

## FUZZY LOGIC-BASED ALGORITHM RESOURCE SCHEDULING FOR IMPROVING THE RELIABILITY OF CLOUD COMPUTING

MOHAMMAD ZAVVAR  
MEYSAM REZAEI  
SHOLE GARAVAND  
FARHAD RAMEZANI

### ABSTRACT

Cloud computing is an important infrastructure for distributed systems with the main objective of reducing the use of resources. In a cloud environment, users may face thousands of resources to run each task. However, allocation of resources to tasks by the user is an impossible endeavor. Accurate scheduling of system resources results in their optimal use as well as an increase in the reliability of cloud computing. This study designed a system based on fuzzy logic and followed by an introduction of an efficient and precise algorithm for scheduling resources for improving the reliability of cloud computing. Waiting and turnaround times of the proposed method were compared to those of previous works. In the proposed method, the waiting time is equal to 26.99 and the turnaround time is equal to 82.99. According to the results, the proposed method outperforms other methods in terms of waiting time and turnaround time as well as accuracy.

Keywords: cloud computing, fuzzy logic, scheduling, reliability, waiting time, turnaround time

### INTRODUCTION

Cloud computing is a model for enabling ubiquitous, convenient, and demand-driven network access to a set of shared and configurable computing resources such as networks, servers, storage, and application services among others. These can be provided with sufficient speed and minimal effort. Furthermore, little contact with server managers is required. The common approach in cloud computing services is to provide a source for an agreed amount of time for a certain demand. Known cloud servers take advantage of this method in return for hourly use of the CPU. Despite being one of the most basic methods of selling goods or services, this prevents the complex interaction between cloud buyer and provider.

The cloud computing environment is composed of different types of resources which are assigned based on user demand for the works they perform and provide service to different users (Garg et al., 2013). The scheduling of the resources in the cloud computing environment is an important issue. Scheduling is the proper allocation of resources to works (Ye, 2015). In a cloud environment, each user may have to choose from thousands of resources to run each work. However, it is not possible for the user to manually allocate resources to jobs. The purpose of cloud computing systems is to minimize the cost of resources and to optimize provision of services to consumer applications (Zissis & Lekkas, 2012). To this end, the scheduling system is responsible for different tasks in the cloud to increase completion rates, increase resource utilization, and enhance computing power. Therefore, the scheduling problem is an important issue in managing resources in the cloud, because the large number of requests preclude manual scheduling in the data center. With respect to scheduling, a number of input parameters such as the cost of implementation, timeline, and efficiency need to be considered. A major goal of cloud computing is to reduce costs associated with the use of resources, which

are available virtual machines in a cloud computing system (Fernando, Loke & Rahayu, 2013). Therefore, the scheduling algorithms play an important role in cloud computing, because the goal of scheduling is to reduce response times and improve resource utilization.

Reliability is one of the main criteria in making decisions and planning engineering and maintenance tasks. The most common definition of reliability is "possibility of normal operation without failure for a specified length of time, according to pre-existing conditions" (Zheng & Lyu, 2010). Reliability is the ability to ensure the stability of the system. A high-performance system can continue to function without any interruption over a relatively long period of time. Today, top features and high reliability are necessary for any product. Therefore, the guarantee of reliability is one of the important characteristics of products which requires improved reliability as a fundamental principle. Software reliability is also a critical factor which determines product quality. As a result of the wide use of cloud environment, improving reliability results in reduced physical and financial risks (Chiu, Huang & Lee, 2008). Given the importance of resource allocation and improvement of the reliability of cloud computing. In this paper, an algorithm is proposed based on fuzzy system for scheduling resources and improvement of the reliability in the cloud environment.

## BACKGROUND

In this section, a number of resource scheduling algorithms are examined and advantages and disadvantages of each are discussed. FIFO is an exploratory type of scheduling which was initially used in the first heuristic. In this algorithm, a new job exists the queue based on the time of its entry. Jobs are selected and processed in the order they arrive. The scheduling algorithm has minimal overhead and is easy to implement however, takes longer to execute on machines with lower throughput (Pan & Yang, 2005).

