EVALUATION OF CONGESTION CONTROL METHODS FOR JOINT MULTIPLE RESOURCE ALLOCATION IN CLOUD COMPUTING ENVIRONMENTS

Shin-ichiKuribayashi

Department of Computer and Information Science, Seikei University, Japan

ABSTRACT

As cloud computing provides not only services that have been traditionally provided on the Internet but also many other services, it has a dramatically higher risk than conventional networks that an occurrence of congestion in one service leads to congestion in other services. Unlike conventional networks, cloud computing environments should provide not only bandwidth but also processing ability simultaneously.

First, this paper compares two congestion control methods (Methods A and B)in cloud computing environments, assuming that multiple types of resource are allocated simultaneously, and clarifies the effective areas of two congestion control methods with computer simulations. Method A postpones the service completion time by delaying resource allocation. Method B reduces the size of required resource and allocates to the request, extending in turn the duration of resource allocation so that the total amount of resource required by the request will be satisfied. The effective areas of two congestion control methods are clarified with computer simulations.

Then, this paper compares three control methods (Methods 1, 2 and 3) tocope with the excessive generation of requests from a specific access point, which results in the degradation in service quality of requests from other access points, and clarifies the effective areas of three control methods with computer simulations. Method 1 allocates minimum resources dedicated to each access point in each center. Method 2reduces the size of required resources of requests from a specific access point, and Method 3 thins out some of requests from a specific access point.

KEYWORDS

Congestion control, Joint multiple resourceallocation, Resource management, cloud computing environments

1. INTRODUCTION

Cloud computing services are rapidly gaining in popularity. They allow the user to rent, only at the time when needed, only a desired amount of computing resources (processing bility and storage capacity) out of a huge mass of distributed computing resources without worrying about the locations or internal structures of these resources [1]-[16]. In cloud computing environments, it has a dramatically higher risk than conventional networks that an occurrence of congestion in one service leads to congestion in other services, and that abnormal traffic in one service (including spam) degrades the quality of other services. Unlike conventional networks, it is necessary to simultaneously allocate both processing ability and network bandwidth needed to access it, and handles different services having different QoS requirements in an integrated

DOI: 10.5121/ijcnc.2017.9303

manner. Therefore, the conventional congestion control method is not applicable to cloud computing environments.

The authors have identified basic congestion control concepts for cloud computing environments, assuming that multiple types of resource are allocated simultaneously [5]. The concepts were derived from the investigation and analysis of conventional network congestion control methods. To implement these concepts, we have proposed the new congestion control method ("**Method B**" hereinafter) that reduces the size of required resource and allocates to the request, thereby increasing the number of requests that can be processed and resource utilization [17]. This method extends in turn the duration of resource allocation so that the total amount of resource required will be satisfied. Thismeans that it takes more time for a service to be completed. We have also proposed another method ("**Method A**" hereinafter), which also postpones the service completion time [18],[19]. If the required multiple types of resource are not available to meet the service request at the arrival of a request, this method delays resource allocation for the request until required resources are available. Although this method was originally not intended for the congestion control, it could relieve the congestion. This paper compares and clarifies the effective areas of two congestion control methods with computer simulations.

Moreover, the authors have proposed the control method to cope with the excessive generation of requests from a specific access point (which results in the degradation in service quality of requests from other access points) [20]. This paper clarifies the effective areas of the proposed method compared with other two possible control methods.

The rest of this paper is organized as follows. Section 2 explains related works. Section 3 compares two congestion control methods, Methods A and B, and clarifies the effective areas of each method with computer simulation. Section 4 compares three control methods which cope with the excessive generation of requests from specific access point. Then, the effective areas of each method is clarified with computer simulations. Finally, Section 5 gives the conclusions. This paper is an extension of the study presented in References[19] and [20].

