Bandwidth Guaranteed Method to Relocate Virtual Machines for Edge Cloud Architecture

Konomi Mochizuki, Hirofumi Yamazaki, and Akira Misawa
NTT Network Service Systems Laboratories
NTT Corporation
Musashino-shi, Tokyo, Japan
Email: mochizuki.konomi@lab.ntt.co.jp

***Abstract*—We are investigating an edge cloud architecture to achieve the flexible management and operation of the edge functions in a network. For developing an edge cloud, a scheme in which the edge functions are virtualized is effective. The virtualized edge function is achieved by carrying the edge function in a virtual machine (VM), and the VM live migration technology is used for relocating the VMs without disrupting services. Electrical power saving and efficient use of equipment are attained by using this technology for the resource management in the edge cloud. We have focused on the competition for bandwidth by user traffic and VM migration traffic that occurs when VMs are migrated in case that user traffic and VM migration traffic share a physical link. We propose a method to control bandwidth and a method to determine the order of VM migrations in order to complete all migrations while ensuring the minimum guaranteed bandwidth of user traffic. The results of evaluating the proposed methods revealed that it was possible to relocate VMs while maintaining the minimum guaranteed bandwidth of user traffic, which was not possible when the proposed methods were not used.**

I. INTRODUCTION

The edge functions typified by a session border controller (SBC) in Internet protocol (IP)-phone connections and a broadband access server (BAS) in Internet connections are carried on highly functional routers [1]. These routers handle many users' packets, and each router is fixed to a certain area. Since one highly functional router is needed even in an area with few users, equipment cost increases. Therefore, we propose an edge cloud architecture that enables efficient operation of the edge function by using cloud and virtualization technologies, which have progressed remarkably in data centers in recent years. When an edge function is set in an appropriate area for the right amount of equipment, the capital expenditure required is based on the actual demand, in contrast to the case where a user accommodation area is assigned to equipment in a fixed way. The edge functions referred to in this paper are assumed to be fundamental network services such as address translation, packet monitoring, and packet filtering. Moreover, further flexibility and reduced costs are achieved by carrying the edge function in a virtual machine (VM) on a general-purpose server. For example, when extending equipment in order to raise the throughput of a certain edge function, a new general-purpose server is introduced into the edge cloud, and extension becomes possible simply by using the clone creation function of the VM. Furthermore, when the number of users to be processed decreases at night, two or more VMs can be collected, i.e., allocated, to one server through live migration, and a reduction in electric power consumption can be achieved by turning off the other servers [2][3]. We propose here a VM relocation method to complete all migrations of multiple VMs while maintaining the guaranteed bandwidth of user traffic. This paper is organized as follows. The edge cloud architecture is described in Section 2, and the problems that occur in VM relocation are explained in Section 3. Our new methods are described in Section 4, and the evaluation results obtained through numerical simulation are reported in Section 5. Finally, Section 6 concludes the paper.

