2.png) Impact of user distribution on the query rate. LMA, local mobility anchor; MR, multiple registration; MU, mobile user; SR, single registration
changing demands of SFs. In the following, we present our assumptions regarding key aspects and challenges of SF networks and discuss the way in which our software-defined framework addresses these challenges. We envision SFs to be very heterogeneous in terms of machinery, vendors, and equipment types, as well as in terms of data representations, communication protocols, and communication requirements. Industrial networks will become larger and more heterogeneous in terms of both communication requirements and deployed technologies. In addition, the devices themselves will become more heterogeneous, which makes device management especially challenging. Legacy systems will accompany the transformation to SFs for a significant amount of time. We do not expect the large amount of different, often very specialized, automation protocols and standards to converge into a single one soon, if at all. Instead, we assume that future network architectures and frameworks might integrate a wide range of existing solutions. Thus, interoperability with industrial communication architectures and protocols, such as Ethernet-based fieldbuses, or OPC UA as protocol for machine-to-machine communication, is expected to be an important aspect of SF networks. Considering these assumptions, the challenges that need to be addressed are twofold: On the one hand, SF networks must enable connectivity across a wide range of communication technologies and standards. Thus, the network must both provide the physical hardware to interface with the different deployed technologies, while also acting as an agent between devices using incompatible protocols. On the other hand, device management must be addressed as well, to provide

![Impact of user distribution on response time](image1.png)

**FIGURE 11** Impact of user distribution on the update rate. LMA, local mobility anchor; MR, multiple registration; MU, mobile user; SR, single registration

![Impact of user distribution on response time](image2.png)

**FIGURE 12** Impact of user distribution on response time. LMA, local mobility anchor; MR, multiple registration; MU, mobile user; SR, single registration
an abstracted management interface to the manufacturing execution system. The network itself also requires such facilities, to perform configuration changes during its adaption process. The flexibility of the SDN paradigm enables SF networks to cope with different protocols and standards. The Proxy-LMA can be easily extended to support new protocols by means of the software modules provided by different vendors, whereas LMA with the SDN switch role need to implement approaches like protocol oblivious forwarding. Virtualized gateways deployed on demand (by using NFV) enable devices using incompatible protocols or standards to communicate with each other. Due to standardized open interfaces for remote programmability, the SDN paradigm provides a foundation for easy to use and efficient, automated network management tools. Within a SF, automatic software-based mass configuration and management of networking devices is possible, which might be less prone to misconfigurations caused by human failure, or due to the usage of different (vendor specific or communication protocol dependent) management systems. Furthermore, it makes it also easier to integrate devices and systems of different vendors. Because software-based networks provide sophisticated tools for traffic engineering (based on the global network view) and fine-grained centralized network control by default, the factory should be able to easily cope with such dynamics.

5 | CONCLUSIONS

In this paper, we describe our vision of a framework for software-based SFs based on current trends in internet-based networking, like software-defined networks and network virtualization. The framework for SDN in SFs as outlined in this paper can provide sufficient flexibility to deal with the increased production agility anticipated for the factory of the future. Standardized open interfaces and remote programmability of forwarding devices inside the core network can be used to cope with the trends for increased and more heterogeneous industrial networking.

So far, most studies on NGNW have only considered domains within one mobility management protocol. However, there are few studies on heterogeneous network environment, signaling message exchange procedure, and cost reduction. The heterogeneous environment discussed in this paper has many limitations. In particular, some problems have been found, such as limit of mobility, increase of signaling cost, and handover delay. To solve these problems, we propose the PMIPv6 and MIPv6-based Proxy-LMA gateway method and reduce signal cost and handover delay. In this paper, we investigated the performance of proposed Proxy-LMA gateway method, SR and MR techniques for network traffic reduction, fast response time generation, fast user location, quick query execution, and information update. As a result of the performance evaluation, it was confirmed that the proposed Proxy-LMA gateway method is more advantageous than SR, MR, and others. Therefore, it is confirmed that the proposed method can guarantee global mobility and interoperability in the heterogeneous network environment. The proposed technique supports global mobility from the perspectives of intellectualization and SFs. Therefore, it enables facilitating the cooperation among sites within a factory and other factories. It contributes to the establishment of SFs wherein production, networking, and computing are closely integrated. The future research plan is to verify the accuracy of the simulation results through the network.
simulator and the results of the research through the mathematical analysis and to study various control messages efficiently in the handover between heterogeneous networks.