Min-Min is a scheduling algorithm wherein a task with the shortest expected time to completion is assigned to a processor. This method is the same as MTC based on the minimum completion time (Chen & Huang, 1998). However, unlike MTC which only considers one task at a time, Min-Min examines all unassigned tasks at every turn. Min-Max algorithm is the same as the way using Min-Min. Min-Max algorithm is a set of all unscheduled tasks for each task in the set and then calculates the minimum completion time. The difference between Min-Max and Min-Min lies in the fact that the tasks are completed and mapped with maximum time to the machine. Then, the scheduled task is removed from the collection. The process continues until all tasks are scheduled (Bhoi & Ramanuj, 2013).

A hybrid scheduling algorithm known as RASA was proposed Parsa and Entezari-Maleki (2009) which combines the two previous methods. The algorithm uses the advantages of the methods while eliminating their shortcomings. According to experimental results, the algorithm exhibits superior performance compared to large-scale scheduling algorithms in distributed systems. Another scheduling algorithm known as SHEFT was proposed by Wu et al., (2013). Its objective is to increase scheduling flexibility in the cloud computing environment. Experimental results show that the algorithm achieves better performance in terms of number of scheduled tasks, workflow runtime optimization, flexibility, and scalability. A scheduling algorithm for resource allocation is presented in Bittencourt, Senna and Madeira (2010) which is based on a novel architecture. The algorithm considers two types of prioritization i.e. the prioritization of resources and prioritization of tasks. Prioritization of resources are determined by the numbers of CPU cores rather than the price of those resources. To prioritize the requests received by the user, first, resources requiring more time are selected. If the necessary resources are available in the private cloud, they can be used at no cost; otherwise, the public cloud is employed so that deadlines can be met. Using the public cloud requires payment.

In Ming, Mengkun & Fuqin (2010), an algorithm and a framework for scheduling processing are provided. The algorithm manages requests and allocation of available resources in a hierarchical manner. In the SaaS architecture, an area known as the buffer is devised which is responsible for breaking user requests down to numerous stages. Authors within cloud architecture on services SaaS presented an area as buffer- pool that its work is the division of users on various stages. In each stage, different methods are created and sent to resources in the cloud in order to be processed.

A method to allocate requests to resources is provided with the aim to minimize total run time (Paul & Sanyal 2011) where a cost matrix is created on credits used for each task which has been assigned to a source. Each task is more likely to have a higher degree of credibility in allocating the best resources available. In this way architecture is provided storing responsibility for user submissions in a buffer of central middleware. As a result, the tasks are hierarchically divided among local middleware. Subsequent to this classification, scheduling is performed by the software.

A scheduling algorithm for the cloud environment was devised by Selvarani and Sadhasivam (2010) which considers tasks in groups and assigns different costs and mathematical functions to resources. The classification operation allocates computing resources and optimized communication to larger tasks. The grouping algorithm works by bundling independent tasks requiring small processing and larger tasks requiring greater processing.

## CLOUD COMPUTING

Cloud computing offers infrastructure, platform and software as services, which can be accessed by subscription. Consumers are charged to the extent that they use such services. The consumers are charged according to pay-as-you-go model. The three possible types of services are known as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS)(Krishna, 2013).

Cloud computing means differently to different people making the concept accorded with various definitions. Calheiros et al. (2011), define cloud computing based on concepts such as scalability, virtualization, universal concepts and pay to use. Cloud computing is a style of computing services wherein the service is provided over the internet using models and various layers of abstraction (Li et al., 2009).

Armbrust et al. (2010) assert that cloud computing refers to both applications as services provided over the internet and software and system software in data centers that provide these services. This definition relates to the novel nature of cloud computing because it implies that both the software applications and hardware infrastructure have been transformed from the private media to third-party data centers and are available via the Internet. Buyya et al. (2009) define cloud computing as a type of parallel and distributed systems comprising a series of interconnected computers and virtual computing resources which are provided as integrated, dynamic forms based on service level agreements. This definition is based on a market-oriented approach, underlining the economic nature of cloud computing.

