PAPER

Proposal and Evaluation of a Function-Distributed Mobility Architecture for the Future Internet

Gen MOTOYOSHI[†,††a)], **Kenji LEIBNITZ**[††b)], ***Members, and*** **Masayuki MURATA**[††c)], ***Fellow***

SUMMARY Several task forces have been working on how to design the future Internet in a clean slate manner and mobility management is one of the key issues to be considered. However, mobility management in the future Internet is still being designed in an "all-in-one" way where all management functions are tightly kept at a single location and this results in cost inefficiency that can be an obstruction to constructing flexible systems. In this paper, we propose a new function-distributed mobility management architecture that can enable more flexible future Internet construction. Furthermore, we show the effectiveness of our proposed system via a cost analysis and computer simulation with a random walk mobility model.

key words: *Distributed mobility; future Internet; random walk model; cost analysis*

1. Introduction

Recently, societal requirements have become so complicated that the current Internet is struggling to meet them completely and it is now facing many challenging issues. In order to accomplish not only an incremental, but drastic innovation, many research works for new generation networks, in other words the future Internet, are being attempted from scratch all over the world. High capacity, huge numbers of devices, high reliability, as well as ecological and sustainable society support are among these challenging issues. We can see high activities in future Internet research efforts [1] in order to support the requirements above and make our future life more prosperous and it is easy to imagine that supporting user mobility is one of those critical functions.

In the current IP network era, *Mobile IP* (MIP) is the de facto standard mobility-supporting protocol [2, 3]. After Mobile IP was launched, several similar mobility protocols have been proposed, which are more efficient for handling specific objectives. Among them, *Fast MIP* (FMIP) [4] is suitable for carrying out fast handovers by using context transfer technology. On the other hand, *Hierarchical MIP* (HMIP) [5] has benefits in signaling cost reduction by using a localized binding update procedure with hierarchical network configuration. Furthermore, *Proxy MIP* (PMIP) [6] is well suited for *Mobile Node* (MN) workload reduction by shifting mobility functions up to the edge node.

In addition, a hierarchical network configuration and centralized management system are commonly used in the mobility management of cellular networks. In next generation mobile networks, such as System Architecture Evolution/Evolved Packet Core (SAE/EPC) and Long Term Evolution (LTE), both GPRS-based (General Packet Radio System) and IP-based mobility management systems are considered and standardized [7, 8] for an all-IP mobile network and PMIP is adopted as IP-based mobility protocol. Gradually, IP mobility and cellular mobility are being harmonized and unified for future mobile networks.

Nowadays, several research task forces, such as AKARI [9] in Japan, have been working on future Internet research. Due to their efforts, we can see some classification of primary functions for the future Internet, such as security, content delivery mechanisms, delay tolerant networking (DTN), management and control framework, service architecture, routing, and future Internet infrastructure design for experimentation. As for mobility, it is one of the important features within the management/control framework and routing, and MILSA [10] seems to be one of the promising architectures dealing with mobility issues, such as multi-homing and ID/locator separation.

In this paper, we propose a new mobility management framework to provide a more efficient scheme leading to a more flexible system construction. We believe that this framework suits well to the requirements of the future Internet. We study the efficiency of our proposed method via a cost analysis based on location update cost and packet delivery cost and verify the validity of the cost analysis by computer simulation.

The rest of this paper is organized as follows. In Section 2 we discuss related works. Then, in Section 3 we describe the design principles and in Section 4 the proposed function-distributed mobility system is explained in detail. In Section 5, evaluation results are shown comparing conventional mobility systems to our

Manuscript received June 16, 2010.
Manuscript revised January 1, 201x.
Final manuscript received January 1, 201x.
[†]The author is with NEC Corporation, System Platforms Research Laboratories, 1753 Shimonumabe, Nakahara-ku, Kawasaki, Kanagawa, 211–8666, Japan.
[††]The authors are with the Graduate School of Information Science and Technology, Osaka University, 1–5 Yamadaoka, Suita, Osaka 565–0871, Japan.
a) E-mail: g-motoyoshi@aj.jp.nec.com
b) E-mail: leibnitz@ist.osaka-u.ac.jp
c) E-mail: murata@ist.osaka-u.ac.jp
DOI: 10.1587/transcom.E0.B.1

proposed one by performing a cost analysis with a random walk mobility model. Moreover, simulation results are shown and the validity of the analyses is confirmed. Finally, Section 6 concludes this paper.

2. Background and Related Work

In the case of a mobility management system using geographically static anchors, supporting user mobility has some drawbacks in terms of the signaling cost and data transfer cost. Therefore, dynamically controlled mobility management systems are examined under various aspects. Wong et al. [11] show that a dynamic scheme outperforms a static scheme by assigning each user an own local area according to his profile. In addition, Ho et al. [12] and Li et al. [13] study a dynamic location update scheme from the viewpoint of location update cost and paging cost analysis. Chen et al. [14] propose a system to use dynamic location area management with minimum total cost. Choi et al. [15] distinguish MNs into several categories, such as predictable and unpredictable, and change the user's profile adaptively according to his mobility pattern.

In general, a centralized and static mobility management system is in the cases above regarded as the basic concept. However, such a management node could be a single point of failure and an obstruction to constructing a flexible system. Therefore, distributed systems have been studied under various aspects. Zheng et al. [16] propose a temporary home agent in the visited network and perform dynamic Home Agent (HA) assignment in order to avoid a single point of failure and gain signaling cost efficiency at the same time. Yen et al. [17] use the anycast protocol to find the nearest HA and register there, which leads to cost reduction. Song et al. [18] propose another dynamic mobility management system with a flexible foreign agent grouping scheme and Bertin et al. [19] define a dynamic mobility to work in a flat network architecture in a distributed manner. Pack et al. [20, 21] and Singh [22] perform a cost analysis of HMIP compared with MIP and prove the effectiveness of HMIP.

Several effective mobility methods have been examined up to date, however, all of them only consider a single mobility core function converged at a single node even though its geographical position can be dynamically changed. In other words, the mobility management function consists of several features such as *location management*, *authentication management*, *state management*, and so forth. Each of these sub-functions has its own characteristics and its own appropriate place to be located according to its characteristics. Therefore, a single mobility core function cannot accomplish to manage all these objectives at the same time and this leads to inefficiency in system management.

In this paper, we propose a function-distributed mobility management system [23], which is more efficient than ordinary methods. It can meet the needs of each sub-function at the same time and construct a more flexible mobility management system.