2. RELATED WORK

Resource allocation for a cloud computing environment has been studied very extensively in References [6]-[20]. References [6] and [7] have proposed the market-oriented allocation of resources including auction method. Reference [8] has proposed to use game-theory to solve the problem of resource allocation. Automatic or autonomous resource management in cloud computing have been proposed in References [9] and [10]. Reference [11] has presented the system architecture to allocate resources assuming heterogeneous hardware and resource demands. Energy-aware resource allocation methods have been proposed in References [12] and [13].Reference [14] has proposed a formalism based on the Event-B language for specifying Cloud resource allocation policies in business process models and analyzed its correctness according to user requirements and resource capabilities. Reference [15] has proposed a resource allocation mechanism for machines on the cloud, based on the principles of coalition formation and the uncertainty principle of game theory. Reference [16] has proposed dynamic resource allocation with the combinational auction model.

References [17], [18] and [19] have proposed two congestion control methods, assuming that multiple types of resource are allocated simultaneously. One is to reduces the size of required resource for congested resource type and extends in turn the duration of resource allocation so that the total amount of resource required by the request will be satisfied. The other is to postpones the service completion time by delaying resource allocation. It was demonstrated by simulation evaluations that the first method is suitable for services in which a minimum size of

resource needs to be allocated or for services in which there are many requests that do not require a large extension of the duration of resource allocation. Reference [20] has proposed the cloud resource allocation which supposes multiple heterogeneous resource-attributes for each resource-type. For example, resource-attributes of bandwidth are network delay time, packet loss probability, required electric power capacity, etc.

This paper tries to clarify the effective areas of two congestion control methods which have been proposed in References [17]-[19]. One is to postpone the service completion time by delaying resource allocation, and the other is to reduce the size of required resource for congested resource type. This paper also tries to clarify the effective areas of three control methods, discussed in Reference [20], to prevent the degradation in service quality of requests from other access points when requests from a specific access point occur more than expected.

3. COMPARISON OF CONGESTION CONTROL METHODS FOR CLOUD COMPUTING ENVIRONMENTS

3.1 Resource allocation Model For Cloud Computing Environments

Figure 1 illustrates the system model for cloud computing environments. Figure 2 illustrates the resource allocation model based on the system model in Figure 1. These are the same as the models described in References [17]-[20]. There are k different resource sets, and each resource set in each center has some processing ability and bandwidth. When a request is generated, the optimal resource set is selected from among k resource sets, and processing ability and bandwidth is allocated simultaneously from the same selected set. It is not allowed to allocate processing ability from one resource set and bandwidth from another resource set.

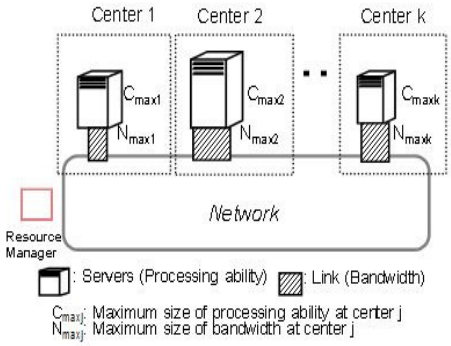


Figure 1. System model for cloud computing services

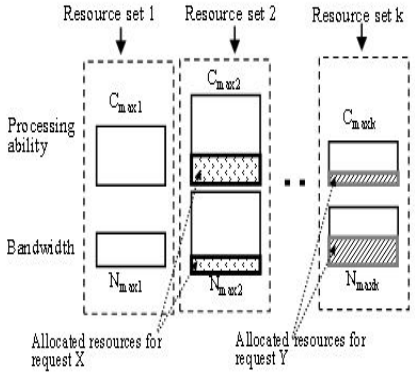


Figure 2. Resource allocation model

3.2 Resource Allocation Algorithms

3.2.1 Main Differences

Main differences between Methods A and B are illustrated in Figure 3. If the required amount of both processing ability and bandwidth are not available to meet a request, Method A delays resource allocation (keeps the request waiting) while Method B reduces the size of required resource and allocates to the request, extending the duration of resource allocation (namely, allocating a smaller size of resource over a long period of time). H, X, q, L and M in Figure 3 are duration of resource allocation, size of requested resource, resource size reduction rate, maximum permissible service completion time and allocation duration rate (defined in section 3.2.3) respectively.