The main feature of the cloud computing definition expressed by Armbrust and Buyya in Armbrust et al. (2010) and Buyya et al. (2009) refer to the capability of delivering infrastructure and software as services to consumers based on pay-per-use. Cloud computing strategy is global cloud and covers all kinds of calculations. Cloud computing is based on a service level agreement, which includes requirements for quality of service. Essentially, the cloud sits between the consumer and provider. The contract determines the details of the service that should be provided in the form of metrics agreed upon by both sides. The contract also defines penalties for breaching. Service level agreement serves to guarantee users that the cloud functions properly. Clouds aim to provide features that enable users to achieve their goals

anywhere in the world on demand. Moreover, relevant services with competitive prices and quality requirements need to be supported. Cloud computing provides its services for all users, ranging from those who wish to keep their documents on the Internet to the companies which outsource their entire IT infrastructure to external data centers. As a result, companies can reduce management costs and maintenance of IT infrastructure with by renting IT infrastructure from cloud providers.

## FUZZY LOGIC

The concept of fuzzy logic was first developed in 1965 (Zadeh, 1965). This theory is a new form of mathematics for formulating and analyzing concepts and features sets. In fact, the fuzzy set theory is a generalization and a natural extension of the agreed common language and understanding of human nature (Mendel, 1995).

Definition 1: Let  $X$  be an arbitrary reference set, characteristic function of each normal sub-set of  $X$  and  $A$ ,  $\mu_A : X \rightarrow \{0, 1\}$  is defined as follows:

$$\mu_A(x) = \begin{cases} 1 & : x \in A \\ 0 & : x \notin A \end{cases} \quad (1)$$

According to the above definition, for each  $x \in X$ ,  $\mu_A(x)$ , it will take only one of the values 0 and 1.

Definition 2. If the function board  $\mu_A$  from the set of two members  $\{0, 1\}$  to intervals  $[0, 1]$  is developed, a function is obtained for each member of  $X$  which attributes a number in the range  $[0, 1]$ . Thus,  $A$  is known as a fuzzy set.

In the above definition, assuming  $\mu_A(x) \in (0, 1)$ , then membership to  $A$  becomes uncertain.

Also, the membership  $\mu_{\tilde{A}}(x)$  represents the membership element  $x$  to the fuzzy set  $\tilde{A}$ . If the membership of an element of the set equals to 0, the member is removed from the set and if the degree of membership of one member equals 1, the member belongs to a set (Sugeno & Yasukawa, 1993). If the membership of a member falls between 0 and 1, the number is indicative of the gradual membership (Lee, 1990).

## THE PROPOSED ALGORITHM

As mentioned earlier, the cloud is a set of computing resources configured as networks, servers, storage space, applications and services. Cloud computing has different service layers (Jadeja & Modi, 2012) as shown in Figure 1.