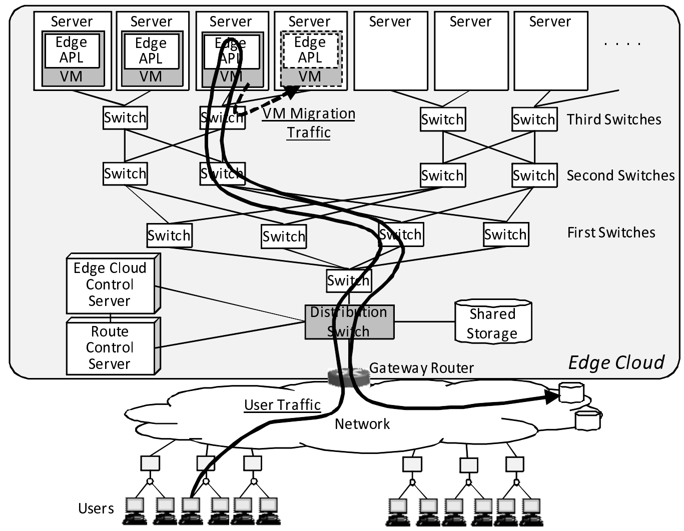

Fig. 1. Example of network composition in edge cloud

II. SYSTEM CONFIGURATION OF EDGE CLOUD

A. Network and Traffic in Edge Cloud

An example of network composition in the edge cloud is shown in Fig. 1. Two or more physical servers are connected to each other using switches. The switches construct a fat tree topology and maintain connectivity between physical servers and an external network, for example. Several different types of traffic such as the user traffic processed by a VM, management traffic, shared storage traffic, and VM migration traffic that occurs at the time of VM migration compose the traffic that is output from and input to a physical server. The cost increases if a physical link is prepared for exclusive use for each kind of traffic. Therefore, the development of 10 Gbps Ethernet has allowed different types of traffic to use a common physical

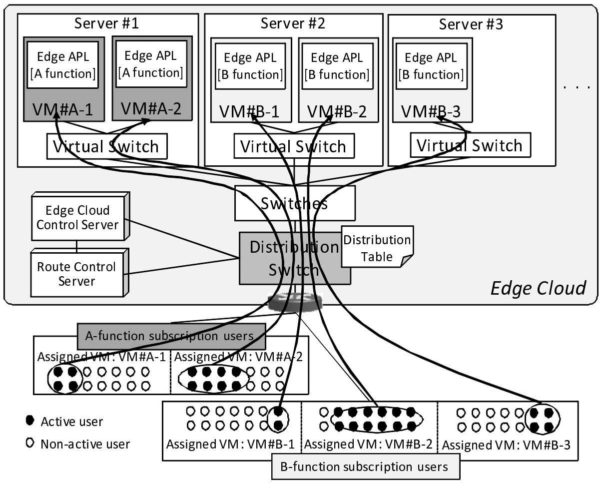

Fig. 2. Model of user traffic distribution to VMs

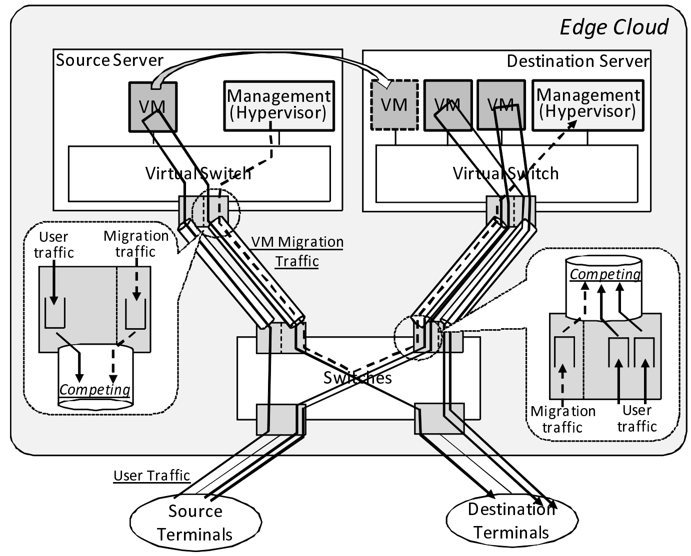

Fig. 3. Competition for bandwidth by user traffic and VM migration traffic

link in recent years [4][5].

B. The Use of VMs in Edge Cloud

Edge functions are run as applications in VMs in an edge cloud. Two or more VMs are prepared for one particular edge function, and user traffic is distributed at the distribution switch. If an edge cloud is compared to an edge router, the structure of the VMs in the edge cloud is the same as that of two or more service modules per existing function. The model of user traffic distribution to VMs is shown in Fig. 2. Users who subscribe to a service of the edge function are statically assigned to each VM, and the distribution switch holds the distribution table. If the packets of an active user who is currently communicating reach the distribution switch, the switch will distribute the packets to the VM according to the distribution table. Two or more VMs may run the same edge function, but even if the number of users assigned to each VM is the same, the number of active users that the VM is actually processing will vary from VM to VM.