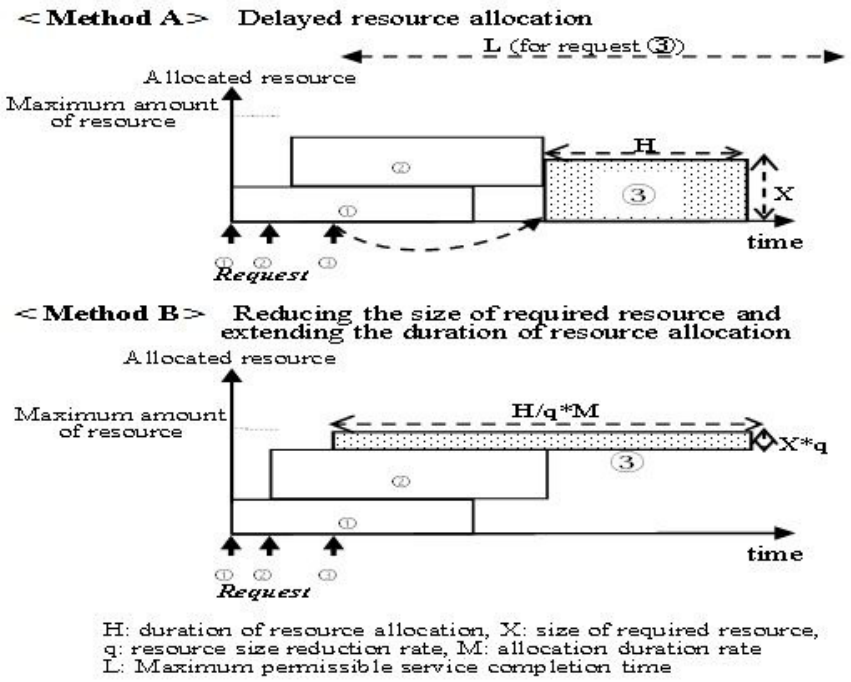


Figure 3.Main differences between Method A and Method B

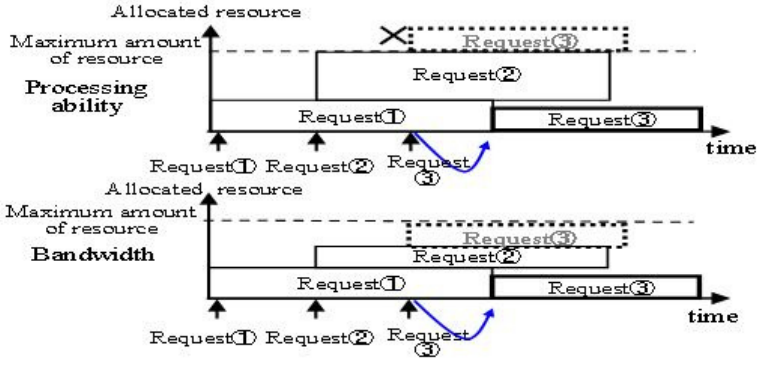


Figure 4.Resource allocation image by method A

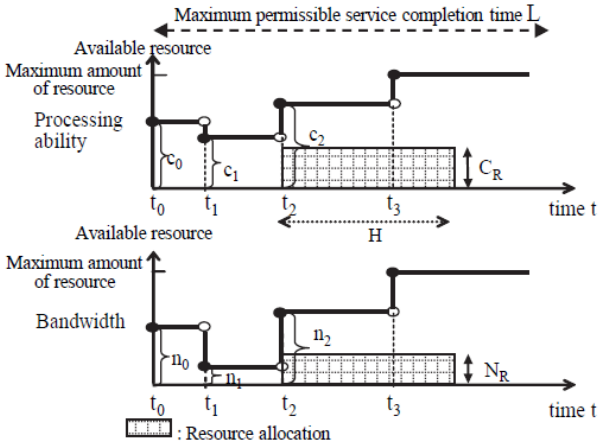
Although Figure 3 shows only one resource type for the sake of simplification, both two types of resource (processing ability and bandwidth) are allocated simultaneously in Methods A and B. As shown in Figure 4, Method A delays resource allocation because of the lack of available processing ability when Request ③ is generated although bandwidth is available. In the same situation, Method B reduces the size of both processing ability and bandwidth at the same ratio and allocates to the request. Not that any request should complete its service within L in Methods A and B.

3.2.2 Resource Allocation Algorithm Of Method A

The resource allocation algorithm is outlined below. Refer to Reference [18] for detail.