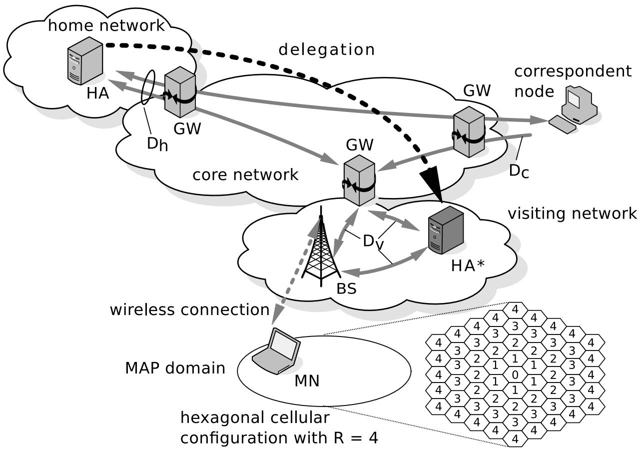


Fig. 1 Mobility model network configuration

3. Design Principles

In this section, the analytical mobility model and system parameters are discussed. Figure 1 illustrates the mobility model network configuration and an asterisk '*' indicates a delegated node in this figure. A *Mobile Node* (MN) is connected over a wireless link to a *Base Station* (BS) in a visiting network. All traffic between the MN and the remote destination is being relayed via *gateway* (GW) nodes to the *Home Agent* (HA) located in the home network of the MN. The values D_h, D_v, and D_c are the number of hops between each network and HA^* represents the delegated node with a distributed mobility function. Figure 1 indicates the situation where the master HA in the home network executes the delegation of a function to a node labeled as HA^* in the visiting network.

As in many other papers, we assume a hexagonal cellular mobility network model [21, 22], where a *Mobility Anchor Point* (MAP) domain consists of the same number of rings and each ring r has $6r$ cells. Then, the number $N(R)$ of cells up to ring R is calculated as:

$$N(R) = \sum_{r=1}^{R} 6r + 1 = 3R\,(R+1) + 1.$$

In this paper, we perform cost analysis by applying a random walk mobility model, which is a typical mobility model. The following subsection will show how the steady-state probabilities for a MN being in a certain cell r are obtained.

3.1 System Modeling

Figure 2 illustrates the one-dimensional Markov chain

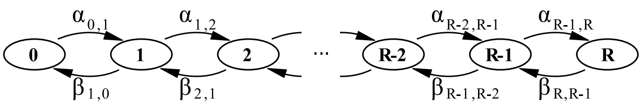


Fig. 2 State diagram for random walk mobility model

model that we use in this analysis. The number of each state corresponds to the ring number in the hexagonal cellular model. We define q as the probability of the MN staying in the current cell and thus it moves to another cell with probability $1-q$. Therefore, an MN located in a cell of ring r can move in an outward direction with the probability $p^{+}(r)$ and inward with $p^{-}(r)$.

$$p^{+}(r) = \frac{1}{3} + \frac{1}{6r} \qquad \text{and} \qquad p^{-}(r) = \frac{1}{3} - \frac{1}{6r} \tag{1}$$

Using the probabilities in Eqn. (1), we obtain the transition probabilities $\alpha_{r,r+1}$ and $\beta_{r,r-1}$ for an MN in an arbitrary ring r as in Eqns. (2) and (3).

$$\alpha_{r,r+1} = \begin{cases} 1-q & \text{if } r=0 \\ (1-q)\,p^{+}(r) & \text{if } 1 \le r < R \end{cases} \tag{2}$$

$$\beta_{r,r-1} = (1-q)\,p^{-}(r) \qquad \text{if } 1 \le r \le R \tag{3}$$

Here, if we define $\pi_{r,R}$ as steady-state probability of state r within the MAP domain being composed of cells inside ring R, it is calculated as

$$\pi_{r,R} = \pi_{0,R} \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}} \qquad \text{for} \quad 1 \le r \le R$$

$$\pi_{0,R} = \left(1 + \sum_{r=1}^{R} \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}}\right)^{-1}$$

with $\sum_{r=0}^{R} \pi_{r,R} = 1$.

3.2 Cost Functions

In this section, the considered cost functions are defined. In order to conduct a fair evaluation of the true properties that each mobility method has, the total cost for each method is analyzed without the paging cost. This is done because paging is not always supported by all methods, and existing studies like [13] indicate that paging cost is at most of the order of location update cost. Since all mobility methods use a similar paging-like function, the paging costs are nearly same irrespective of the mobility method and we therefore expect that it does not have a large impact on the qualitative cost comparison. Therefore, the total cost

$$C_{tot} = C_{loc} + C_{pkt}$$

consists of the sum of location update cost C_{loc} and packet delivery cost C_{pkt}.

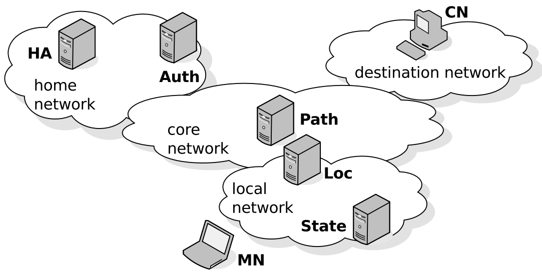


Fig. 3 DisMob network configuration example

3.2.1 Location Update Cost Model

In HMIP, DynMob, and DisMob, there are location update procedures executed localized, whereas a global location update procedure is only done all the time in MIP. Here, *DynMob* is the abbreviation of *dynamic mobility* and indicates a single mobility core function that can be dynamically assigned via a temporary home agent [16], as shown in Fig. 1. Also, *DisMob* stands for *distributed mobility* to indicate the system where the mobility function is divided into several sub-functions and each sub-function is dynamically assigned to different nodes according to their characteristics, see Fig. 3.

Take the location update cost in DisMob for example. Let C_H, C_{LD}, and C_{LN} be the global location update cost, localized location update cost with delegation, and localized location update cost without delegation, respectively. Here, it is assumed that the most appropriate nodes for location and path management change as the MN moves within the MAP domain and the delegation procedure for location and state management takes place even within a single MAP domain. Performing global location update means that the MN in the boundary ring R moves in an outward direction. In other situations, the MN performs a localized location update and in order to calculate the delegation probability, parameter ω is introduced as the probability with which the MN performs the delegation process. Here, the probability to perform global location update is calculated as $\pi_{R,R}\,\alpha_{R,R+1}$ and, therefore, the location update cost is calculated as follows:

$$C_{loc} = \frac{1}{\overline{T}} \big(\pi_R\, \alpha_{R,R+1} C_H + (1 - \pi_R \alpha_{R,R+1}) \left[\omega\, C_{LD} + (1-\omega)\, C_{LN}\right] \big)$$

with

$$\begin{aligned} C_H &= 2\left[\kappa + \tau\left(D_v + D_h\right)\right] \\ &\quad + N_{CN}\left(2\left[\kappa + \tau\left(D_v + D_c\right)\right] + PC_{CN}\right) \\ &\quad + PC_{HA} + PC_{HA_d} + PC_{HA_d^*} \end{aligned} \tag{4}$$

$$C_{LN} = 2\left(\kappa + \tau D_v\right) + PC_{HA^*} \tag{5}$$

$$\begin{aligned} C_{LD} &= 2\left(Z_s\, Y_s + Z_l Y_l\right)\left(\kappa + \tau D_v\right) \\ &\quad + 2\, Z_p\, Y_p\, N_{CN}\left[\kappa + \tau\left(D_v + D_c\right)\right] \\ &\quad + PC_{HA^*} + PC_{HA_d^*} + PC_{HA_d^{**}} \end{aligned} \tag{6}$$