For example, at night, if the number of active users decreases, the processing ability that each VM needs will also decrease. In this case, it is possible to turn off some servers by collecting the VMs to a specific server. On the contrary, when the number of active users increases, it is possible to distribute the VMs to some servers in order to raise the processing ability of each VM. VMs can be moved to other servers, and the processing of each VM can be continued by using VM live migration to relocate the VMs. This scheme achieves the flexible management of the server resource in the edge cloud.

III. Problems in VM Relocation

We describe in this section the problems that can occur when relocating two or more VMs in the edge cloud with the above-mentioned composition. For simplification, we only focus on user traffic and VM migration traffic sharing a physical link within the edge cloud.

A. Bandwidth Competition

The model of bandwidth competition between user traffic and VM migration traffic is shown in Fig. 3. After user traffic is input into a VM in the edge cloud from a source terminal and processing is carried out, it is output to a destination terminal. The same processing is performed in the VMs of the source server and the destination server. In this situation, the VM of the source server is migrated to the destination server. As shown in Fig. 3, in the physical interface of the source server, the user traffic that is bound for the destination terminal competes for bandwidth with the VM migration traffic which is output from the hypervisor. In the physical interface of the switches, the user traffic that is input to the edge cloud competes for bandwidth with the VM migration traffic. To secure the bandwidth for VM migration traffic, it is necessary to control the bandwidth of user traffic. If the bandwidth of all user traffic is restricted to zero at the time of VM relocation, the bandwidth for VM migration traffic can certainly be secured. However, it is necessary to avoid the situation where the edge cloud that offers fundamental network functions becomes a bottleneck. Here, the minimum guaranteed bandwidth is set for user traffic, and the condition is added that user traffic must not be less than this at the time of VM relocation. In this condition, we propose new methods to manage the bandwidth of the competing interfaces.

B. Mechanism of VM Migration and Problems

Here, the mechanism of VM migration is explained. VM migration is achieved by copying the memory information of a VM intended for migration from the source server to the destination server through a network [6][7][8]. Since the state of the edge function such as packet monitoring changes for every received packet, memory information changes while it is being copied. Therefore, the difference information that arose during copying needs to be transmitted again to the destination server. This is done repeatedly, and when the difference information becomes small enough, the VM of the source server is stopped, the last difference copy is performed, and the VM in the destination server is started. The size of the memory space the VM is using at the time a migration starts is set to M, the amount of state change per second

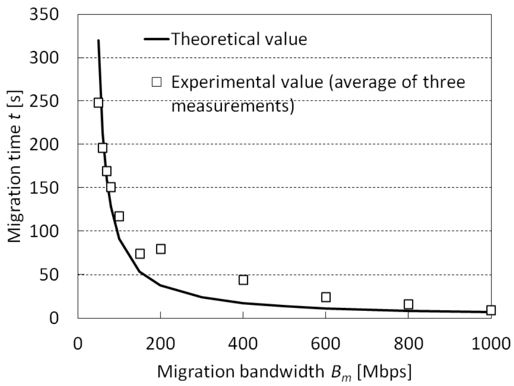

Fig. 4. Theoretical and experimental values of VM migration time

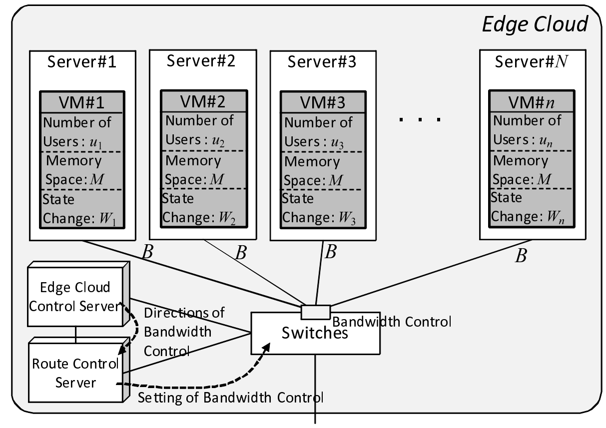

Fig. 5. Assumed model used in explanation of the proposed method

is set to W, and the time the migration takes is set to t; thus, the information $M + Wt$ will be transmitted to the destination server. The following equation must be satisfied in order to transmit the information of this quantity over the link of bandwidth B_m, and to complete the migration in t,