ACKNOWLEDGEMENTS

This work was supported by Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (no. B0184-15-1003, oneM2M The Development of oneM2M Conformance Testing Tool and QoS Technology). This research was supported by the MIST (Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information and Communications Technology Promotion) (2015-0-00914).

ORCID

Chulhee Cho http://orcid.org/0000-0002-2915-2572

REFERENCES

1. Kovatsch M, Lanter M, Duquennoy S, eds.. Actinium: a restful runtime container for scriptable Internet of Things applications. Internet of Things (IOT). In: 2012 3rd International Conference on the Internet of Things (IOT); October 2012; Wuxi, China:135-142.
2. Bormann C, Castellani AP, Shelby Z. CoAP: An application protocol for billions of tiny internet nodes, Vol. 16; 2012:62-67.
3. Moritz G, Golatowski F, Lerche C, Timmermann D. Beyond 6LoWPAN: web services in wireless sensor networks. IEEE T Ind Inform. 2013;9(4):1795-1805.
4. Guo H, Wang L, Duan B, Liu C, Liu Z. LMA/HA discovery mechanism on the interaction between MIPv6 and PMIPv6. In: 2009 WiCom’09 5th International Conference on 2009 Wireless Communications, Networking and Mobile Computing; 2009; Beijing, China:1-4.
5. Gundavelli S, Leung K, Devarapalli V, Chowdhury K, Patil B. Proxy mobile IPv6. IETF RFC 5213; August 2008.
6. Lee J-H, Ernst T, Chung T-M. Cost analysis of IP mobility management protocols for consumer mobile devices. IEEE T Consumer Electr. 2010;56(2):1010-1017.
7. Liao J, Qi Q, Li T, Cao Y, Zhu X, Wang J. An optimized QoS scheme for IMS-NEMO in heterogeneous networks. Int J Commun Syst. 2012;25:185-204.
8. Cho C, Cho JD, Jeong J. Cost-effective Proxy-LMA mobility management scheme in mobile networks with global mobility support; 2014:121-127.
9. Gandhi R, Liu HH, Hu YC, et al. Duet: cloud scale load balancing with hardware and software. ACM SIGCOMM Comput Commun Rev. 2014;44(4):27-38.
10. Hong C-Y, Kandula S, Mahajan R, et al. Achieving high utilization with software-driven WAN. ACM SIGCOMM Comput Commun Rev. 2013;43(4):15-26.
11. Jain S, Kumar A, Mandal S, et al. B4: experience with a globally-deployed software defined WAN. ACM SIGCOMM Comput Commun Rev. 2013;43(4):3-14.
12. Basta A, Kellerer W, Hoffmann M, Morper HJ, Hoffmann K. Applying NFV and SDN to LTE mobile core gateways, the functions placement problem. Chicago, Illinois, USA; 2014:33-38.
13. Girbea A, Suciu C, Nechifor S, Sisak F. Design and implementation of a service-oriented architecture for the optimization of industrial applications. IEEE Trans Ind Inf. 2014;10(1):185-196.
14. Eliasson J, Delsing J, Derhamy H, Salicz C, Wang K. Towards industrial internet of things: an efficient and interoperable communication framework. In: 2015 IEEE International Conference on Industrial Technology (ICIT); March 2015; Seville, Spain:2198-2204.
15. Bauer R, Bless R, Haas C, Jung M, Zitterbart M. Software-based smart factory networking. Automatisierungstechnik 2016. 2016;64(9):765-773.
16. Wang W, Akyildiz IF. A new signaling protocol for intersystem roaming in next-generation wireless systems. IEEE J Sel Areas Commun. 2001;19:2040-2052.
17. Park KI, Lin Y-B. Reducing registration traffic for multtier personal communications services. IEEE Trans Veh Technol. 1997;46(3):597-602.
18. Markouliadakis JG, Lyberopoulos GL, Tsirkas DF, Sykas ED. Evaluation of location area planning scenarios in future mobile telecommunication systems. Wir Nettw. 1995;1(1):17-28.
19. Open Networking Foundation. SDN Architecture Overview, Version 1.0. Technical Report; 2013.
20. McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Comput Commun Rev*. 2008;38(2):69-74.