where $\bar{T}$ is the average cell residence time and κ and τ are the unit transmission costs for wired and wireless links, respectively. Here, $\bar{T}$ is calculated as the average time period for a mobile user to stay in the same cell, D_h, D_c, and D_v are the number of hops in each traversed network, and N_{CN} is the number of CNs (*Correspondent Nodes*) that the MN communicates with; PC_{xx} represents the processing costs for binding update procedures at each node, where '*' means that the costs are caused by the delegation process; '*' with and without 'd' means the costs to be delegated by another node and to delegate a sub-function to another node, respectively. In this paper, PC_{HA^*} is assumed equivalent for PC_{HA}, PC_{HA_d}, and $PC_{HA_d^*}$, while $PC_{HA_d^{**}}$ is equivalent to $PC_{HA_{MAP}}$. Finally, $Y_{s/l/p}$ and $Z_{s/l/p}$ are delegation probability parameters for state, location, and path management, respectively. $Y_{s/l/p}$ indicates the cost reduction improvement ratio by delegation and delegation starts only when cost reduction effects are expected to be obtained by delegation. $Z_{s/l/p}$ indicates the ratios of the cost by each function process to the overall cost. Therefore, the sum over $Z_{s/l/p}$ is 1. Although our system has four types of distributed functions, *authentication* is only performed initially and then at a low frequency, which is the reason why we will not include the authentication process in our cost analysis.

Let us discuss C_H as expressed in Eqn. (4) as example. When the MN sends a location update message toward the HA in the home network, it will be first transferred over a wireless link to a BS in the visiting network, then over a wired links to a GW between visiting and core network in the visiting network, then to the GW between home and core network in the core network, until it finally reaches the HA via a wired link in the home network. The location update response message from the HA will retrace these steps to the MN. The costs C_{LD} and C_{LN} in Eqns. (5) and (6) are derived in the same way.

We should also remark that in the case of DisMob, there will be interactions among each sub-functions. For example, location management and path management will choose the best delegation node for an MN using its state information. If the MN accesses each sub-function management node securely, it also needs to use the authentication management function. In addition, each sub-function itself may be distributed on a user-by-user basis requiring each function to coordinate with each other. However, in this paper, this kind of cost by interaction processes is assumed to be accommodated within the delegation cost. We plan to extend the cost analysis to interaction process cost in future work. The location update costs for MIP, HMIP, and DynMob are calculated in a similar way and the equations can be found in the appendix.

3.2.2 Packet Delivery Cost Model

In this section, packet delivery cost is calculated. First, $N_{MN} = N_{AR}\,K$ is the total number of users in the current MAP domain and it is the product of the number N_{AR} of *Access Routers* (ARs) in the current MAP domain and the average number of users K within the coverage of an AR. Here, an AR is assumed to be located in a cell. Since N_{AR} is equivalent to the number of cells $N(R)$ in the MAP domain, the number of MNs N_{MN} increases with R.

Then, the packet delivery cost is expressed as follows, where C_{MAP}, C_{HA}, and C_T are processing costs for packet delivery at MAP and HA, and the packet transmission cost from CN to MN, respectively.

$$C_{pkt} = C_{MAP} + C_{HA} + C_T \quad (7)$$

Firstly, C_{MAP} is in general divided into two parts, such as lookup and routing cost. Lookup cost is proportional to the mapping table size, i.e., the number of MNs. On the other hand, routing cost is known to be proportional to the logarithm of the number of ARs in the MAP domain [20]. Secondly, in terms of C_{HA}, route optimization only lets the first packet transit the node which has the function corresponding to the HA. All following packets of the session are transferred directly to the MN. Thirdly, C_T is the transmission cost and it depends the distance, i.e., the number of hops between MN and CN. Finally, according to the discussion above, the total packet delivery cost is calculated as follows:

$$\begin{aligned} C_{pkt} = \lambda_s \big(&\bar{S}\,[w_{MN}\,N_{MN} + w_{AR}\,\log(N_{AR})] + \theta_{HA} \\ &+ \tau\left[(\bar{S}-1)(D_c + D_h) + (D_h + D_v)\right] + \kappa\,\bar{S}\big) \end{aligned}$$

where λ_s, $\bar{S}$ and θ_{HA} are the session arrival rate, average session size in units of packets, and unit packet processing cost, respectively. The terms w_{MN} and w_{AR} are used as weighting factors.

4. System Description

In the current mobility management system, all mobility-related functions are "all-in-one" and centrally located in the same place. However, each function has very specific characteristics of its own that each function should be divided and located at its own appropriate place in a dynamical manner. For that reason, we propose a function-distributed mobility system and this distribution mechanism is performed by a delegation procedure. The details of this scheme are explained in this section.

4.1 Function-Distributed Mobility Architecture

In this section, we explain the function-distributed mobility system. First, Fig. 3 briefly sketches a network

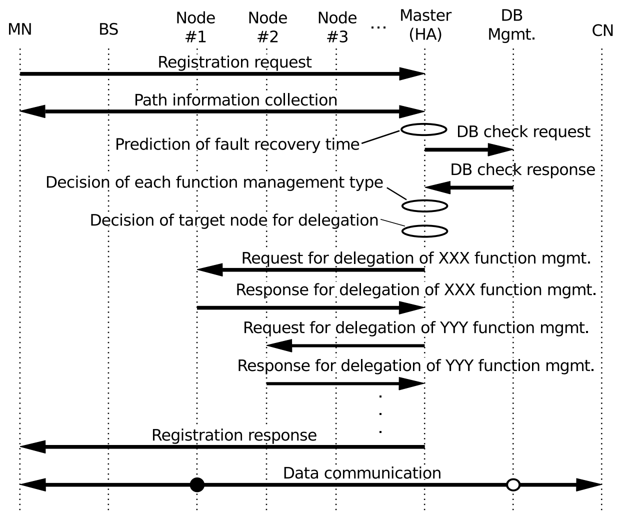


Fig. 4 Basic sequence of delegation procedure

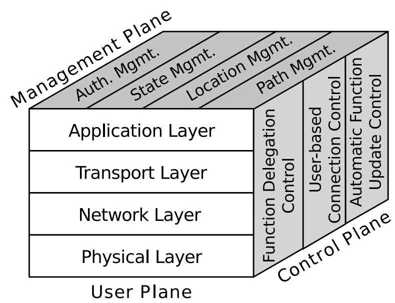


Fig. 5 Layered structure of function-distributed mobility

configuration example supporting function-distributed mobility (DisMob). In typical cases, the whole network consists of home network, core network, local network, destination network, and mobile node. Home network is where the mobile user is registered, destination network is where the targeted CN is connected, local network is where the MN is currently visiting, and core network connects all the above-mentioned networks. *Auth*, *Path*, *Loc*, and *State* in Fig. 3 are all distributed and delegated function elements.

Next, a basic delegation sequence procedure is shown in Fig. 4. The MN starts to send a registration request to a pre-defined *master node*, which is named as HA in Fig. 4, after power on and the network attachment procedure, and then the MN and master node maintain the connecting path information. At the master node, fault recovery time is calculated with database information of the environment using real-time information if possible. After that, the master node selects the most appropriate node for each function to be delegated according to the information above and starts to execute the delegation procedures. After all delegations have been successfully completed, the master sends an acknowledgement signal to the MN. Finally, the MN can communicate with the CN and receives optimal function-placed service leading to optimal cost.