$$M + Wt = B_m t. \tag{1}$$

By transforming Eq. (1), the equation of time t that the migration takes is obtained as follows,

$$t = \frac{M}{B_m - W}. \tag{2}$$

Here, in order to show the validity of Eq. (2), the VM migration time was measured on an actual system, and the result is shown in Fig. 4. In this figure, the solid line shows the theoretical value acquired from Eq. (2), and the plots show the experimental values. In the experiment, we measured the VM migration time when a VM of $M = 4000$ Mbit and $W = 30$ Mbps was migrated from the source server to the destination server. The bandwidth B_m that can be used for VM migration was changed using the traffic shaping function in the virtual switch on the hypervisor of virtual software. In Fig. 4, since the experimental values show the same tendency as the theoretical values, it can be said that Eq. (2) is appropriate. In Eq. (2), because t cannot have a positive finite value when $B_m \leq W$, it is shown that the VM migration is impossible. Moreover, if the difference is small even if $B_m > W$, t will increase and the VM migration will take a long time.

When relocating two or more VMs, the number of VMs of the source server or destination server will change at every migration of a VM. Therefore, since the states of the bandwidth competition in Fig. 3 also differ at every VM migration, the bandwidth B_m that can be used for VM migration traffic will differ. Moreover, because the number of users varies from VM to VM as explained in Section 2, the amount of traffic to be processed also varies from VM to VM. As a result, W also varies from VM to VM. For example, in the timing to which B_m becomes the smallest, if a VM with a large W value is going to be migrated, the migration will take long time or will become impossible. To solve this problem, a method is needed to determine the order of VM migration in order to migrate VMs with suitable W values according to the changing B_m. Various studies have investigated ways to complete the VM migration quickly[9] and to optimize the migration schedule[10][11]; however, they have not considered VM migration when the user traffic and VM migration traffic use a common physical link.

IV. Proposed Methods

A. Model Definition

Before explaining the proposed methods, we define key elements including the assumed network composition and when VM relocation is carried out. The composition of the edge cloud before VM relocation is carried out is shown in Fig. 5. One VM operates on each server, and there is mutual connectivity between servers through switches. Each server has routes with the same bandwidth in both directions. In particular, the bandwidth that can be used by user traffic and VM migration traffic is set to B. The number of servers (VMs) is set to $N(= n)$. This is based on the assumption that the same application is carried on each VM, and the amount of memory space used by each VM is equal to M. The amount of state change per second in each VM shall be denoted by W_i =(user traffic bandwidth of each VM)$\times x + C$. Here, x is called the state change coefficient; x is depending on the packet processing method of the application or the OS carried in the VM, and it is assumed to be equal in all the VMs in this model. C is called state change constant; C is the amount of state change per second which occurs without received packets, and it is also assumed to be equal in all the VMs in this model. Moreover, the number of active users that each VM is processing is set to u_i. The maximum number of users being processed per server is B/G, where the minimum guaranteed bandwidth of user traffic is G. When the total number of users $\sum_{i=1}^{N} u_i$ becomes smaller than B/G, $n-1$ VMs are collected at one server. During the VM relocation, the number of users of each VM is assumed not to change a lot, because the relocation time is assumed to be several minutes or tens of minutes in this paper.

B. Bandwidth Control Method

Once it is determined that VM relocation will be carried out because of a reduction in the number of users, the fixed bandwidth B for user traffic and VM migration traffic will first be secured for every server. Then, shaping will be carried out at the minimum guaranteed bandwidth of G for all user traffic currently being output and input from each VM. The amount of state change per second W_i of each VM at this stage becomes $W_i = G \times u_i \times x + C$. Reservation of bandwidth B and shaping to the minimum guaranteed bandwidth G are achieved by a bandwidth control setup in the virtual switches in each server and switches. If the switches are flow switches such as open flow switches[12], the reservation and shaping are achieved by bandwidth control using flow identification. After the bandwidth control setup is completed, the VM is migrated according to the order determination method described in the following section. VM migration traffic can use the remaining bandwidth B_m after the bandwidth used by user traffic B_u, which is restricted to the minimum guaranteed bandwidth, has been subtracted from bandwidth B, i.e., $B_m = B - B_u$. However, since the number of users differs between the source server and the destination server, only the remaining bandwidth of the server that has a greater number of users can be used.