1. If the required multiple types of resources are available to meet the service request at the arrival of a request, resources are allocated to it normally.

2. If the required multiple types of resources are not available to meet the service request, the request is handled as follows. The system estimates the time when the requests will be completed and sufficient sizes of all types of resource will become available with the resource management chart as illustrated in Figure 5. In this figure, t_0 means the time when a request is generated, at t_1, new resource allocation starts for another request, and at t_2 and t_3, resources allocated to other requests are released. At the time of $t=t_0$ or $t=t_1$, bandwidth is not available although processing ability is available for the duration of resource allocation. As both types of resource are available at t_2, the time of $t=t_2$ is chosen as the appropriate start time of resource allocation. It then tentatively allocates resources to the request from the time of $t=t_2$ so that they will become available to the request. When the time comes for the tentative allocation is to take effect, the resources are actually allocated.



t_0: the time when a request generates
t_i (i=0~x) : the time when an new resource allocation starts for another request or resources allocated to other requests are released
t_x: the last t_i until t_0+L
C_R:Size of required processing ability
N_R:Size of required bandwidth

Figure 5. Resource management chart to decide the start time of resource allocation in Method A

3.2.3 Resource Allocation Algorithm Of Method B

The resource allocation algorithm is outlined below. Refer to Reference [17] for detail.

1. If processing ability is congested, Method B reduces the size of required processing ability for any request which requires a size of processing ability above a certain level. The duration of resource allocation is extended in turn so that the total amount of resources required will be satisfied. The service should be completed within L.

2. If bandwidth is congested, the request is handled in the same manner as described above.

3. The user registers in advance the resource size reduction rate, q ($0<q\leqq 1$). That is, the size will be reduced to q*W when W is the size of required resource.

Extending the duration of resource allocation and thus providing the required amount of resource is suitable for a file transfer service. However, for a streaming delivery service, it may not necessary to extend the duration of resource allocation. In such a service, the reduction of resource size results in the degradation of QoS. To evaluate the effect of extending the duration of resource allocation, an allocation duration rate, M, defined in the following expression, is proposed:

$$M=H_1/[H/q] \quad (q\leqq M\leqq 1) \tag{1}$$

In this expression H_1 is the actual duration of resource allocation.

3.2.4 Simulation Evaluation

3.2.4.1 Evaluation model

1. The evaluation is performed by a computer simulation. The simulation model is based on Figure 2 with k=2. That is, there are two resource sets. Servers at each center, a network for connecting to the servers, user access points, and user requests are virtually realized on a UNIX-based computer. The simulation program using C language was created by ourselves, not the existing products.

2. The sizes of processing ability and bandwidth follow a Gaussian distribution (with its variance=1), with their averages being C and N, respectively. The actual sizesof resource requested are C_r and N_r, respectively.

3. The intervals between requests follow an exponential distribution with the average, r. The duration of resource allocation, H, is constant.

4. The maximum permissible service completion time L($\geqq$H)is constant.

5. The pattern in which requests occur is a repetition of {$C=a_1$, $N=b_1$; $C=a_2$, $N=b_2$; …; $C=a_w$, $N=b_w$} , where w is the number of requests within one cycle of repetition, a_u(u=1～w) is the size of C of the u-th request, and b_u (u=1～w) is the size of N of the u-th request.

3.2.4.2 Simulation Results And Evaluation

The simulation results are shown in Figures 6 to 8. Figure 6 shows the maximum generation