In addition, the layer structure is shown in Fig. 5. On user plane, there is no difference from the conventional communication layer structure like the OSI model. As for the control plane, the function delegation procedure is of course required and flexible path management such as user-based management follows, which is a key factor of function-distributed mobility. In addition, each distributed function should be kept automatically and dynamically updated. Finally, the management plane accommodates each distributed function management entity.

4.2 Features of Function-Distributed Mobility

As for the function-distributed mobility, the appropriate placement of each function is of great importance. Table 1 represents the impact of each function management scheme from mobility system parameters. Most mobility system parameters are dynamically updated and the higher the impact is, the closer the distributed function should be located to the MN. Therefore, the *authentication function* should be located deep inside the network, e.g., in the home network, while the *state function* should be located at the nearest node in the visited local network. In addition, the *path management function* should be located somewhere in-between the source and destination, e.g., in the middle of the core network, to cope with fast switchover of paths. Finally, the *location management function* consisting of both anchoring and casting should be placed at intermediate locations, e.g., in the middle of the local network, where it is near to the core network and also the MN. Here, *anchoring* is the function that a specific node provides a constant presence of static IP address for the MN and works like an anchor to support loss-less communication to user movement. The *casting function* multicasts user data toward potential nodes that the MN moves in advance to support user movement.

This distribution does not lead to a mere trade-off between efficiency and management cost. As stated earlier in Section 4.2, each location of sub-functions in Fig. 3 illustrates one type of ideal conditions. This distributed disposition of sub-functions makes it possible for us to carry out well-shaped and well-optimized location update processes. For instance, mobile nodes and a mobile infrastructure only have to follow well-optimized location update processes, meaning that messages with minimum necessary payloads can be exchanged along minimum necessary routes on each occasion.

4.3 Comparison between Function-Distributed and Conventional Mobility Systems

In this section, function-distributed mobility is compared with MIP, HMIP, DynMob, and LTE in terms of

Table 1 Impact of each function placement from mobility system parameters (H:high, M:medium, L:low)

Parameter	Each mobility function				
	Anchor	*Cast*	*Path*	*State*	*Auth*
Access frequency	H	M	M	H	L
MN speed	M	H	H	H	L
Recovery time	M	H	H	H	L
Traffic volume	M	H	H	L	L
MN number	L	M	M	H	M
Application type	M	L	H	H	L

Table 2 Qualitative comparison between mobility systems

	MIP	HMIP	DynMob	LTE	DisMob
Location update	global	global, local	global, local	global, local	global, local
Anchor point	fixed	fixed	dyn.	dyn.	dyn.
Function "all-in-1"	yes	yes	yes	yes	*no*
Path mgmt.	fixed	fixed	fixed	fixed	*flexible*

mobility specific metrics, such as location update procedure type, anchor point flexibility, function distribution, and path management capability.

Table 2 indicates a qualitative comparison with other conventional mobility methods. From the viewpoint of location update type, MIP executes all the time a global registration whereas the other methods have the ability of using a localized and cost-efficient registration process. In terms of anchor point flexibility, MIP and HMIP basically use a fixed point. However, the other schemes could change their anchor point dynamically according to user preferences and environment conditions. As for function distribution, which leads to a flexible system construction, distributed mobility is the only way to accomplish this feature. Also only DisMob has the ability to manage transit paths from source to destination. These two functions are key differentiating factors between DisMob and conventional systems.

5. Numerical Evaluation

In this section, we show numerical results based on the mobility system we discussed in Section 4. First of all, the system parameters are shown in Table 3, which are following previous works that we believe are realistic [21]. All parameters are the same that appeared in previous sections, except for the delegation parameter for DynMob γ. Four methods, MIP, HMIP, DynMob, and our proposed DisMob are evaluated by their costs in the following section.

5.1 Simulation Settings

While our focus lies on the analytical evaluations, we also show results from computer simulation to validate our numerical analyses. Simulation conditions are shown in Table 4. In general, we assume the same conditions for the simulations and numerical analysis as much as possible. Mobile nodes are simulated to move in a random walk manner within hexagonal cellular fields and establish connections with the correspondent node according to the session arrival rate. In order to compare numerical analysis and simulation, an ideal wireless condition is assumed in simulations and our computer simulation program is implemented in the C programming language. Each plot shows the average of 100 simulation runs, corresponding to 100 different random seeds. We show numerical evaluation results for parameters given in Table 3 and for MAP domain sizes $R = 1$ and $R = 4$. We show the results for $R = 1$ since they can be regarded as baseline and $R = 4$ because they revealed the most interesting phenomena including crossover points of cost. Other values of R yielded similar results following a nearly linear tendency.

Table 3 Basic system parameters used in the evaluation

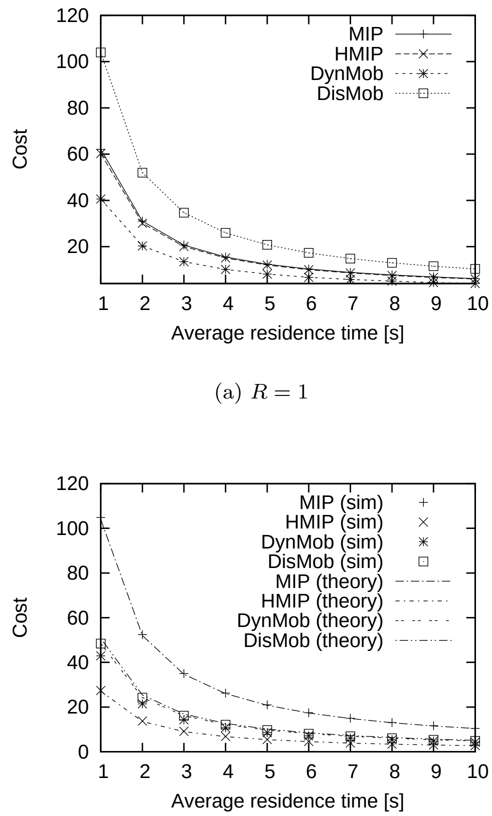
PC_{CN}	PC_{HA}	PC_{HA^*}	PC_{HAd}	PC_{HAd^*}	$PC_{HAd^{**}}$
6	24	24	12	12	12
θ_{HA}	θ_{HA^*}	N_{AR}	N_{CN}	ω	γ
20	20	1	2	0.55	0.43
Z_s	Z_l	Z_p	Y_s	Y_l	Y_p
0.1	0.6	0.3	0.7	0.7	0.7
τ	κ	K	D_h	D_c	D_v
1	2	4	6	4	2
w_{MN}	w_{AR}	$\bar{S}$	q	λ_s	
0.1	0.2	10	0.4	0.1	

5.2 Location Update Cost Analysis

One of the most characteristic features in mobility systems is the location update procedure. Therefore, location update cost is evaluated first. Figures 6(a) and 6(b) show the location update cost against average cell residence time for MAP domain sizes $R = 1$ and $R = 4$, respectively. Location update costs are closely related to user mobility and cell residence time is regarded as a good indicator for that. Therefore, location update cost decreases as cell residence time increases. In addition, the smaller the MAP domain size R is, the more often an MN also performs a global location update rather than a localized location update. For comparison, location update cost normalized by the number of