C. Method to Determine Order of VM Migration

1) Description of the Concept: First, the server in which the VMs will be collected among N servers is determined. This server is called the aggregation server. Since the VM that requires the largest bandwidth for migration is that with the largest amount of state change per second, i.e., the VM with the most users, the aggregation server is determined to be the server in which the VM with the most users is operating.

Next, the order of migration is determined for VMs other than the VM on the aggregation server. The principle of this method is shown in Fig. 6. When the distributed VMs are migrated to the aggregation server, because the user traffic is being distributed to each server at the time the relocation starts, the bandwidth available for VM migration traffic is large. However, when all the VMs are concentrated at the aggregation server, the competition for bandwidth by the user traffic input to the edge cloud in the aggregation server becomes very intense at the end of relocation. Therefore, the bandwidth that can be used for VM migration traffic is actually small. Based on this tendency, the order of migration is set sequentially starting with the VM that requires the largest bandwidth for migration, i.e., the VM with the largest $W_i(u_i)$. Therefore, the VM with the smallest W_i migrates at the end of relocation, which increases the possibility that all the VMs will be able to migrate.

However, there are some cases in which the relocation time is shorter when migration is carried out starting with the VM with the smallest u_i, depending on the condition. If the VM with large u_i is migrated, as mentioned above, in the early stage, the amount of user traffic in the aggregation server will increase rapidly. Since the bandwidth that can be used for VM migration decreases, even if a VM that migrates after that has a small $u_i(W_i)$ value, t will be long. On the contrary, if a VM with small u_i is migrated in an early stage, since the user traffic of the aggregation server does not increase rapidly and sufficient bandwidth remains for the VMs that migrate after that, t will not be too long. This phenomenon happens when the state change or the number of users in each VM is small, i.e., when there is enough bandwidth for VM migrations.

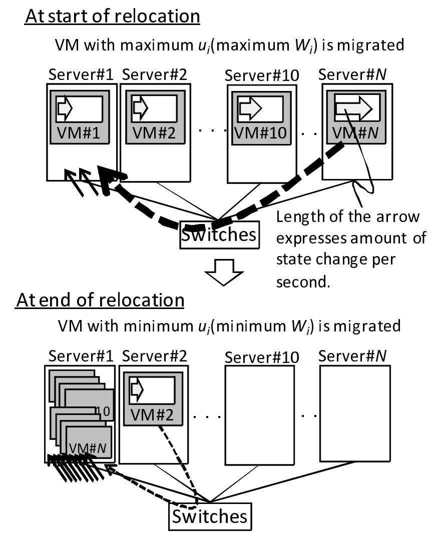

Fig. 6. Depiction of method to determine order of VM migration

2) Detailed Flow: The detailed flow of the VM migration order determination method is as follows.

1) VMs are arranged sequentially from (i) the VM with the largest u_i and from (ii) that with the smallest u_i.
2) In (i) and (ii), it is assumed that the aggregation server is the server that has (i) the VM with the largest u_i and (ii) the VM with the smallest u_i.
3) In (i) and (ii), the bandwidth that can be used for VM migration traffic for each VM migration is calculated in the order of VM migration. The calculated bandwidth values are set to $P_1 \sim P_{N-1}$ in (i) and set to $Q_1 \sim Q_{N-1}$ in (ii). Moreover, the number of users in each VM in order of VM migration is set to $R_1 \sim R_{N-1}$ in (i) and $S_1 \sim S_{N-1}$ in (ii).
4) In (i), it is judged whether there are one or more migrations that satisfy $P_i \leq (R_i G x + C)$. If there is one or more, it is judged that the VM relocation with this composition is impossible.
5) In (i) and (ii), the relocation times T_1 and T_2 are respectively calculated. In (i),

