

# A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic

Amin Mehranzadeh  
Department of computer Science, Islamic Azad  
University of Dezfoul Branch,  
Dezfoul, IRAN.

Seyyed Mohsen Hashemi  
Dean of the Software Engineering and Artificial  
Intelligence Department, Science and Research  
Branch, Islamic Azad University, Tehran, IRAN.

## ABSTRACT

Cloud computing is a new technology and Due to its economical and operational benefits it is the most popular concept in IT today. Clouds consist of a collection of virtualized resources that can be provisioned on demand, depending on the clients' needs. In this technology almost everything like hardware, software and platform are provided as a service. In cloud computing one Host is a component that represents a physical computing and a Datacenter is composed by a set of hosts, which are responsible for managing Virtual Machines during their life cycles. An important issue in cloud computing is the scheduling of virtual machines requests, so that the requested tasks can be completed in a minimum time according to the user defined time. This paper presents and evaluate a new scheduling algorithm that is an efficient technique for scheduling virtual machines between datacenters. The simulation results show the effectiveness of our algorithm by comparing it with the two scheduling techniques First Fit (FCFS) and Round Robin (RR). The results depict that the proposed scheduling algorithm can be effectively deployed on Clouds.

## General Terms

Soft Computing, Distributed Systems, Algorithms.

## Keywords

Cloud computing, scheduling algorithm, fuzzy logic.

## 1. INTRODUCTION

Cloud computing is an extension of parallel computing, distributed computing and grid computing. Virtualization, distribution and dynamic extendibility are the basic characteristics of cloud computing [1]. Cloud computing services are divided into three classes, according to the abstraction level of the capability provided and the service model of providers, namely:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

Fig1 shows the layered architecture of the cloud where services of a higher layer can be composed from services of the underlying layer. Infrastructure as a Service (IaaS) Offering virtualized resources (computation, storage, and communication) on demand [2]. Infrastructure services are considered the bottom layer of cloud computing systems.

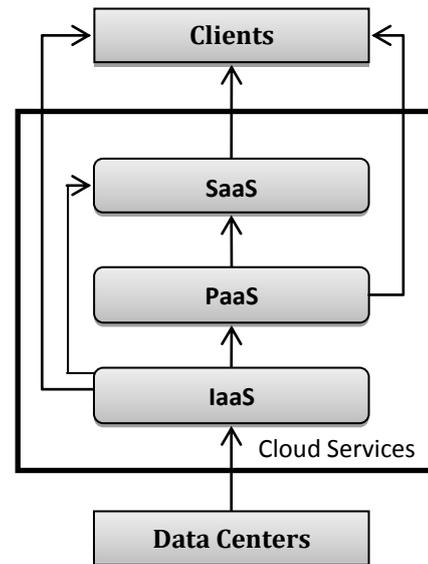


Fig 1: Layered architecture of the cloud

A cloud infrastructure enables on-demand provisioning of servers running several choices of operating systems and a customized software stack. In addition to infrastructure-oriented clouds that provide raw computing and storage services, Platform as a Service (PaaS) offer a higher level of abstraction to make a cloud easily programmable. Cloud platform offers an environment on which developers create and deploy applications and do not necessarily need to know how many processors or how much memory that applications will be using. In addition, multiple programming models and specialized services (e.g., data access, authentication, and payments) are offered as building blocks to new applications [3]. End users through Web can access services provided by the top of the cloud stack layer. Therefore, consumers are increasingly shifting from locally installed computer programs to on-line software services that offer the same functionality. This model of delivering applications, known as Software as a Service (SaaS), alleviates the burden of software maintenance for customers and simplifies development and testing for providers [4, 5].