Table 4 Simulation conditions

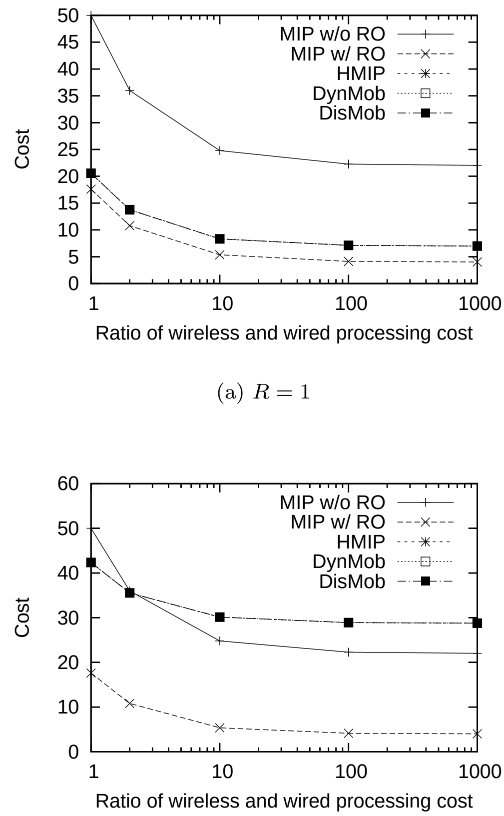
parameter	value
simulation field	hexagonal cellular
mobile node movement	random walk
mobility speed	10 [m/s]
session arrival rate	0.1
traffic type	constant bit rate
wireless link	ideal (no packet loss)
simulation time	1,000,000 [s]



(a) $R = 1$

(b) $R = 4$

Fig. 6 Location update cost over residence time



(a) $R = 1$

(b) $R = 4$

Fig. 7 Packet delivery cost over wireless and wired processing cost ratio

handovers normalized by simulation time are used as simulation results. It can be seen that the simulation results fit well with theoretical values.

HMIP, DynMob, and DisMob show better performance than MIP. This is because these three methods can perform a localized location update process to omit location update cost as long as the MN moves within a small localized area. On the other hand, MIP has to perform global location updates all the time. This property also explains the fact that the cost in HMIP, DynMob, and DisMob decreases as the MAP domain size R increases, whereas the costs in MIP is independent of R. Besides, we could see some small difference among the results by these three methods and they are simply caused by the delegation cost in the case of DynMob and DisMob. Unlike HMIP, DynMob requires a delegation process and DisMob needs a more precise delegation process than DynMob. Therefore, the delegation cost of DynMob is smaller than that of DisMob. As a result of this section, the methods other than MIP seem to be the better solutions just from the viewpoint of location update cost.

5.3 Packet Delivery Cost Analysis

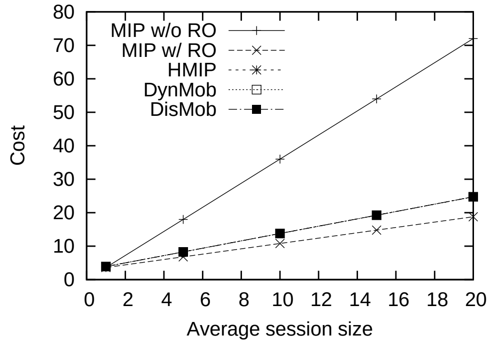
Next, the packet delivery cost is evaluated in this section, which is another typical metric for mobility system evaluation. The packet delivery process is closely related to user population, whereas location update cost is influenced by user mobility. In other words, packet delivery cost increases with the number of MNs in the MAP domain. From most previous works [20, 21], it is well-known that the lookup procedure to confirm whether the targeted MN is registered in the mapping table or not is the most costly process. In our previous work, we analyzed packet delivery cost against number of MNs in a cell [23].

Furthermore, the wireless communication environment is rapidly growing these days. The growth of wireless processing cost is observed and the amount of data transmission over the wireless link is growing steadily. From this point of view, wireless and wired processing costs and average session size are important factors for evaluating mobility systems.

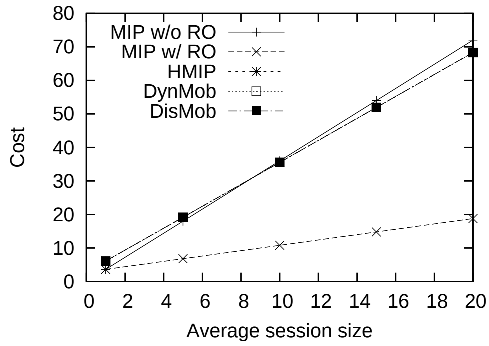
First, Figs. 7(a) and 7(b) show the packet delivery cost against ratio of wireless and wired processing cost for MAP domain sizes $R = 1$ and $R = 4$, respectively. The wireless and wired processing costs are τ and κ, respectively, so by keeping a fixed wired processing cost of $\kappa = 2$, we can observe different ratios κ/τ. Just for reference, MIP results with and without route optimization (RO) are also given in this evaluation. The other three methods except for MIP show the same performance results and this is because packet delivery cost is heavily dependent on the route of packet delivery and session arrival rate. In MIP, packet delivery costs are same for $R = 1$ and $R = 4$, whereas in the other three methods it is higher for $R = 1$ than for $R = 4$. For smaller MAP domain size $R = 1$, costs in the other three methods are closer to that of MIP with RO. On the other hand, with larger MAP domain size $R = 4$, the costs in the other three methods approach those of MIP without RO and a crossover point at a ratio of about 2, which is well within the practical processing cost range, is seen. The other three methods show better results at lower ratio, while MIP without RO is better at higher ratio. The crossover point appears because the lookup procedure cost in the three methods except for MIP is linear to the number of MNs in a cell and MIP has constant behavior to perform the same transmission mechanism using the MIP tunnel.

Figures 8(a) and 8(b) show the packet delivery cost against average session size, with respect to transmission data volume such as a unit of bytes for $R = 1$ and $R = 4$, respectively. The other three methods except for MIP show identical values due to the same reasons stated earlier. It is obvious that the cost increases linearly with the average session size. Again, the costs for the other three methods are closer to that of MIP with RO when $R = 1$, whereas it is closer to that of MIP without RO when $R = 4$. A crossover point appears at *average session size* $= 10$.

From these results, MIP especially with RO seems to be the best solution, just from the viewpoint of packet delivery cost. The management of system configuration and cost reduction are effective in making good use of these crossover point properties.



(a) $R = 1$



(b) $R = 4$

Fig. 8 Packet delivery cost over session size

5.4 Total Cost Analysis vs. SMR

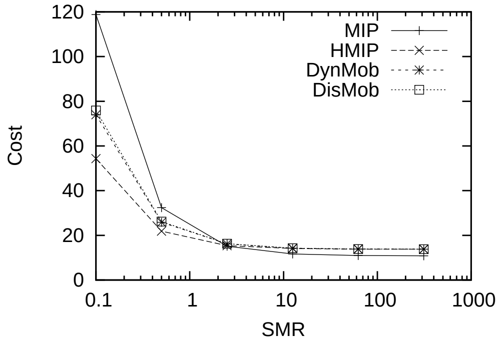
In the previous two sections, we could find benefits for each of MIP and the other three methods via location update and packet delivery cost analysis. In this section, the total cost is evaluated as the sum of both.