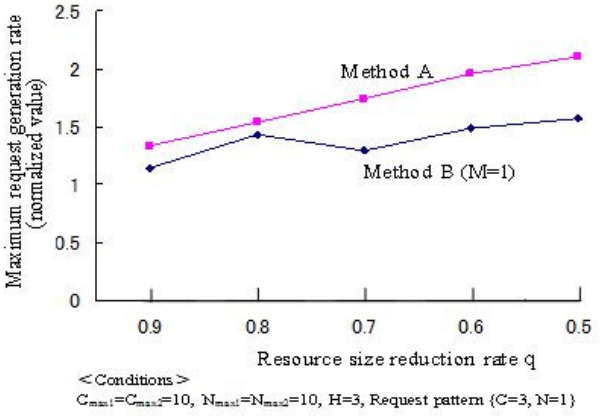


Figure 6.Evaluation of Maximum request generate rate

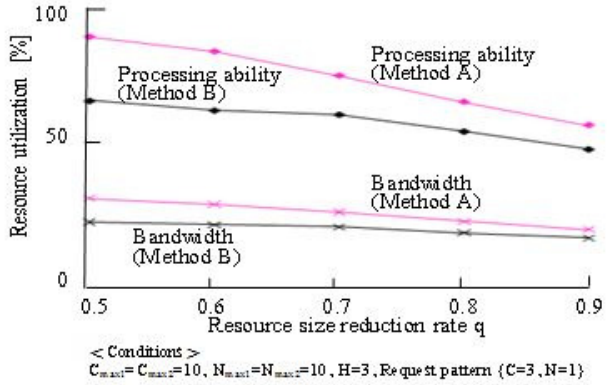


Figure 7.Evaluation of Resource utilization

rate of requestwhich ensures that the average request loss probability is less than 5%. It is supposed that L is given by H/q for both Methods A and B (L becomes large as the value of q becomes small (namely, as the size of resource allocated is reduced)). The maximum generation rate of request will increase as the value of q becomes small, even for Method A. Figure 7 shows resource utilization of Methods A and B under the same conditions as in Figure 6. Figure 8 shows how the maximum request generation rate is affected by the ratio of the number of requests with M=1 to the number of requests with M=q, α, in Method B. As in Figure 6, Figure 8 supposes that the average request loss probability isless than 5%.

The following points are clear from these figures:

1. If the maximum permissible service completion time L is equal for both Methods A and B, Method A is always more effective than Method B. In other words, delaying resource allocation without reducing the size of required resource makes it possible to handle more requests and use both types of resources more efficiently than reducing the size of required resource.

As shown in Figure 9, Method B allocates resources immediately at the arrival of a request. Even if the size of resource allocated is small, the service completion time would exceed L,

There by making it necessary to discard subsequent requests. On the other hand, the service would be completed within L in Method A.

2) As the ratio of the number of requests which do not require a long extension of the duration of resource allocation (namely, M is low), Method B becomes able to handle more requests than Method A even when L is the same for both methods. In the example of Figure 8, Method B can handle 1.3 times more requests than Method A when α=0.5. This is because the amount of resource requested is smaller than the originally required amount of resource when M is low in Method B.

From the above evaluation, it can be concluded that Method B will be suitable for services in which a minimum size of resource needs to be allocated at the arrival of a request or for services in which there are many requests that do not require a large extension of the duration

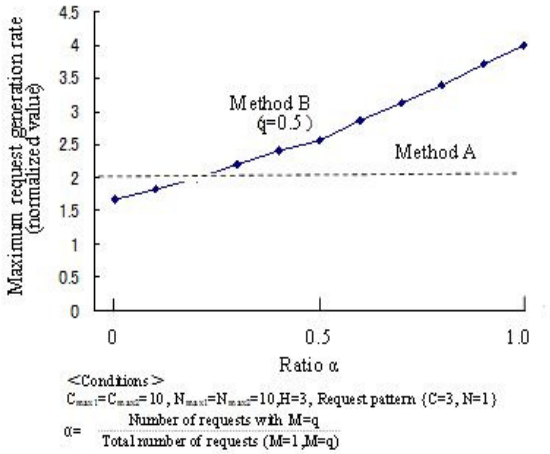


Figure 8.Evaluation of Maximum request generate rate

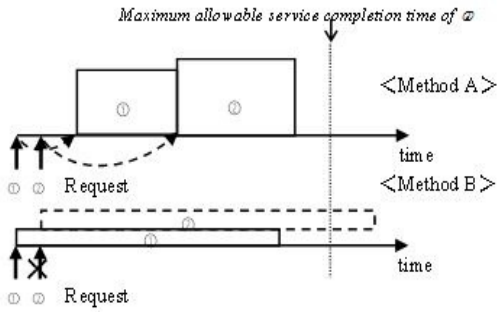


Figure 9.Comparison between Method A and Method B

of resource allocation when the size of required resource will be reduced. Method A is suitable for all other services. However, since the values of L and q are dependent on the user's service requirements, these factors should be also taken into consideration in determining the effective area of Methods A and B.