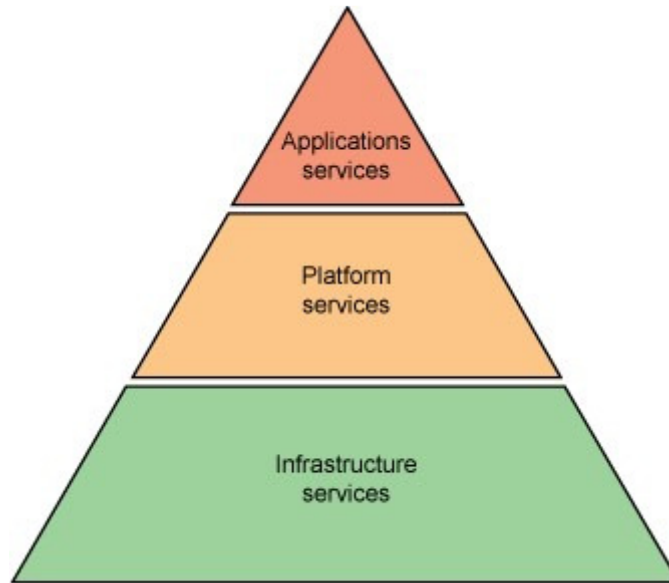


FIGURE 1. Layers in the cloud computing environment

Cloud software service is placed as SaaS on the Internet. The service layer allows the consumer of the cloud platform to monitor and apply various settings on the environment according to the program. In the service layer of the cloud infrastructure, the consumer can use base computing resources such as processing power, storage and network components (Dinh et al., 2013; Modi et al., 2013). In the cloud computing environment, resources are managed in the infrastructure layer; furthermore, the purpose of this paper is to propose an algorithm for scheduling resources in a cloud computing environment; therefore, the proposed algorithm is to be associated with the infrastructure layer. Scheduling pattern should be atomic in the scheduling of processes; in other words, either it should be run completely or should not be run and withdrawn.

In this paper, the authors intend to discuss the different metrics of reliability for effective cloud environment. These are the fuzzy metrics of cost, trust and length to provide resource scheduling algorithm based on fuzzy logic to enhance the reliability of cloud computing. However, metrics must first be transferred to phase space prior to implementing the proposed system. Table 1 shows the fuzzy metric of cost, trust and length.

TABLE 1. Phase of the input parameters

Verbal Variable	Trust	Cost	Length
Low	$[-inf\ 0\ 50]$	$[-inf\ 0\ 50]$	$[-inf\ 0\ 50]$
Middle	$[0\ 50\ 100]$	$[0\ 50\ 100]$	$[0\ 50\ 100]$
High	$[50\ 100\ inf]$	$[50\ 100\ inf]$	$[50\ 100\ inf]$

In Table 2, the range of each of the variables verbal output is shown.

TABLE 2. Scope of each of the variables verbal output

Priority	Output
Very Low	$[-inf\ 0\ 25]$
Low	$[0\ 25\ 50]$
Middle	$[25\ 50\ 75]$
High	$[50\ 75\ 100]$
Very High	$[75\ 100\ inf]$

In this paper, the membership function trimf for the fuzzification of the inputs and outputs is used. Figure 2 depicts the input and output membership functions for the proposed model.