For evaluation of mobility systems, *Session-to-Mobility Ratio* (SMR) is commonly used as metric, which is a mobile packet network's counterpart of Call-to-Mobility Ratio in Personal Communication Service (PCS) networks. SMR is the relative ratio of the session arrival rate to user mobility. Here in the random walk model, SMR is defined as $\lambda_s \bar{T}$.

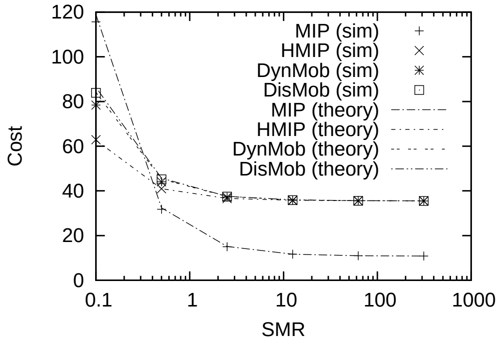
Figures 9(a) and 9(b) present the total cost analysis against SMR for $R = 1$ and $R = 4$, respectively. The three methods show almost the same performance as in location update and packet delivery cost analysis. MIP shows a cost beneficial behavior for higher SMR and the other methods for lower SMR. Around SMR $= 1$ is the borderline where the tendency is turned over and this phenomenon is clearer at large MAP domain size.

In addition, Figs. 10 and 11 present the total cost against ratio of wireless processing and average session size, respectively. The three methods show almost the same performance as for the ratio of wireless processing and average session size. From Fig. 10, MIP with RO shows cost beneficial behavior in most ratios, however, the other three methods have potential to improve to higher levels in the future when wireless exceeds wired communication conditions, which means lower ratio than 1. Also, from Fig. 11, MIP with RO shows cost beneficial behavior in most average session sizes, however, in the case of $R = 4$, an average session size between 5 and 8 appears as turning point where the other three methods become the most cost beneficial.

As for total cost versus MAP domain size R, the increase of R has no impact on MIP in terms of total cost. However, especially at low SMR such as 0.1,

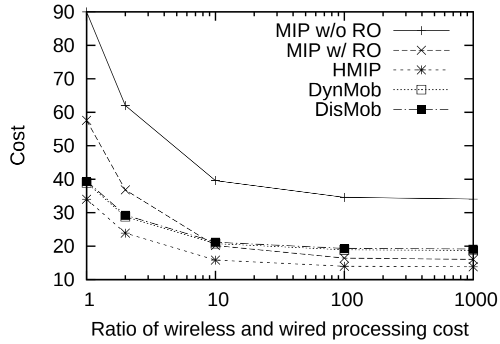


(a) $R = 1$

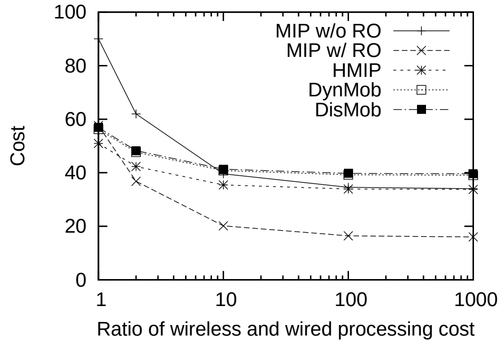


(b) $R = 4$

Fig. 9 Total cost over SMR



(a) $R = 1$



(b) $R = 4$

Fig. 10 Total cost over wireless/wired processing cost ratio

the minimal total cost in the other three methods is reached for MAP domain size $R = 2 \sim 3$ and in case of high SMR, it increases almost linearly with R. This tendency towards an optimal MAP domain size is only visible for the total cost analysis and not for location update and packet delivery costs.

In summary, MIP with RO tends to be beneficial in most cases, however, the other three methods can become good candidates in some specific conditions.

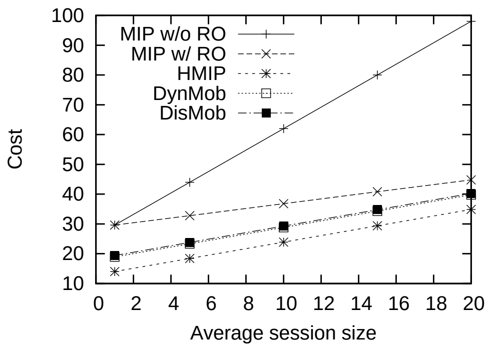
5.5 Fair Evaluation by Total Cost Analysis

Until here, the benefits of the three methods except for MIP have been confirmed by cost analysis at equal MAP domain sizes. Delegation imposes additional costs compared to mobility management without delegation. Therefore, on comparison between different mobility methods at equal MAP domain sizes, the methods with delegation may appear worse from the cost view. However, if we compare costs under the condition of the same localized area size, where no delegation is necessary, the cost increase by delegation management and cost reduction by delegation efficiency effects can be investigated and evaluated. For this reason we consider this type of evaluation fairer than those in the previous sections.

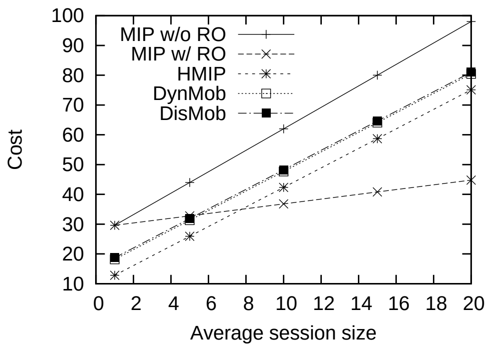
Figure 12 presents the unified localized area concept. In case of DisMob, according to the probability ω, the delegation procedure is assumed to be executed, i.e., the area of no-delegation corresponds to the area expressed by $1 - \omega$. Similarly, in case of DynMob, according to the probability γ, the delegation procedure is assumed to be executed, i.e., the area of no-delegation corresponds to the area expressed by $1 - \gamma$. In order to equalize the localized area among all considered systems, i.e., the area without delegation, we adjust delegation parameters γ and ω. First, we select $R = 3$ for HMIP, $R = 2$ for DynMob, and $R = 1$ for DisMob in order to have a margin for producing a potential overlap. Then, we calculate the localized area size for each method using γ and ω. Finally, each localized area size is unified by adjusting γ and ω as follows.

$$\frac{1-\gamma}{\gamma} = \frac{9\pi R^2}{16\pi R^2 - 9\pi R^2} \qquad \Rightarrow \qquad \gamma = \frac{7}{16}$$

$$\frac{1-\omega}{\omega} = \frac{4\pi R^2}{9\pi R^2 - 4\pi R^2} \qquad \Rightarrow \qquad \omega = \frac{5}{9}$$



(a) $R = 1$



(b) $R = 4$

Fig. 11 Total cost over session size

Figures 13(a) and 13(b) represent the cost for location update and packet delivery in a unified local area, respectively. From Fig. 13(a), HMIP still shows the best performance among the three dynamic methods. However, from Fig. 13(b), DisMob shows the best performance among the three when we consider packet delivery cost. This is because delegation cost itself seems to have a closer relationship with location update cost, but packet delivery cost has a closer relationship with MAP domain size.