4. COMPARISON OF METHODS TO COPE WITH THE EXESSIVE GENERATION OF REQUESTS FROM SPECIFIC ACCESS POINT

4.1 Methods To Cope With The Exessive Generation Of Requests

If requests from a specific access point in the cloud computing environment occur more than expected, the service quality of requests from other access points will be deteriorated. The authors have proposed the control method to cope with the excessive generation of requests from specific access point (point **γ**)[20]. This method, **Method 1,**allocates minimum resources dedicated to each access point in each center. One other possible control method is to reduce the size of required resources of requests from point **γ** (**Method 2**), in the same way as Method B in Section 3. The size of required resources of requests from point γ will be reduced to q*W when W is the size of required resource. q ($0<q\leqq1$) is the resource size reduction rate. The other method is to thin out s[%] of requests from point γ automatically (Method 3).s ($0\leqq s<100$) is the request blocking rate. Another possible control method is to thin out some of requests from a specific access point (**Method 3**).

As far as service quality of request from other access points is guaranteed, qand s should be kept as high as possible, in order to avoid largely deteriorating service quality of requests from point **γ**.

4.2 Evaluation Results And Discussions

4.2.1 Simulation Evaluation Model

Figure 10 illustrates the resource allocation model which is the same model adopted Reference [20]. And the following simulation conditions are supposed:

i) The network delay and service processing time are as follows:
$2d_{x1}=50$, $2d_{x2}=150$, $2d_{y1}=150$, $2d_{y2}=50$; $h_{x1}=1000$, $h_{x2}=500$,$h_{y1}=1000$, $h_{y2}=500$
Network delay and processing time are assumed to be constant distribution.

ii) The amount of resources is calculated with the algorithm proposed in [20] as follows:
$x_1=40$, $y_1=40$, $xy_1=10$, $x_2=20$, $y_2=20$, $xy_2=10$
It is noted that x_1& $y_1$and x_2& $y_2$will be added to xy_1, xy_2 respectively for Method 2 and Method 3.

iii) It is assumed that requests from access point Y in Figure 10 occur two times more than expected

Other simulation conditions are the same as those in Section 3.2.4

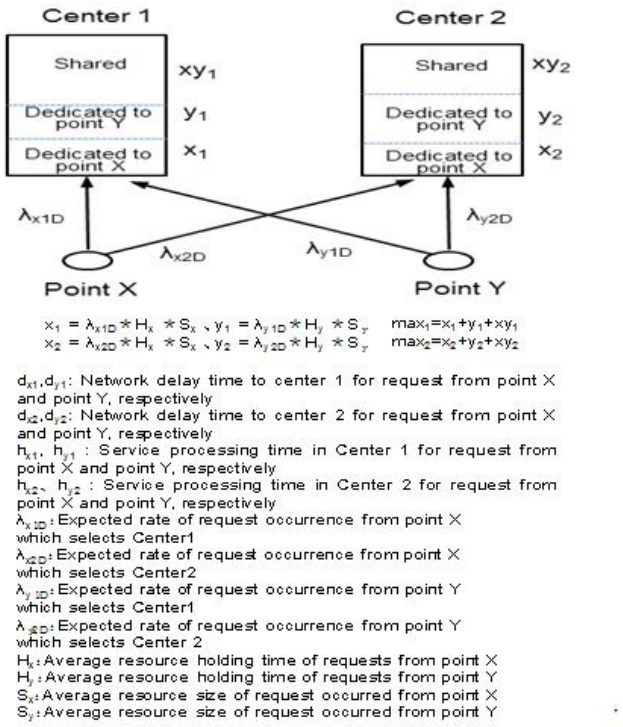


Figure 10.Resource allocation model

4.2.2 SIMULATION RESULTS AND EVALUATION

The simulation results are shown in Figures 11 and 12. Figure 11 compares Method 1 and Method 2, and shows how the average request loss probability of requests from point X is affected by resource size reduction rate **q**. Figure 12 compares Method 1 and Method 3, and shows how the average request loss probability of requests from point X is affected by request blocking rate **S**.