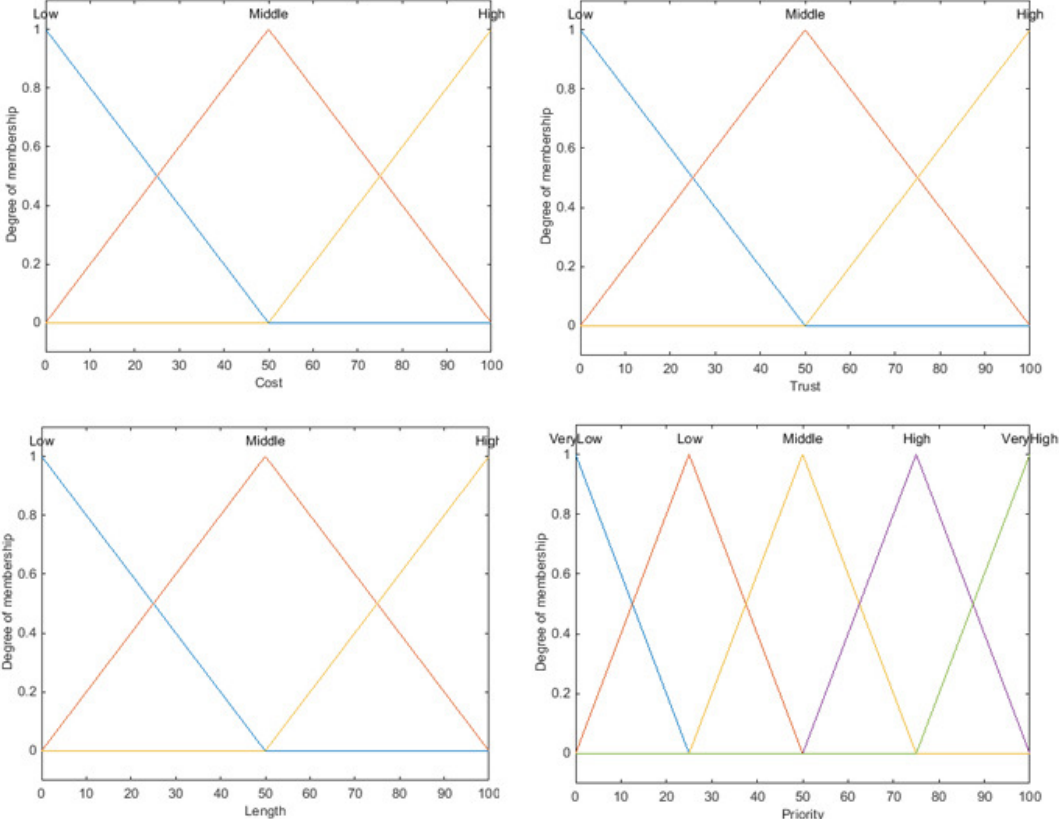


FIGURE 2. Membership functions of input and output variables

The overall system model is shown in Figure 3. The output of the system is to determine the priority based on costs, trust and length.

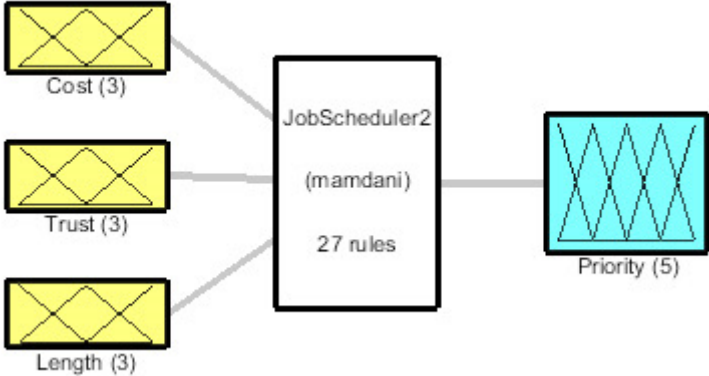


FIGURE 3. The overall system model

As shown in Figure 3, the proposed model has three inputs (i.e. cost, trust, and length) and one output (i.e. priority). Since each of the three input parameters include three membership functions, a total of 27 rules are used in this model. These rules are shown in Table 3. The proposed scheduling algorithm is performed to assign resources to tasks based on user needs according to the following steps.

TABLE 3. The proposed scheduling algorithm

```

// receive tasks
Algorithm ResourceScheduling (tasks[n])
begin
  for i = 0 to n do
    // fuzzification of cost, Length, trust with trimf(x,[a b c])
    // the triangular curve is a function of a vector, x, and depends on three scalar
    parameters a, b,
    and c, as given by  $trimf(x; a, b, c) = \max(\min(\frac{x-a}{b-a}, \frac{c-x}{c-b}), 0)$ .
    // the parameters a and c locate the "feet" of the triangle and the parameter b locates the
    peak.
    params = [ a b c ];
    y1 = trimf (cost, params); y2 = trimf (length, params); y3 = trimf (trust, params);
    // determine prioritize tasks based on fuzzy inference Rules
    // defuzzification with centroid.
    // centroid defuzzification returns the center of area under the curve
    results [i] = inference (y1, y2, y3);
  end
  sort (results); // scheduling tasks
  // allocate resources
end

```

## RESULTS AND DISCUSSIONS

This section evaluates the proposed algorithm and shows the results on separate diagrams. To evaluate the proposed algorithm, a dataset consisting of random numbers between zero and one with three inputs and one output is used. This dataset contains 21 records each of which is related to one of the tasks. After the delivery of this dataset to the proposed system, the waiting time and the time of the proposed algorithm are calculated; then mean waiting time and turnaround time of the proposed algorithm are compared to those of other algorithms. This is then followed by introducing a better performance algorithm. In calculating the waiting and turnaround time, the relations between (2) and (3) are used.

$$\text{Waiting time} = \text{Taking time} - \text{Time of arrival} \quad (2)$$

$$\text{Turnaround time} = \text{Exit time} - \text{Time of arrival} \quad (3)$$

Turnaround and waiting times for the proposed algorithm and other algorithms are shown in Figures 5 and 6 respectively.