$$T_1 = \sum_{i=1}^{N-1} \frac{M}{P_i - (R_i G x + C)}. \tag{3}$$

In (ii),

$$T_2 = \sum_{i=1}^{N-1} \frac{M}{Q_i - (S_i G x + C)}. \tag{4}$$

6) When $T_1 < T_2$, the migration order is determined to be from the VM with the largest number of users to that with the smallest number of users. By contrast, when $T_1 > T_2$, the migration order is determined to be from the VM with the smallest number of users to that the largest number of users.

V. Efficiency Evaluation

In this section, we describe how we evaluated the efficiency of the proposed method by conducting a numerical simulation,

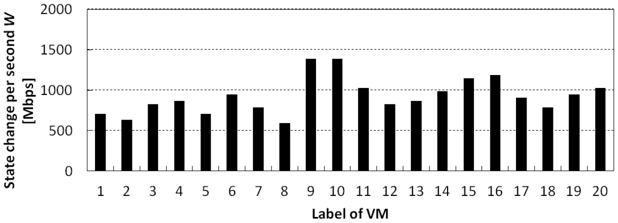

Fig. 7. Example of state change per second in each VM

and show the results of the evaluation.

A. Evaluation Conditions

The model shown in Fig. 5 was used in this evaluation. In a system in which there are $N = 20$ servers and one VM is operating on each server (i.e., the number of VMs is also 20), all the VMs are gathered at one server as the number of users decreases. In this situation, we evaluated the relocation time taken to migrate all the VMs. The bandwidth of the physical link of a server is 10 Gbps, and the bandwidth that can be used for user traffic and VM migration traffic in the 10 Gbps link is $B = 5$ Gbps. The minimum bandwidth guarantee is $G = 10$ Mbps, and the memory use space is $M = 32000$ Mbit in each VM. The distribution of the number of users in each VM shall follow a Poisson distribution. We compared the results of using the proposed method that takes the migration order into consideration, and a method that does not take the migration order into consideration (random method).

B. Evaluation Examples

An example evaluation result is shown for the case when the state change coefficient x was 4 and the state change constant C was 30 Mbps. The total number of users $\sum_{i=1}^{N} u_i$ was 450; that is, 450 users were distributed over all of the VMs according to a Poisson distribution of average 22.5. Since the maximum number of users per server was $B/G = 500$, the utilization efficiency of the aggregation server was 90% if 450 users were assigned to a server.

Fig. 7 shows the state change per second W_i of each VM when each user traffic is restricted to G. Transitions of the bandwidth B_m that can be used for VM migration and of the migration time of each VM are shown in Fig. 8. User traffic is concentrated in the aggregation server, and the bandwidth that can be used for VM migration decreases as the relocation progresses, as shown in the figure. Therefore, the migration time increases as the relocation progresses. Even in this situation, the proposed method enables the migration of all the VMs to be completed within a limited time, as shown in Fig. 8(a). However, in the random method, the last VM is not able to be migrated because of $B_m < W$, as shown in Fig. 8(b).

C. Results of Evaluation

The VM relocation time changes greatly with the state change coefficient x or the utilization efficiency of the aggregation server after VM relocation. Therefore, we focused on these factors when we evaluated the VM relocation time. Ten