Figure 13(c) shows the total cost analysis in the unified local area. In SMR above 10, MIP with RO shows the best performance followed by DisMob. In the medium range of SMR around 1, DisMob shows the best performance, and for SMR less than 1, HMIP shows the best performance followed by DynMob and DisMob. In summary of these results, MIP has a slight advantage at higher SMR, but disadvantages at lower SMR, while HMIP has a slight advantage at lower SMR, but disadvantages at higher SMR. On the other hand, DisMob outperforms DynMob in most SMR ranges. As a consequence, DisMob has high flexibility to cope with overall conditions in all SMR ranges.

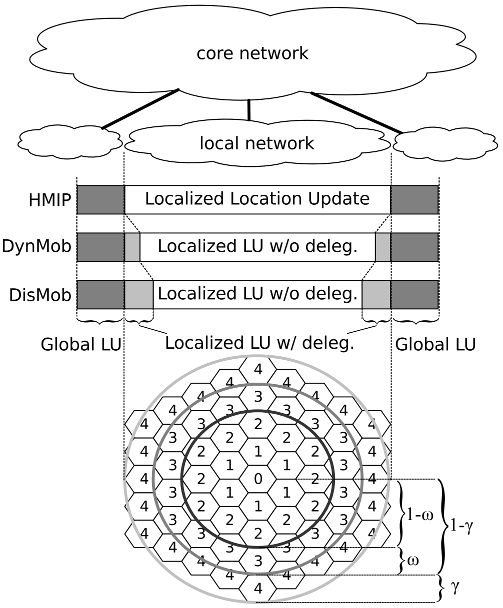


Fig. 12 Concept of localized area

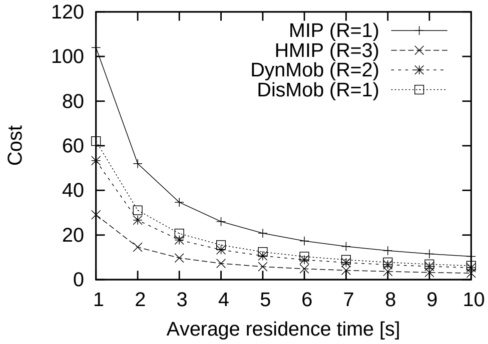
6. Conclusion

In this paper, we proposed a function-distributed mobility system for the future Internet. First, we showed qualitative benefits of our proposed method. Then, compared with conventional MIP, HMIP, and DynMob, location update cost and packet delivery cost were analyzed for a random walk mobility model from the viewpoint of the mobility method and SMR. The cost analysis showed that HMIP, DynMob, and DisMob show a similar performance and have a better performance than MIP. At higher SMR, MIP has more benefits, however at lower SMR, the other three methods are better. In addition, we showed that our proposed function-distributed mobility method showed better performance by cost analysis on a unified localized area in order to cope with overall SMR environment at the same time. Finally, we shared simulation results, confirmed the validity of our numerical analysis, and showed that our proposed function-distributed mobility method is effective and flexible.

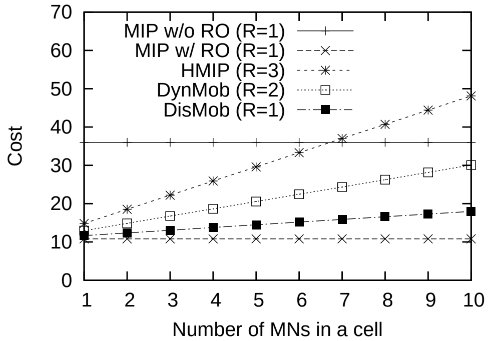
In our future work, we are planning to include the effects of paging cost and interaction process cost, different mobility models, and more realistic parameter fluctuations, as well as further analytical studies and efficient function placement.

Acknowledgment

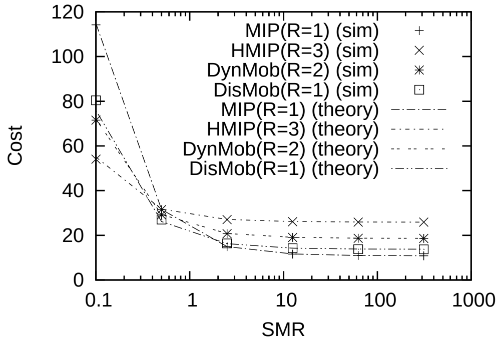
The authors would like to thank Toshiyuki Kanoh, General Manager of the System Platforms Research Laboratories at NEC. This work would not have been possible without his continuous and generous support.



(a) Location update cost



(b) Packet delivery cost



(c) Total cost

Fig. 13 Cost analysis in unified local area

References

[1] S. Paul, J. Pan, and R. Jain, "Architecture for the future networks and the next generation internet: A survey," Tech. Rep. 2009-69, WUSTL, `http://www.cse.wustl.edu/~jain/papers/i3survey.htm`, October 2009.
[2] C. Perkins, "IP mobility support for IPv4." IETF RFC3344, August 2002.
[3] D. Johnson, C. Perkins, and J. Arkko, "Mobility support in IPv6." IETF RFC3775, June 2004.
[4] R. Koodli, "Mobile IPv6 fast handovers." IETF RFC5568, July 2009.
[5] H. Soliman, C. Castelluccia, K. ElMalki, and L. Bellier, "Hierarchical mobile IPv6 (HMIPv6) mobility management." IETF RFC5380, October 2008.
[6] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, "Proxy mobile IPv6." IETF RFC5213, August 2008.
[7] 3GPP TS23.401, "General packet radio service (GPRS) enhancements for evolved universal terrestrial radio access network (E-UTRAN) access." V9.3.0, December 2009.
[8] 3GPP TS23.402, "Architecture enhancements for non-3GPP accesses." V9.3.0, December 2009.
[9] AKARI Project, "New generation network architecture AKARI conceptual design (ver2.0 in Japanese and ver1.1 in English)." `http://akari-project.nict.go.jp/eng/overview.htm`.
[10] J. Pan, S. Paul, R. Jain, and M. Bowman, "MILSA: a mobility and multihoming supporting identifier locator split architecture for naming in the next generation internet," Proc. of IEEE Global Communications Conference (GLOBECOM'08), November 2008.
[11] V.W.S. Wong and V.C.M. Leung, "Location management for next-generation personal communications networks," IEEE Network, pp.18–24, September 2000.
[12] J.S.M. Ho and I.F. Akyildiz, "A dynamic mobility tracking policy for wireless personal communications networks," Proc. of IEEE Global Communications Conference (GLOBECOM'95), November 1995.
[13] J. Li, H. Kameda, and K. Li, "Optimal dynamic mobility management for PCS networks," IEEE/ACM Transactions on Networking, vol.8, no.3, June 2000.
[14] K.T. Chen, S.L. Su, and R.F. Chang, "Design and analysis of dynamic mobility tracking in wireless personal communication networks," IEEE Transactions on Vehicular Technology, vol.51, no.3, May 2002.
[15] W.J. Choi and S. Tekinay, "An adaptive location registration scheme with dynamic mobility classification," Proc. of IEEE International Conference on Communications (ICC'02), April 2002.
[16] R. Zheng, Y. Ge, J.C. Hou, and S.R. Thuel, "A case for mobility support with temporary home agents," ACM SIGMOBILE Mobile Computing and Communications Review, vol.6, no.1, pp.32–46, January 2002.
[17] Y.S. Yen, C.C. Hsu, Y.K. Chan, and H.C. Chao, "Global dynamic home agent discovery on mobile IPv6," 2nd International Conference on Mobile Technology, Applications and Systems, November 2005.
[18] M. Song, J. Huang, R. Feng, and J. Song, "A distributed dynamic mobility management strategy for mobile IP networks," IEEE 6th International Conference on ITS Telecommunications Proceedings, pp.1045–1048, June 2006.
[19] P. Bertin, S. Bonjour, and J.M. Bonnin, "A distributed dynamic mobility management scheme designed for flat IP architectures," New Technologies, Mobility and Security (NTMS '08), 2008.
[20] S. Pack and Y. Choi, "A study on performance hierarchical mobile IPv6 in IP-based cellular networks," IEICE Transactions on Communications, vol.E87-B, no.3, pp.462–469, March 2004.
[21] S. Pack, M. Nam, T. Kwon, and Y. Choi, "A performance comparison of mobility anchor point selection schemes in hierarchical mobile IPv6 networks," Computer Networks, vol.51, no.6, pp.1630—1642, April 2007.
[22] B. Singh, "Signaling cost analysis in mobile IP," Proc. of IET International Conference on Wireless, Mobile and Mul-