It is clear in these example that three methods can expect the same effect when q=0.75 or s=65%. Method 1 can guarantee the minimum service quality without measuring the real time number of requests occurring from each access points. However, Method 1 would produce the division loss of resources (i.e., resources are not utilized efficiently) when the number of access points increases. Although it is necessary for Method 2 and Method 3 to continue measuring the real time number of requests occurring from each access points, the division loss of resources will not occur (i.e., resources are utilized more efficiently than Method 1) even if the number of access point increases. In addition, Method 2 could process more requests by degrading QoS, compared with Method 3 and Method 1. However, Method 2 cannot be applied to all services.

In this way, three control methods discussed in this Section can cope with the excessive generation of requests from specific access point, and it is desirable to select one optimal method according to the system and service conditions.

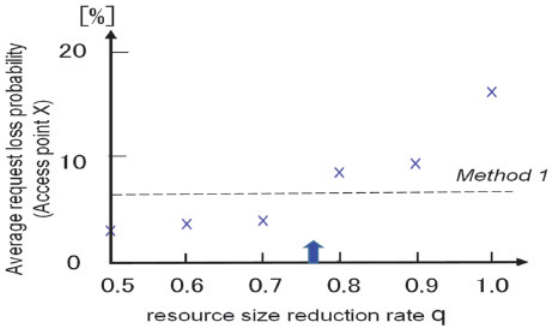


Figure 11. Evaluation result of Method 1 and Method 2

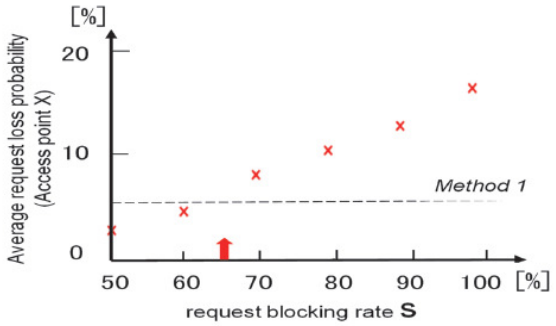


Figure 12. Evaluation result of Method 1 and Method 2

5. CONCLUSIONS

This paper has evaluated two congestion control methods (Methods A and B) in cloud computing environments, assuming multiple types of resource are allocated simultaneously. In case of congestion, Method A delays resource allocation but allocates the requested size of resource. Method B reduces the size of required resource and allocates to the request, extending in turn the duration of resource allocation. It was demonstrated by simulation evaluations that Method B is suitable for services in which a minimum size of resource needs to be allocated or for services in which there are many requests that do not require a large extension of the duration of resource allocation when the size of required resource will be reduced. Method A is suitable for all other services.

Next, this paper compares three control methods (Methods 1, 2 and 3) to cope with the excessive generation of requests from a specific access point. Method 1 allocates minimum resources dedicated to each access point in each center. Method 2 reduces the size of required resources of requests from a specific access point, and Method 3 thins out some of requests from a specific access point.It was demonstrated by simulation evaluations that all three control methods could prevent the degradation in service quality of requests. It is required to select the most suitable method according to the system and service conditions.

Since the model used for evaluation contained only limited numbers of access points and centers, it is required to evaluate the effectiveness of the proposed method and to identify the conditions under which the method is effective, assuming more access points and centers.

REFERENCES

[1] G.Reese: "Cloud Application Architecture", O'Reilly& Associates, Inc., Apr. 2009.

[2] P.Mell and T.Grance, "Effectively and securely Using the Cloud Computing Paradigm", NIST, Information Technology Lab., July 2009.

[3] V. Vinothina, R.Sridaran, and P. Ganapathi, "A Survey on Resource Allocation Strategies in Cloud Computing", International Journal of Advanced Computer Science and Applications,Vol. 3, No.6, 2012.

[4] S.Kuribayashi,"Optimal Joint Multiple Resource Allocation Method for Cloud Computing Environments", International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 2, No.1, Feb. 2011.

[5] K.Hatakeyama and S.Kuribayashi, "Proposedcongestion control method for all-IP networks includingNGN", ICACT2008 (2008.2)

[6] S.Kuribayashi,"Joint Multiple Resource Allocation Method for Cloud Computing Services with different QoS to users at multiple locations", International journal of Computer Networks & Communications (IJCNC), Vol.5, No.5, pp.1-18, Sep. 2013.