## 2. RELATED WORK

The scheduling algorithms in distributed systems usually have the goals of spreading the load on processors and maximizing their utilization while minimizing the total task execution time [6]. Task scheduling, one of the most famous combinatorial optimization problems, plays a key role to improve flexible and reliable systems. The main Purpose is to schedule tasks to the adaptable resources in accordance with adaptable time, which involves finding out a proper sequence in which tasks can be executed under transaction logic constraints [7]. Based on the system information used by the scheduling approaches, there are two main categories, namely static and dynamic. Both have their own limitations. In [8], an optimized algorithm based on the Fuzzy-GA optimization was proposed which makes a scheduling decision by evaluating the entire group of task in the job queue. In [9], a new Bee Swarm optimization algorithm called Bees Life Algorithm (BLA) was proposed which applied to efficiently schedule computation jobs among processing resources onto the cloud datacenters. It is considered as NP-Complete problem and it aims at spreading the workloads among the processing resources in an optimal fashion to reduce the total execution time of jobs and then, to improve the effectiveness of the whole cloud computing services. BLA has been inspired by bees' life in nature represented in their most important behaviors, which are reproduction and food source searching. This paper presents and evaluates a new scheduling algorithm that is an efficient technique for scheduling virtual machines between datacenters. The simulation results show the effectiveness of our algorithm by comparing it with the two scheduling techniques First Fit (FCFS) and Round Robin (RR).

## 3. MATERIALS AND METHODS

### 3.1 Fuzzy logic control system

The proposed fuzzy controller uses fuzzy logic, introduced by Zadeh in 1965 [10]. Fuzzy logic has no strict assignment of elements to sets like binary logic. Instead, every element has a degree of membership to a set. This degree is represented by a value between 0 and 1. To be able to apply fuzzy logic to a specific problem such as the scheduling between cloudlets or virtual machines, a fuzzy system must be constructed.

The construction consists of three steps:

- 1) **Fuzzyfication:** In this step, the degree of membership of the input values is assigned to fuzzy sets. The degree of membership is given by  $\mu: X \rightarrow [0, 1]$ , where X is the set of input values. So every input value is mapped to a value between zero and one.
- 2) **Fuzzyfication:** In this step, the degree of membership of the input values is assigned to fuzzy sets. The degree of membership is given by  $\mu: X \rightarrow [0, 1]$ , where X is the set of input values. So every input value is mapped to a value between zero and one.
- 3) **De-Fuzzyfication:** In this step, a numerical output value is generated from the output set.

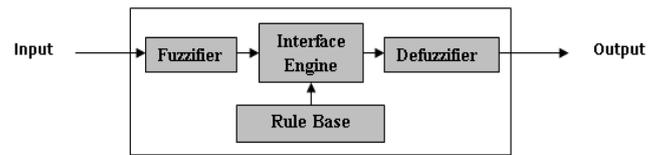


Fig 2: Fuzzy control system

The difference between these logics is that fuzzy set theory provides a form to represent uncertainties; that is, it accepts conditions partially true or partially false. Fuzzy logic is a good logic to treat random uncertainty, i.e., when the prediction of a sequence of events is not possible. A fuzzy control system [11] is a rule-based system, which a set of so-called fuzzy rules represents a control decision mechanism to adjust the effects of certain causes coming from the system. The aim of the fuzzy control system is normally to substitute for or replace a skilled human operator with a fuzzy rule based system. Specifically, based on the current state of a network an inference engine equipped with a fuzzy rule base determines an online decision to adjust the system behavior in order to guarantee that it is optimal in some certain senses. There are generally two kinds of fuzzy logic controllers. One is the feedback controller, which is used in this paper, is shown in Fig.2. The output of the fuzzy logic controller in Fig.2 is used to tune the controlled system's parameters based on the state of the system. The design process of a fuzzy control system consists of a series steps. The first step in fuzzy control is to define the input variables and the control variables. Each variable must be quantified. Then each quantification of the variable is assigned a membership function. Then a fuzzy rule base must be design, this rule base determines what control action take place under what input conditions. The rules are written in an if-then format. An implication formula is used to evaluate the individual if-then rules in the rule base [11]. A composition rule is used to aggregate the rule results to yield a fuzzy output set. In the proposed fuzzy system, Mamdani minimum inference method [12] was used as the fuzzy inference method.