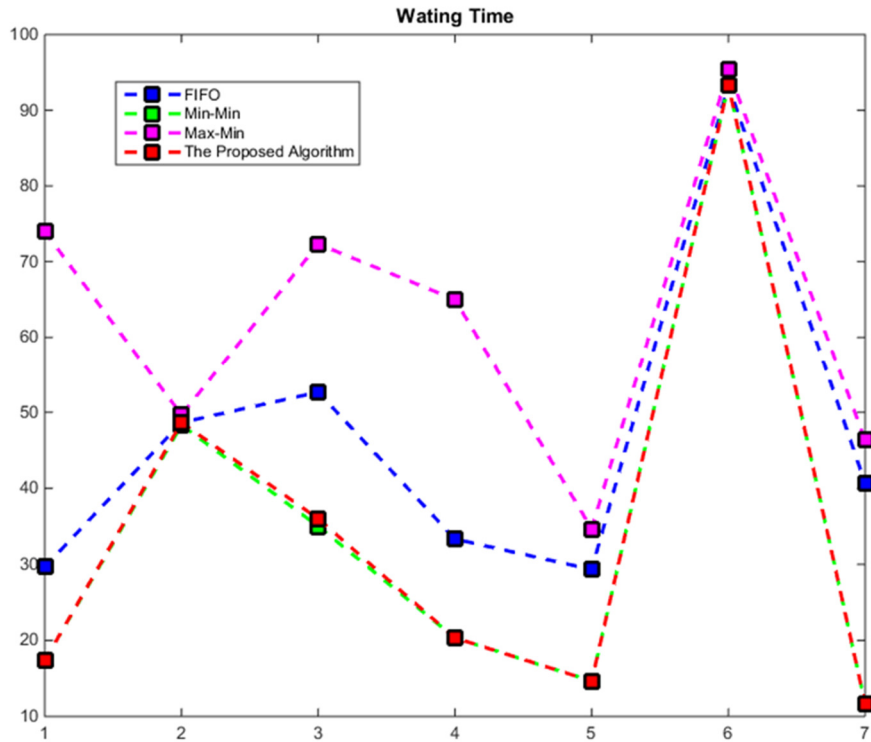


FIGURE 5. Waiting times of different algorithms

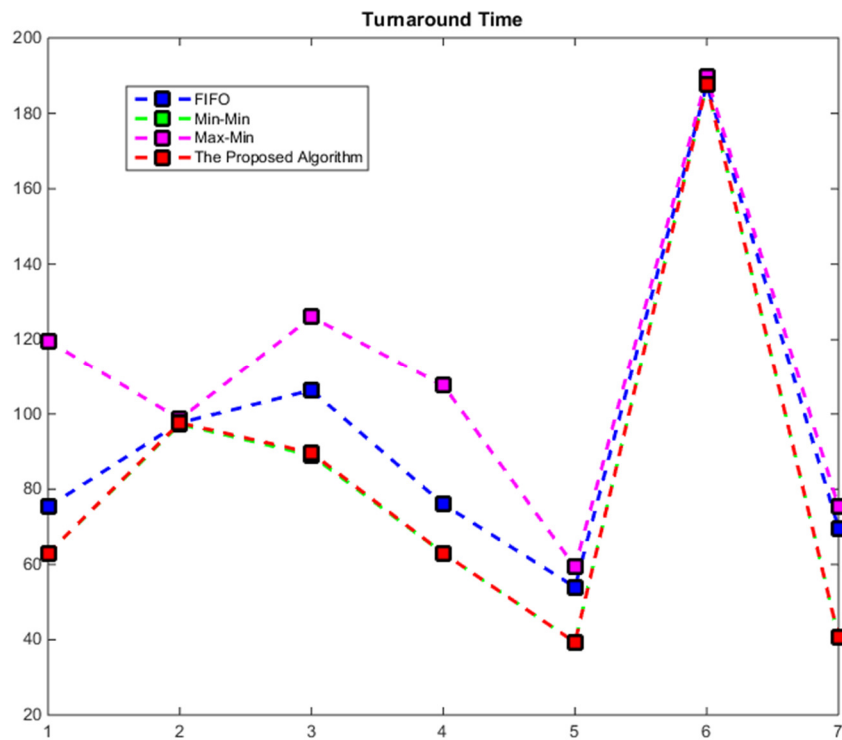


FIGURE 6. Turnaround times of different algorithms

As the figures indicate, the waiting period for the proposed algorithm equals 26.99, while for FIFO, Min-Min, and Max-Min the values equal 43.52, 26.85, and 66.66, respectively. The turnaround time for the proposed algorithm equals 82.99, for FIFO equals 95.23, for Min-Min equals 82.85, and for Max-Min equals 11.09. Based on these results, the waiting and turnaround times of the proposed algorithm are superior to the waiting and turnaround times of FIFO and



Max-Min. Furthermore, the proposed algorithm has a small difference with the Min-Min algorithm. Additionally, resource allocation with algorithms is prioritized based on length, while the proposed algorithm consider both cost and trust criteria in addition to length. Therefore, due to improving waiting and turnaround time in the proposed algorithm and the use of appropriate metrics, it can be concluded that the proposed algorithm has better performance and is more accurate than other algorithms.

## CONCLUSION

In this paper, an efficient algorithm based on fuzzy system for scheduling resources was proposed to improve the reliability of cloud computing. The output of the fuzzy system is the allocation of resource priority to each of the tasks according to three working time metrics. The output results in lower cost and increased reliability. In fuzzy system, a total of three linguistic variables and a type of membership function, namely trimf, were considered. Finally, the proposed algorithm was compared with other algorithms on the same random data sets. The results indicate that waiting and turnaround time of the proposed algorithm are better than those of FIFO and Max-Min while showing a small difference with the Min-Min algorithm. Thus, the improved waiting and turnaround time in the proposed algorithm and the use of appropriate metrics indicate that the proposed algorithm is superior in terms of both accuracy and performance.

