Chapter 6
Fuzzy-Based Resource Reallocation Scheduling Model in Cloud Computing

Jaekwon Kim, Taeyoung Kim, Minoh Park, Youngshin Han and Jongsik Lee

Abstract A cloud computing system consists of physical resources for processing large-scale tasks. With a recent trend of rapidly growing data, a cloud computing system needs a processing method to process a large-scale task in a physical resource. Generally, a physical resource divides a requested large-scale task to several tasks. And a processing time of each divided task varies with two factors which are processing efficiency of each resource and distance between resources. Although a resource completes a task, the resource is standing by until all divided tasks are completed. When all resources complete a large-scale task, each resource can start to process a next task. In this paper, we propose a Fuzzy-based Resource Reallocation Scheduling Model (FRRSM). Using fuzzy rule, FRRSM reallocates an uncompleted task to with a resource in considering efficiency and distance factors of the resource. FRRSM is an efficient method for processing a large-scale task or multiple large-scale tasks.

Keywords Cloud computing · Resource reallocation · Fuzzy

J. Kim · T. Kim · M. Park · J. Lee (✉)
Department of Computer and Information Engineering, Inha University, Inchon, South Korea
e-mail: jslee@inha.ac.kr

J. Kim
e-mail: jaekwonkorea@naver.com

T. Kim
e-mail: silverwind@gmail.com

M. Park
e-mail: park-min-o@nate.com

Y. Han
Division of Liberal Arts and Teaching (Information and Communication), Sungkyul University, Anyang, South Korea
e-mail: hanys@sungkyul.ac.kr

J. J. (Jong Hyuk) Park et al. (eds.), *Frontier and Innovation in Future Computing and Communications*, Lecture Notes in Electrical Engineering 301,
DOI: 10.1007/978-94-017-8798-7_6,

6.1 Introduction

A cloud computing environment handles a large-scale task processing. And the system performance depends upon the resource states and task scheduling [1]. However, the task scheduling issue is well-known as NP-complete problem [2]. The task scheduling issue still has many challenges for researchers, and many studies bring out various solutions to overcome the task scheduling problem [3]. Typically, the scheduling policy applies a partitioning approach for the large-scale task. The cloud computing also divides large tasks into processing units according as the composition of each resource. However, the system efficiency has an influence upon the performance and distance of resources. And those factors also affect on a processing time.

In order to allocate the cloud resource for processing, existing research considers a task size and priority. However, the resource state has less concerned for scheduling. Even if some resource is finished the task, the available resource should wait until the completion of other busy resources. Thus, more study is needed to utilize those available resources. If the system provides a resource reallocation policy for available resources, It may improve a task processing speed and a utilization of the resources. The resource reallocation is the method to allocate the other large-scale task to the available resource. The method considers a network state on the virtualization environments [4]. Hence, the simulation-based verification is needed to adopt on the cloud computing system.

This paper proposes a fuzzy-based resource reallocation scheduling model (FRRSM). The proposed FRRSM includes the reallocation method for early finished available resource. The FRRSM relocates the available resource to other virtual machines and assigns a new large-scale task. In this process, the FRRSM utilizes a fuzzy method. The fuzzy logic for FRRSM handles a queue size of resources, a job size and a network distance as fuzzy input variables. The fuzzy logic for FRRSM handles a queue size of resources, a job size and a network distance as fuzzy input variables. And the logic finds the resource for reallocation with rule-based selection. To prove the performance of FRRSM, we design the virtual cloud computing environment with the DEVS-based modeling and simulation methodology [5].

6.2 Related Works

Some related study is underway to detect and solve the resource hotspot of the VM on the virtualization environments. In order to deploy the VM, Sandpipier [6] utilizes the state of the CPU, RAM and network. These states named 'volume' are the basis of the migration and replacement of the VM. However, this volume has focused on the simple viewpoint such as CPU-side states. It does not fit to utilize through the geographically distributed system likes cloud environments. Real time resource reallocation for better resource utilization (RTRRBRU) [7] calculates the usage of the CPU and memory in real time. RTRRBRU can measure the workload and real time reallocates the VM. This approach may reduce the workload for several resources. However, the reallocation in real time increases the network traffic. Thus RTRRBRU also does not fit to the cloud environments. And the cloud computing includes the dynamic resource allocation policy with the global scheduling method. However, the global scheduling only consider the resource allocation. Therefore, the cloud environments require the dynamic resource reallocation to manage the job finished resource.