### 3.2 Proposed Scheduling Algorithm

This paper proposes a dynamic scheduling algorithm, utilizing the fuzzy logic controller. The controller takes advantage of two inputs, the number of requests that received in host from virtual machines and the value of tasks precedence that run in each virtual machine. The number of request as AGE and the value of tasks precedence named as PREC for simply. The output of the fuzzy controller is the virtual machine id that must assigned to the host. Using this scheduling scheme, the virtual machines fairly assigned to the host according to the time slot of waiting in the queue and their precedence. In our simulation, each task has the precedence value from 0 to 3 and the AGE value is between 0 to 3 for four virtual machines. Design of a fuzzy control routing system consists of series steps:

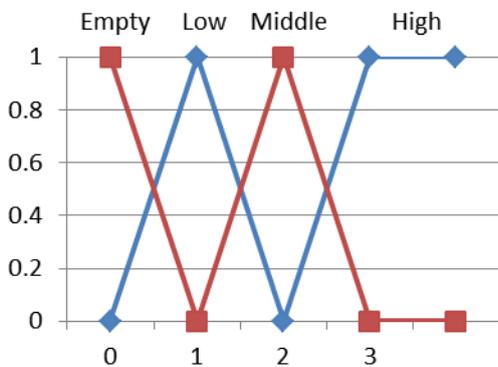
**Step 1:** Define the input and output variables, respectively the number of requests that received in host from virtual machines and the value of tasks precedence that run on each virtual machine are as inputs and the virtual machine id that must assigned to the host is as the output.

**Step2:** Each variable quantified, for instance the AGE because is requests that sent to the host and set to 0, 1, 2 and 3 (four virtual machines), it quantified as Zero, Small, Medium and Big. Each quantification of the variable is assigned a membership function as shown in figure 3 and 4. Membership function for cost is chosen to be singletons.

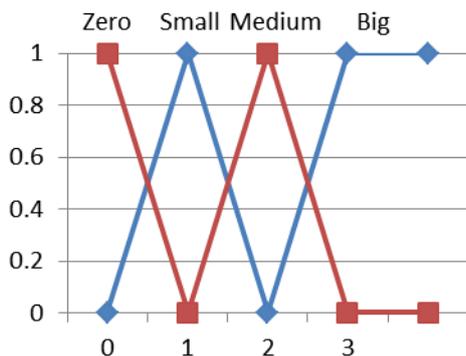
**Step3:** A fuzzy rule base designed. These rule bases determine what action take place under what input conditions. The rules are written in if-then formats as shown in table 1.

**Step4:** a defuzzification method is applied to the fuzzy control action to produce a crisp cost. We use one of the famous defuzzification method "centers of gravity" to produce a crisp output.

**Step5:** The virtual machine id with the low cost of AGE and PREC values is selected for output to access to the host.



**Fig 3: Membership function for the tasks precedence which run on each virtual machines (PREC)**



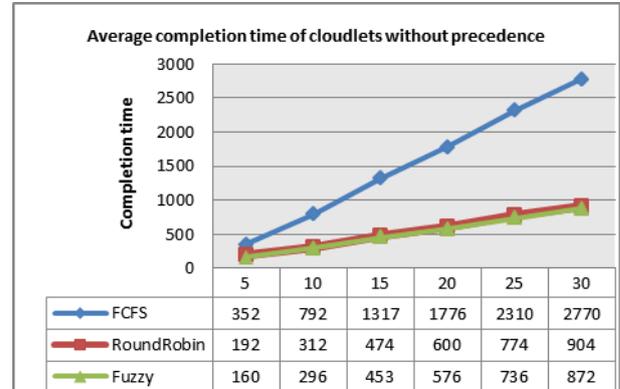
**Fig 4: Membership function for number of the virtual machine requests that sent to the host (AGE)**

**Table 1. Rule Base for fuzzy based approach for scheduling algorithm**

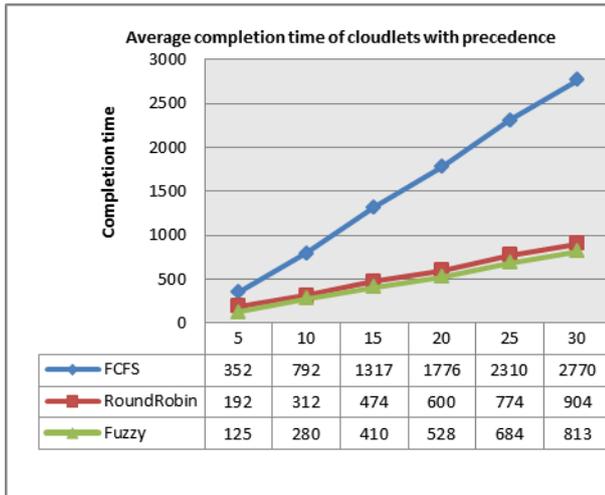
PREC \ AGE	Zero	Small	Medium	Big
Empty	Low	Low	Low	Very High
Low	Low	Medium	Medium	Very High
Middle	Medium	Medium	High	Very High
High	Medium	High	Very High	Very High