timedia Networks, pp.188–191, January 2008.
[23] G. Motoyoshi, K. Leibnitz, and M. Murata, "Function-distributed mobility system for the future internet," 5th IFIP/IEEE International Workshop on Broadband Convergence Networks (BcN 2010), Osaka, Japan, April 2010.

Appendix A: Location Update Cost

In this appendix, location update cost formulae for MIP, HMIP, and DynMob are shown. Each parameter has the same meaning as in Section 3.2.1.

A.1 MIP Location Update Cost

Firstly, in the case of MIP, let C_g be the location update cost. This location update procedure is executed all the time globally, irrespective of the mobile node's geographic position. Then, the location update cost is calculated as follows:

$$C_{loc} = \frac{1}{\overline{T}} C_g$$

with

$$\begin{aligned} C_g = & \, 2\left[\kappa + \tau\left(D_v + D_h\right)\right] \\ & + N_{CN}\left(2\left[\kappa + \tau\left(D_v + D_c\right)\right] + PC_{CN}\right) \\ & + PC_{HA} \end{aligned}$$

A.2 HMIP Location Update Cost

Secondly, in the case of HMIP, let C_g and C_l be the global location update cost and localized location update cost, respectively. Global location update is performed when the MN in the boundary ring R moves in an outward direction. In other situations, the MN performs a localized location update. The location update cost is calculated as follows:

$$C_{loc} = \frac{1}{\overline{T}} \left(\pi_R \, \alpha_{R,R+1} C_g + (1 - \pi_R \alpha_{R,R+1} C_l\right)$$

with

$$\begin{aligned} C_g = & \, 2\left[\kappa + \tau\left(D_v + D_h\right)\right] \\ & + N_{CN}\left(2\left[\kappa + \tau\left(D_v + D_c\right)\right] + PC_{CN}\right) \\ & + PC_{HA} + PC_{MAP} \\ C_l = & \, 2\left(\kappa + \tau D_v\right) + PC_{MAP} \end{aligned}$$

A.3 DynMob Location Update Cost

Finally, in the case of DynMob, let C_H, C_{LD}, and C_{LN} be the global location update cost, localized location update cost with delegation, and localized location update cost without delegation, respectively. Performing global location update means that the MN in the boundary ring R moves in an outward direction. In other situations, the MN performs a localized location update. This delegation is performed in a function-bundled manner and in order to calculate the delegation probability, parameter γ is introduced as the probability with which the MN performs the delegation process. Due to the additional complexity of DisMob with larger localized area and function distribution, the delegation probability γ of DynMob must be larger than that of DisMob, i.e., $\gamma < \omega$ to be efficient. From the above conditions, the location update cost is calculated as follows:

$$\begin{aligned} C_{loc} = & \frac{1}{\overline{T}} \big(\pi_R \, \alpha_{R,R+1} C_H + \\ & \quad (1 - \pi_R \alpha_{R,R+1}) \left[\gamma \, C_{LD} + (1-\gamma) \, C_{LN}\right]\big) \end{aligned}$$

with

$$\begin{aligned} C_H = & \, 2\left[\kappa + \tau\left(D_v + D_h\right)\right] \\ & + N_{CN}\left(2\left[\kappa + \tau\left(D_v + D_c\right)\right] + PC_{CN}\right) \\ & + PC_{HA} + PC_{HA_d} + PC_{HA_d^*} \\ C_{LN} = & \, 2\left(\kappa + \tau D_v\right) + PC_{HA^*} \\ C_{LD} = & \, 2\left(\kappa + \tau D_v\right) + PC_{HA^*} + PC_{HA_d^*} + PC_{HA_d^{**}} \end{aligned}$$

where PC_{xx} represents the processing costs for binding update procedures at each node, where '*' means that the costs are caused by the delegation process; '*' with and without 'd' means the costs to be delegated by another node and to delegate a bundle of functions to another node, respectively.

Gen Motoyoshi received his M.E. degree in electrical engineering from Saitama University, Japan, in 1995. He joined NEC Corporation in 1995 and has been engaged in the research on management of mobile cellular networks and personal area networks. He is now pursuing his Ph.D degree at the Graduate School of Information Science and Technology, Osaka University, Japan. His research interests are also in network management of complex systems. He is a member of IEEE, IEICE, and JSIAM.

Kenji Leibnitz received his master and Ph.D. degrees in information science from the University of Würzburg in Germany, where he was also a Research Fellow at the Institute of Computer Science. In May 2004, he joined Osaka University, Japan, as a Postdoctoral Research Fellow and from July 2006 he has been a Specially Appointed Associate Professor at the Graduate School of Information Science and Technology within the Yuragi Super-COE and Global COE projects. Since April 2010 he is with the National Institute of Information and Communications Technology as a senior researcher working in the interdisciplinary brain information research project. His research interests are in modeling and performance analysis of communication networks, especially the application of biologically inspired mechanisms to self-organization in future networks. He is a member of IEICE.

Masayuki Murata received the M.E. and D.E. degrees in Information and Computer Sciences from Osaka University, Japan, in 1984 and 1988, respectively. In April 1984, he joined IBM Japans Tokyo Research Laboratory, as a Researcher. From September 1987 to January 1989, he was an Assistant Professor with the Computation Center, Osaka University. In February 1989, he moved to the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University. In April 1999, he became a Professor of Osaka University, and since April 2004, he has been with the Graduate School of Information Science and Technology, Osaka University. He has contributed more than four hundred and fifty papers to international and domestic journals and conferences. His research interests include computer communication networks and performance modelling and evaluation. He is an IEICE Fellow, and a member of IEEE and ACM.