6.3 Method

6.3.1 Dynamic Resource Reallocation in Cloud Environment

In the cloud environment, a virtual machine holds several physical machines. And each virtual machine processes its assigned large-scale task. The task managing of the virtual machine basically stores the user's request on the task queue. The manager constructs sub tasks from the task in the queue. The sub task is the MapReduce-based splits of the large-scale task. The manager sends sub tasks to the physical machine, which composed of virtual machines. And the physical machine processes the received sub task. After the processing is complete, the system sends the sub tasks' processing information to the virtual machine. And the sub task managing module merges the gathered information. A dynamic resource reallocation is the method to transfer an available resource between virtual machines. It has no trouble in processing the task on virtual machines. As the heavy-loaded resource may grants more available resources, it may improve the processing speed and time for a plural large-scale task.

6.3.2 Resource Reallocation Algorithm

Figure 6.1 shows the architecture of the FRRSM. FRRSM is composed of five components.

Job Generation listens the user's request for large-scale tasks, and creates the task for processing. Coordinator assigns the generated task to each VM. The VM has one or more physical machines. Resource Detector traces available resources (physical machine). Resource Detector traces available resources (physical machine). When Resource Detector finds some available resource, Monitoring obtains state information of physical machines. And then, Resource Reallocation assigns available resources to other VM using the fuzzy methods. Due to the cloud environment is the geographically distributed, most important factor is the performance of physical resources. And the distance between the physical machine

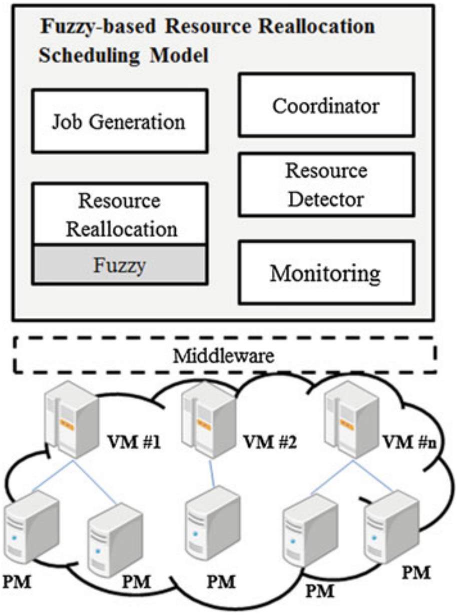

Fig. 6.1 FRRSM architecture

and virtual machine produces some effect on the processing efficiency. Hence, the resource manager should consider the distance and the job finishing time.

This paper utilizes the fuzzy method for resource reallocation. The fuzzy method calculates the queue size, job size and distance of each VM, and estimates the performance of the physical machine. The fuzzy parameter is composed of the queue size (μA), number of job (μB) and distance (μC). The fuzzy logic utilizes these parameters, and infers the Avail_Score (μD) to find a VM for resource reallocation. Each parameter (VL = VeryLow, L = Low, M = Middle, H = High, VH = VeryHigh) is presented as following Fig. 6.2.

Both μA and μB are state information of the queue and job size from the VM. And μC is the calculated distance between each VM. The distance may estimate with the Cartesian coordinate system and the Euclidean distance [6]. In order to infer the output μD, the logic utilizes three fuzzy input variables with 45 rule bases. Table 6.1 shows the 45 rule bases. The fuzzy inference engine is the Mamdani method, and selects the highest scored (μD) VM for resource reallocation.

6.4 Experiment

In order to prove the performance of FRRSM, we design the experiment environment with DEVS methodology. We select the Round-Robin scheduling model (RRSM) for comparison. And the experiment measures an average of job processing time using the RRSM with FRRSM and non-FRRSM.