#### 4. EXPERIMENTAL RESULTS

The CloudSim simulator used in order to evaluate the fuzzy scheduling algorithm. In CloudSim, the tasks are to add the scheduling algorithm. The submitCloudlets () method in DatacenterBroker class of CloudSim can bind the single task to a designated single virtual machine and run it [13]. Compare with traditional algorithms (FCFS and Round Robin), our proposed algorithm that using fuzzy logic is efficient. Fig.5 (Comparing Completion time) shows that proposed algorithm reduces the Completion time when compare with traditional approaches. Fig.5 is used for calculating the Completion time. Number of cloudlets (5 to 30) is simply represents the number of jobs/tasks submitted to the system. In Fig.5 Completion time is calculate without any task precedence.



**Fig 5: Calculation of completion time without precedence**



**Fig 6: Calculation of Completion Time with precedence**

Fig6, shows that proposed algorithm reduces the Completion time when compare with traditional approaches under task precedence. In our simulation, value of tasks precedence set between 1 to 5 that shows the bigger values have bigger precedence. In Fig6, our algorithm compare with traditional approaches reduces Completion time.

## 5. CONCLUSION

This paper presented a new approach for scheduling algorithm on Cloud, namely Fuzzy scheduling that using a fuzzy system. Through CloudSim simulation tool, proposed algorithm implemented and it shows better performance than the traditional algorithm. We investigated the effect of cloudlets number, and tasks precedence on the completion time of jobs/tasks. Simulation results demonstrated the advantage of proposed scheduling algorithm in terms of average completion time compared to FCFS and Round Robin algorithms.

## 6. REFERENCES

[1] R.Buyya, J.Broberg, A.Goscinski, "Cloud Computing Principles and Paradigms", 2011.  
 [2] B. Sotomayor, R. S. Montero, I. M. Lorente, and I. Foster, Virtual infrastructure management in private and

hybrid clouds, IEEE Internet Computing, 13(5):14-22, September/October, 2009.

[3] Appistry Inc., Cloud Platforms vs. Cloud Infrastructure, White Paper, 2009.  
 [4] L. Youseff, M. Butrico, and D. Da Silva, Toward a unified ontology of cloud computing, in Proceedings of the 2008 Grid Computing Environments Workshop, pp. 1-10, 2008.  
 [5] B. Hayes, Cloud computing, Communications of the ACM, 51:9-11, 2008.  
 [6] A.Y.Zomaya, and Y.The, "Observations on using genetic algorithms for dynamic load-balancing", IEEE Transaction on Parallel and Distributed Systems, vol. 12, no. 9, pp. 899-911, 2001.  
 [7] XIAO Zhi-Jiao, CHANG Hui-You, YI Yang, "An Optimization Method of Workflow Dynamic Scheduling Based on Heuristic GA", Computer Science, Vo1.34 No.2, 2007.  
 [8] S.Tayal, "Tasks scheduling optimization for the cloud computing systems", International journal of advanced engineering sciences and technologies, Vol No.5, Issue No.2,111-115, 2011.  
 [9] S.Bitam, "Bees Life Algorithm for job scheduling in cloud computing", ICCIT 2012, 186-191, 2012.  
 [10] L.Zadeh, "Fuzzy Sets", Information and Control, vol. 8, pp. 338–353, 1965.  
 [11] D.Driankov, H.Hellendoom and M.Reinfrank, An introduction to fuzzy control, Springer-Verlag, Berlin, New York, 1993.  
 [12] Hao Ying, "Fuzzy Control and modeling-analytical foundation and applications", IEEE press, 2000.

R.Buyya, R.Ranjan and R.N.Calheiros, "Modeling and simulation of scalable Cloud Computing environments and the CloudSim Toolkit: challenges and opportunities", Proceedings of the seventh high performance Computing and simulation conference (HPCS 2009, ISBN: 978-1-4244-49071), Leipzig, Germany. New York, USA: IEEE Press, June 21–24, 2009.