## REFERENCES

- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A. & Stoica, I. 2010. A view of cloud computing. *Communications of the ACM*, 53(4):50-58.
- Bhoi, U. & Ramanuj, P. N. 2013. Enhanced Max-min task scheduling algorithm in cloud computing. *International Journal of Application or Innovation in Engineering and Management (IJAIEEM)*, 2(4):259-264.
- Bittencourt, L. F., Senna, C. R. & Madeira, E. R. 2010. Scheduling service workflows for cost optimization in hybrid clouds. *2010 International Conference on Network and Service Management*, Niagara Falls, 25-29 Oct 2010.
- Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J. & Brandic, I. 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5<sup>th</sup> utility. *Future Generation Computer Systems*, 25(6):599-616.
- Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. & Buyya, R. 2011. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1):23-50.
- Chen, R-M. & Huang, Y-M. 1998. Multiconstraint task scheduling in multi-processor system by neural network. *10<sup>th</sup> IEEE International Conference on Tools with Artificial Intelligence*, Taipei, 10-12 Nov 1998.
- Chiu, K-C., Huang, Y-S. & Lee, T-Z. 2008. A study of software reliability growth from the perspective of learning effects. *Reliability Engineering & System Safety*, 93(10):1410-1421.
- Dinh, H. T., Lee, C., Niyato, D. & Wang, P. 2013. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Communications and Mobile Computing*, 13(18):1587-1611.
- Fernando, N., Loke, S. W. & Rahayu, W. 2013. Mobile cloud computing: A survey. *Future Generation Computer Systems*, 29(1):84-106.
- Garg, S. K., Versteeg, S. & Buyya, R. 2013. A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4):1012-1023.
- Krishna, P. V. 2013. Honey bee behavior inspired load balancing of tasks in cloud computing environments. *Applied Soft Computing*, 13(5):2292-2303.