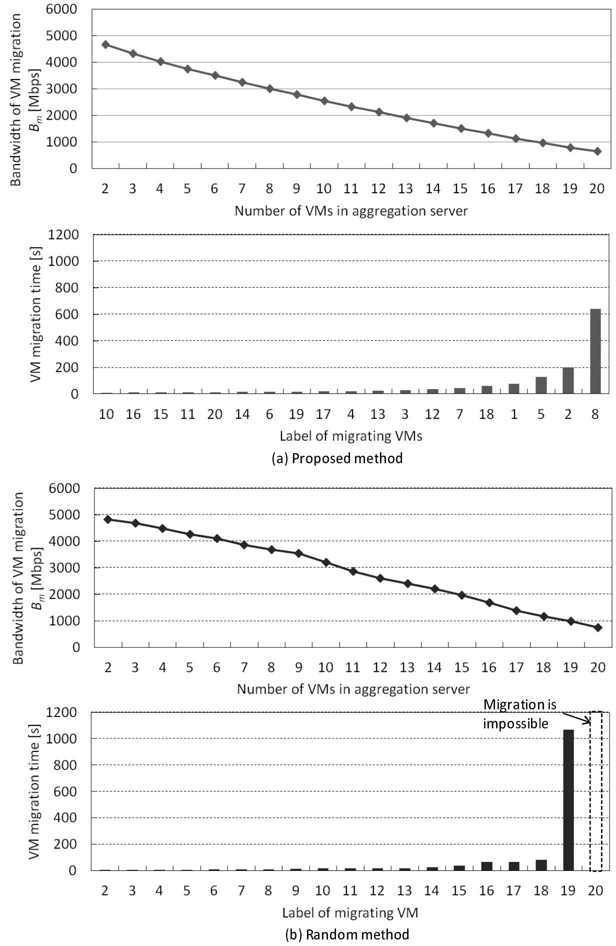

Fig. 8. Transitions of bandwidth which can be used for VM migration and migration time of each VM

samples of user distribution were prepared for one parameter condition. The evaluation described above was performed for the 10 samples. When a migration failure occurred even in just one sample, the VM relocation was considered to be impossible under that parameter condition. When the migrations of all the VMs were able to be completed for 10 samples, the average value of the VM relocation time was calculated.

The results of evaluating the VM relocation time for the parameter of sate change coefficient x are shown in Fig. 9. As x was large, the amount of VM state change per second was also large. Figures 9(a), (b), and (c) show the evaluation results when the utilization efficiency after relocation was 70%, 80%, and 90%, respectively. Since the amount of user traffic processed in the aggregation server increased when the utilization efficiency after relocation was large, the bandwidth used for VM migration was small, and relocation became difficult. In the proposed method, at 80% utilization efficiency, the VM relocation was possible even when coefficient x was as large as 7. In contrast, in the random method, if x became large, a domain that was not relocatable existed. Moreover, with the proposed method, relocation with 90% utilization efficiency was possible when x was 4; the average relocation time was 20 minutes. For example, the VM relocation with traffic change from day to night is possible.