[7] B. Soumya, M. Indrajit, and P. Mahanti, "Cloud computing initiative using modified ant colony framework," in In the World Academy of Science, Engineering and Technology 56, 2009.

[8] R.Buyya, C.S. Yeo, and S.Venugopal, "Market-Oriented Cloud Computing:Vision, Hype, and Reality for Delivering IT Services as Computing Utilities", Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08), Sep. 2008.

[9] G.Wei, A.V. Vasilakos, Y.Zheng, and N.Xiong, "A game-theoretic method of fair resource allocation for cloud computing services", The journal of supercomputing, Vol.54, Issue 2, Nov. 2010.

[10] Yazir, Y.O., Matthews, C., Farahbod, R., Neville, S., Guitouni, A., Ganti, S., and Coady, Y., "Dynamic Resource Allocation in Computing Clouds through Distributed Multiple Criteria Decision Analysis", 2010 IEEE 3rd Internatiuonal Conference on Cloud Computing (CLOUD 2010), July 2010.

[11] B.Malet and P.Pietzuch, "Resource Allocation across Multiple Cloud Data Centres", 8th International workshop on Middleware for Grids, Clouds and e-Science. (MGC'10), Nov. 2010.

[12] B. Rajkumar, B. Anton, and A. Jemal, "Energy efficient management of data center resources for computing: Vision, architectural elements and open challenges," in International Conference on Parallel and Distributed Processing Techniques and Applications, July 2010.

[13] M. Mazzucco, D. Dyachuk, and R. Deters, "Maximizing Cloud Providers' Revenues via Energy Aware Allocation Policies," in 2010 IEEE 3rd International Conference on Cloud Computing. IEEE, 2010.

[14] M.Graiet, A.Mammar, S.Boubaker, and W.Gaaloul,"Towards Correct Cloud Resource Allocation in Business Processes,"IEEE Transactions on Services Computing, Vol.10, Issue1, pp.23-36, July 2016.

[15] P.S. Pillai and S.Rao, "Resource Allocation in Cloud Computing Using the Uncertainty Principle of Game Theory," IEEE Systems Journal Vol. 10, Issue 2, June 2016.

[16] E.I.Nehru, J.I.S.Shyni, R.Balakrishnan, "Auction based dynamic resource allocation in cloud, " International Conference on Circuit, Power and Computing Technologies (ICCPCT 2016), Mar. 2016.

[17] Shin-ichi Kuribayashi, "Proposed congestion control method for cloud computing environments", International Journal of Computer Networks & Communications (IJCNC) Vol.3, No.5, pp.161-176, Sep 2011.

[18] S.Tsumura and S.Kuribayashi, "Delayed resource allocation method for a joint multiple resource management system", APCC2007, TPM2-3 (2007.10)

[19] Takahiro Yoshino and Shin-ichi Kuribayashi, "Evaluation of congestion control methods for joint multiple resource allocation", Proceeding of the 13-th International Conference on Network-Based Information Systems (NBiS-2010), pp.94-97, Sep. 2010.

[20] S.Kuribayashi, "Resource Allocation Method for Cloud Computing Environments with Different Service Quality to Users at Multiple Access", International Journal of Computer Networks & Communications (IJCNC) Vol.7, No.6, pp.33-51, Nov.2015.

AUTHOR

Shin-ichiKuribayashi received the B.E., M.E., and D.E. degrees from Tohoku University , Japan, in 1978, 1980, and 1988 respectively. He joined NTT Electrical Communications Labs in 1980. He has been engaged in the design and development of DDX and ISDN packet switching, ATM, PHS, and IMT 2000 and IP-VPN systems. He researched distributed communication systems at Stanford University from December 1988 through December 1989. He participated in international standardization on ATM signaling and IMT2000 signaling protocols at ITU-T SG11 from 1990 through 2000.Since April 2004, he has been a Professor in the Department of Computer and Information Science, Faculty of Science and Technology, Seikei University. His research interests include optimal resource management, QoS control, traffic control for cloud computing environments and green network. He is a member of IEEE, IEICE and IPSJ.