- Lee, C. C. 1990. Fuzzy logic in control systems: fuzzy logic controller. II. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):419-435.
- Li, B., Li, J., Huai, J., Wo, T., Li, Q. & Zhong, L. 2009. Enacloud: An energy-saving application live placement approach for cloud computing environments. *International Conference on Cloud Computing, Bangalore*, 21-25 Sept 2009.
- Mendel, J.M., 1995. Fuzzy logic systems for engineering: a tutorial. *Proceedings of the IEEE*, 83(3):345-377.
- Ming, C., Mengkun, L. & Fuqin, C. 2010. A model of scheduling optimizing for cloud computing resource services based on Buffer-pool Agent. *International Conference on Granular Computing (GrC, San Jose*, 14-16 Aug 2010.
- Modi, C., Patel, D., Borisaniya, B., Patel, A. & Rajarajan, M. 2013. A survey on security issues and solutions at different layers of Cloud computing. *The Journal of Supercomputing*, 63(2):561-592.
- Pan, D. & Yang, Y., 2005. FIFO-based multicast scheduling algorithm for virtual output queued packet switches. *Computers, IEEE Transactions on*, 54(10):1283-1297.
- Parsa, S. & Entezari-Maleki, R. 2009. RASA: A new task scheduling algorithm in grid environment. *World Applied Sciences Journal*, 7: 152-160.
- Paul, M. & Sanyal, G. 2011. Survey and analysis of optimal scheduling strategies in cloud environment. *World Congress on Information and Communication Technologies (WICT)*, Mumbai, 11-14 Dec 2011.
- Selvarani, S. & Sadhasivam, G. S. 2010. Improved cost-based algorithm for task scheduling in cloud computing. *International Conference on Computational Intelligence and Computing Research (ICCIC)*. Coimbatore, 28-29 Dec 2010.
- Sugeno, M. & Yasukawa, T. 1993. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1(1):7-31.
- Wu, Z., Liu, X., Ni, Z., Yuan, D. & Yang, Y. 2013. A market-oriented hierarchical scheduling strategy in cloud workflow systems. *The Journal of Supercomputing*, 63(1):256-293.
- Jadeja, Y. & Modi, K. 2012. Cloud Computing - Concepts, Architecture and Challenges. *International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, Kumaracoi, 21-22 March 2012.
- Ye, H. 2015. Optimization of Resource Scheduling Based on Genetic Algorithm in Cloud Computing Environment. *Metallurgical and Mining Industry*, 7(6):386-391.
- Zadeh, L.A., 1965. Fuzzy sets. *Information and control*, 8(3):338-353.
- Zheng, Z. & Lyu, M. R. 2010. Collaborative reliability prediction of service-oriented systems. *Proceedings of the 32<sup>nd</sup> ACM/IEEE International Conference on Software Engineering*, New York: ACM. 1 May 2010.
- Zissis, D. & Lekkas, D., 2012. Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3):583-592.

Mohammad Zavvar  
 Sama Technical and Vocational Training College,  
 Islamic Azad University,  
 Gorgan Branch,  
 Gorgan, Iran  
 Email: zavvar.developer@gmail.com

Meysam Rezaei, Shole Garavand, Farhad Ramezani  
 Department of Computer Engineering,  
 Sari Branch, Islamic Azad University,  
 Sari, Iran  
 mem\_re69@yahoo.com, shole\_garavnd@yahoo.com, ramezani.farhad@iausari.ac.ir

Received: 28 December 2015

Accepted: 23 May 2016