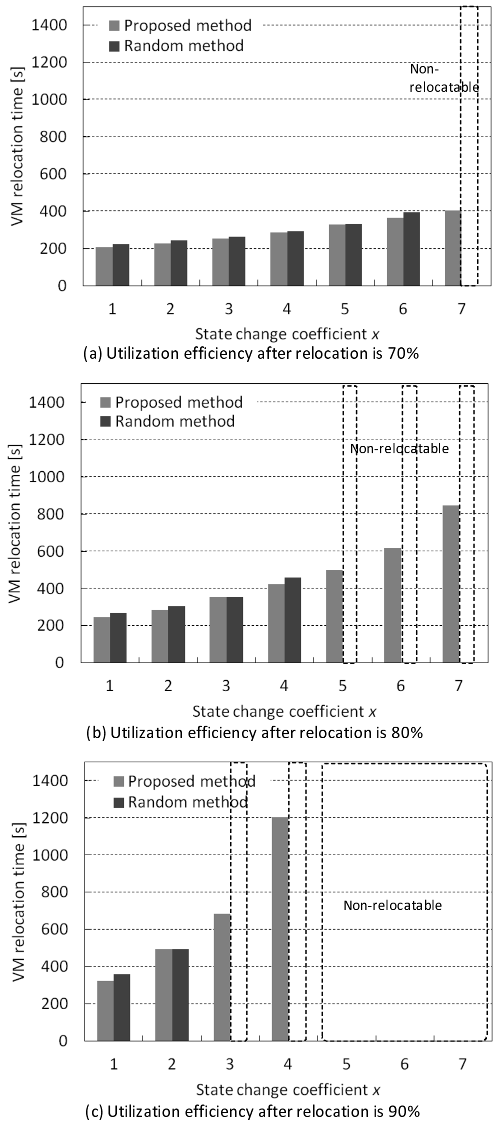

Fig. 9. Evaluation results of VM relocation time

Thus, in a composition in which VM relocation using the random method is not possible, it is possible using the proposed method, while still ensuring the minimum guaranteed bandwidth of user traffic. For example, in the above-mentioned composition with $x = 3$ or $x = 4$, although the VM relocation of one server and 90% utilization efficiency is possible with the proposed method, only VM relocation of two servers and 45% utilization efficiency can be achieved with the random method. Unless a waiting period is held until the number of users decreases further, the relocation to one server cannot be performed using the random method. In cases where such a system would actually be used, since there are dozens of systems that have 20 servers, the potential electric power reduction effect using the proposed method is large, which will significantly contribute to reducing costs.

VI. Conclusion

We have developed an edge cloud architecture in which edge functions are installed on VMs. We proposed a VM relocation method that enables migration of all VMs while ensuring the minimum guaranteed bandwidth of user traffic. In the proposed method, we take into account the order of the VM migrations. When the amount of the bandwidth which can be used for the VM migration is large, the VM with the large amount of changes of state is migrated. On the other hand, when the amount of the bandwidth which can be used for the VM migration is small, the VM with the small amount of changes of state is migrated. In the composition in which relocation is not possible by using the random method that does not take a VM migration order into consideration, relocation is possible by the proposed method. This was demonstrated by numerical simulation. The proposed method is effective when carrying out certain VM relocation for a short time for the purpose of achieving reductions in electric power consumption in the edge cloud.

References

[1] K. Knightson, N. Morita, and T. Towle, "NGN architecture: generic principles, functional architecture, and implementation," Communications Magazine, IEEE, Vol.43, Issue10, pp.49-56, October 2005.

[2] C. Fragni and L. Costa, "ECO-ALOC: Energy-efficient resource allocation for cluster-based software routers," Computer Networks, Elsevier, Vol. 56, Issue 9, pp.2249-2261, March 2012.

[3] T. Wood, P.Shenoy, A. Venkataramani, and M. Yousif, "Black-box and gray-box strategies for virtual machine migration," Proc. 4th Symposium on Networked Systems Design and Implementation, Cambridge, UK, pp.229-242, April, 2007.

[4] IEEE 802.1Qaz, "Enhanced Transmission Selection," http://www.ieee802.org/1/pages/802.1az.html, accessed May 15. 2013.

[5] IEEE 802.1Qbb, "Priority-based Flow Control," http://www.ieee802.org/1/pages/802.1bb.html, accessed May 15. 2013.

[6] C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," Proc. 2nd Symposium on Networked Systems Design and Implementation, Boston, USA, pp.273-286, May 2005.

[7] M. Nelson, B. H. Lim, and G. Hutchins, "Fast transparent migration for virtual machines," Proc. USENIX Annual Technical Conference, Anaheim, USA, pp.391-394, April 2005.

[8] A. Mirkin, A. Kuznetsov, and K. Kolyshkin, "Containers checkpointing and live migration," Proc. the Linux Symposium, Ottawa, Canada, pp.85-90, July 2008.

[9] Z. Liu, W. Qu, W. Liu, and K.Li, "Xen Live Migration with Slowdown Scheduling Algorithm," Proc. 2010 International Conference on PDCAT, pp.215-221, Wuhan, China, Dec. 2010.

[10] A. Stage and T. Setzer, "Network-aware migration control and scheduling of differentiated virtual machine workload," Proc. Software Engineering Challenges of Cloud Computing, Vancouver, Canada, pp.9-14, May 2009.

[11] K. Yang, J. Gu, T. Zhao, and G. Sun, "An Optimized Control Strategy for Load Blancing based on Live Migration of Virtual Machine," Proc. Sixth Annual China Grid Conference, Dalian, China, pp.141-146, August 2011.

[12] Open Networking Foundation, http://www.openflow.org/, accessed May 15. 2013.