21. Gruian F, Roop P, Salicic Z, Radijovic I. The SystemJ approach to system-level design. In: Formal Methods and Models for Co-Design, Fourth ACM and IEEE International Conference on MEMOCODE '06. July 2006; Napa, CA, USA:149-158.

22. Kreutz D, Ramos F, Verissimo P, Rothenberg CE, Azodolmolky S, Uhlig S. Software-defined networking: a comprehensive survey. *Proc IEEE*. 2015;103(1):14-76.

23. Kyusakov R, Makitaavola H, Delsing J, Eliasson J. Efficient XML interchange in factory automation systems. In: IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society; Nov. 2011; Melbourne, VIC, Australia:4478-4483.

24. Jain R, Lin Y-B, Lo C, Mohan S. A caching strategy to reduce network impacts of PCS. *IEEE J Sel Areas Commun*. 1994;12(8):1434-1444.

25. Safa H, Pierre S, Conan J. A new location management strategy for IS-41-based mobile networks. In: Proceedings of the Twelfth Annual International Conference on Wireless Communications, ‘Wireless 2000’; July 2000; Calgary, Canada:295-301.

26. IEEE 802.21 MIH Protocol. http://www.ieee.org.

27. Chen JL, Liu SW, Wu SL, Chen MC. Cross-layer and cognitive QoS management system for next-generation networking. *Int J Commun Syst*. 2011;24(9):1150-1162.

28. McNair J, Akyildiz IF, Bender MD. An inter-system handoff technique for the IMT-2000 system. In: IEEE Infocom 00; March 2000; Tel Aviv, Israel:208-216.

29. Magagula LA, Falowo OE, Chan HA. PMIPv6 and MIH-enhanced PMIPv6 for mobility management in heterogeneous wireless networks. In: AFRICON 2009; Sept. 2009; Nairobi, Kenya:23-25.

30. Weniger K, Velev G. PMIPv6-MIPv6 Interactions, proxy mobile IPv6 and mobile IPv6 interworking issues. draft-weniger-netlmm-pmipv6-mipv6-issues-00; 2007.

31. Jung H. Fast handover for hierarchical MIPv6 (F-HMIPv6). draft-jung-mobopts-flmipv6-00; 2005.

32. Wakikawa R, Gundavelli S IPv4 support for proxy mobile IPv6. draft-ietf-netlmm-pmipv6-ipv4-support-09; January 2010.

33. Vivaldi I, Habaebi MH, Ali BM, Prakash V. Fast handover algorithm for hierarchical mobile IPv6 macro-mobility management. In: APCC 2003, Vol. 2; Sept. 2003; Penang, Malaysia, Malaysia:630-634.

34. Nayak NG, Durr F, Rothermel K. Software-defined environment for reconfigurable manufacturing systems. In: Inproceedings of the 5th International Conference on Internet of Things (IoT 2015); 2015; Seoul, South Korea:122-129.

35. Wong V-S, Leung VC. Location management for next-generation personal communications networks. *IEEE Netw*. 2000;14(5):18-24.

36. Ahn S, Ramaswam V. Fluid flow models and queues—a connection by stochastic coupling. *Stoch Model*. 2003;19(3):325-348.

37. Jain G, Sigman K. A Pollaczek-Khintchine formula for M/G/1 queues with disasters. *J Appl Probab*. 1996;33(4):1191-1200.

**How to cite this article:** Cho C, Chung T-M. A novel architecture of Proxy-LMA mobility management scheme for software-based smart factory networking. *Int J Commun Syst*. 2018;31:e3584. [https://doi.org/10.1002/dac.3584](https://doi.org/10.1002/dac.3584)