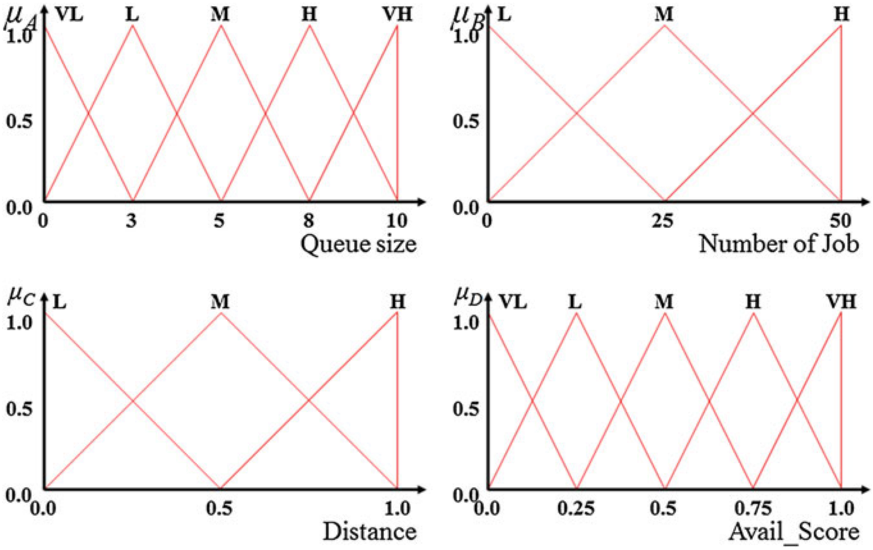

Fig. 6.2 Fuzzy variable

Table 6.1 Rule bases

μ_A, μ_B, μ_C		VL	L	M	H	VH
L	L	VH	VH	H	M	L
	M	VH	H	M	L	VL
	H	H	H	M	L	VL
M	L	VH	H	H	M	L
	M	VH	H	M	L	VL
	H	H	M	M	L	VL
H	L	VH	H	M	L	VL
	M	H	M	L	L	VL
	H	M	M	L	VL	VL

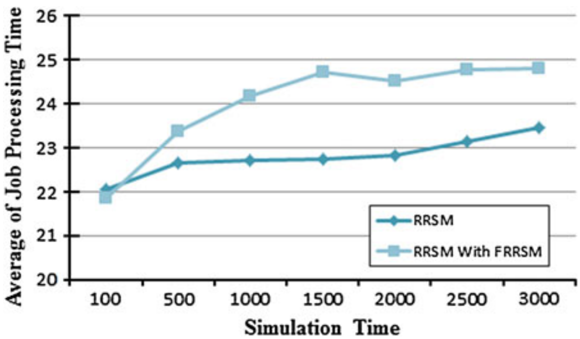

Fig. 6.3 Experiment result

As shown in Fig. 6.3, RRSM records 23.469, and RRSM with FRRSM records 24.814 average of job processing time for 3,000 simulation times. The above result shows that our proposed FRRSM is the effective method for large-scale task processing.

6.5 Conclusions

In this paper, we propose the fuzzy-based resource reallocation model to improve the efficiency of large-scale task processing on the cloud computing. The proposed method reallocates the available resources to other VM. To reallocate the resources, we utilize the fuzzy method. And we prove the effectiveness of our proposed method with the experimental result.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A2002751) and this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2013R1A1A3A04007527).

References

1. Calheiros R, Ranjan R, Buyya R (2011) Virtual machine provisioning based on analytical performance and QoS in cloud computing environments. In: International conference on parallel processing (ICPP), Taipei, Taiwan, pp 295–304
2. Bokhari SH (1987) Assignment problems in parallel and distributed computing. Kluwer Academic Publisher, Berlin
3. Munir EU, Li J, Shi S, Zou Z, Yang D (2008) MaxStd: a task scheduling heuristic for heterogeneous computing environment. Inf Technol 7:679–683
4. Weissman JB, Lee BD (2002) The virtual service grid: an architecture for delivering high-end network services. Concurr Pract Exp 14(4):287–319
5. Zeigler BP et al (1996) DEVS framework for modeling, simulation, analysis and design of hybrid systems in hybrid II. Lecture notes in CS, Springer-Verlag, Berlin, pp 529–551
6. Wood T, Shenoy P, Venkataramani A, Yousif M (2007) Black-box and gray-box strategies for virtual machine migration. In: Proceedings of the 4th USENIX symposium on networked systems design and implementation, pp 229–242
7. Sijin H, Li P, Yike G (2011) Real time elastic cloud management for limited resource. In: Cloud 2011 IEEE international conference on computing (CLOUD), Washington, pp 622–629
8. Tan PN, Steinbach M, Kumar V (2007) Introducton to Data Mining, Addison Wesley, Boston, pp 66–69