International Journal of Innovative Research in Computer Science & Technology (IJIRCST)
ISSN: 2347-5552, Volume-3, Issue-2, March 2015

A New Approach for Task Scheduling of Cloud Computing Using Fuzzy

Anindita Kundu

***Abstract*— Cloud Computing is the distributed computing on internet or delivery of computing services over the internet. Cloud computing involves deploying groups of remote servers and software networks that allow centralized data storage and online access to computer services or resources. Cloud Computing allows the customers to apply the application without set up and access their own files on any device with internet. The cloud service providers have developed the cloud enough to provide services to any number of users. Cloud is a set of virtualized resources that can be produced as demand of the user. In cloud computing basically there are three components: one is the host that represents a physical computing, second is the datacenter that is a collection of server for managing Virtual Machines during their life cycles. Here application is placed, which is accessed by internet. The last is the distributed servers that are in different places but acts as near to the next one. One important issue is the handling of requests from the various machines so that all the tasks can be completed in minimum waiting time. This paper presents and examine a new scheduling algorithm with the use of fuzzy control system for scheduling virtual machines between datacenters. The experimental results show the effectiveness of our algorithm by comparing it with the three scheduling techniques First Come First Serve (FCFS), Round Robin (RR) and Honeybee Foraging (HF) algorithm.**

***Keywords*- Cloud computing, Fuzzy Control system, Fuzzy logic, Scheduling algorithm.**

I. INTRODUCTION

Cloud Computing is the ultimate technology in internet. With the use of it, the user may not require the application setup in its machine but can access their own files on any device with internet. R. Buyya[1], defined the cloud as follows: "Cloud is a parallel and distributed computing system consisting of an aggregation of interconnected and virtualized computers that are dynamically provisioned and delivered as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service supplier and clients". Cloud computing can be divided into three classes as per their abstraction level, namely:

A) Software-as-a-service (SAAS)
B) Platform-as-a-service (PAAS)
C) Infrastructure-as-a-service (IAAS)

The framework of cloud computing is shown in fig. 1.

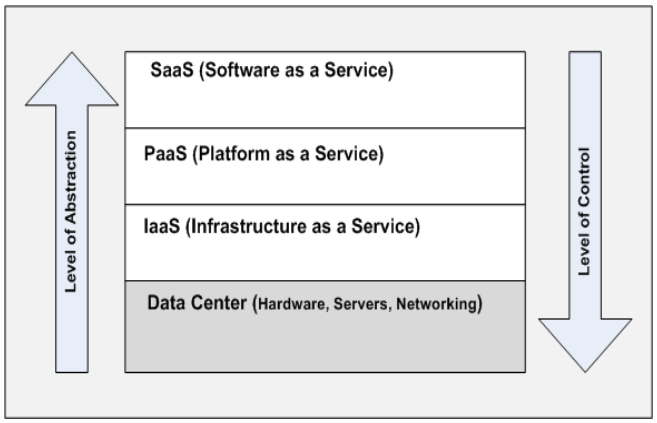

Fig. 1 Framework of cloud computing

A) Software as a Service (SaaS):

In this case, the clients are usually web browsers, provide the point of access to software running on servers. SaaS moves the task of managing software and its deployment to third-party services. Use of SaaS applications tends to reduce the cost of software ownership by removing the need of installation, upgradation of software, as well as reduce the cost of licensing software. Today, SaaS is getting a more prevalent delivery model as the invention of new Web services and service-oriented architecture (SOA) become popular. Example of SaaS are Google Apps, Salesforce, Workday, Concur, Citrix GoToMeeting, Cisco WebEx.

B) Platform as a Service (PaaS):

PaaS functions at a lower level than SaaS, PaaS is built on top of virtualization technology. PaaS makes the development, testing, and deployment of applications quick, simple, and cost-effective. With this technology, a third-party provider can manage OSes, virtualization, servers, storage, networking, and the PaaS software itself. It serves the platform for users who is evolving the software. Enterprises benefit from PaaS because it reduces the amount of coding necessary, automates business policy, and helps migrate apps to hybrid model. Examples of PaaS providers include Heroku, Google App Engine, and Red Hat's OpenShift.

Manuscript received March 25, 2015.
Anindita Kundu, Computer Science Department, Sadabai Raisoni Women's College (SRWC), Nagpur, INDIA, 9970835545

C) Infrastructure as a service (IaaS):

IaaS is comprised of highly automated and scalable compute resources. IaaS clients have direct access to their servers and storage, just as they would with traditional servers but gain access to a much higher order of scalability. IaaS is self-service models for accessing, monitoring, and managing remote datacenter infrastructures, such as compute, storage, networking, and networking servicesIt serves machines, storage and network resources that developers can manage by installing their own operating system, applications and support resources. Iaas can be used by enterprise customers to produce cost efficient and easily scalable IT solutions. The main uses of IaaS include the actual development and deployment of PaaS, SaaS, and web-scale applications. IaaS examples are Amazon Web Services (AWS), Microsoft Azure, Google Compute Engine (GCE), Joyent.

While serve the requests across various virtual machines, cloud computing faces some technical issues also. These are:

A) Task Scheduling:

The server gets requests from various virtual machines at a time. It is very difficult to decide the server, which task should be deal first. For this reason a task scheduler is required. In order to achieve a better balanced load across all the nodes in the cloud environment, the scheduler as discussed in [3], is enhanced by forecasting the execution time of tasks assigned to certain processors and making an optimal decision over the entire group of tasks. This work also assumes centralized scheduling policy where a master processor unit in cloud, collecting all tasks, will take charge of dispatching them to other process units.

B) Security:

Security is one of the main issues in cloud computing. It is the main problem in the case of public cloud. Many people worry about the vulnerability of remote data to such criminals as hackers, thieves, and disgruntled employees. The data must be authenticate and authorized by the user so that the user can safely use the remote data

C) Reliability:

Some people worry that the cloud they use is reliable, their data storage system is trustworthy or not.. Most cloud uses redundant storage techniques, to modify the reliability of data. They also provide a technique to show the degree of reliability.

D) Data Portability:

Some people are concerned that, transferring data to different providers is a difficulty. Porting and converting data dependent upon cloud provider's data retrieval format. This issue can be solved by the evolution of open source programming that supports more cloud providers. And this helps to make the data portable.

E) Data Backup:

Due to system failure, data may be ruined or corrupt. To save the data, backup is required. Cloud providers employ redundant servers and routine data backup processes.

F) Multiplatform Support:

In the evolution of IT, cloud services integrate across different platforms and operating system. The cloud must be platform independent. Multiplatform support requirements will ease as more user interfaces become web-based.

II. RELATED WORK:

The main function of task scheduling is to schedule the resource in the proper time between various virtual machines. 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]. The scheduling algorithm is basically distributed into two parts: static and dynamic. Static algorithms do not depend upon the current state of the system and have prior knowledge regarding system resources and details of all tasks in an application. Dynamic algorithm overcomes the above problem. It take decisions concerning load balancing based upon the current state of the system and don't need any prior knowledge about the system[4]. This has better fault tolerance and overall performance. Example of Static Algorithm is Round robin, First come first serve and many more algorithms. In the round robin [5], processes are divided evenly between all processors. Each new process is assigned to new processor in round robin order. The process allocation order is maintained on each processor locally independent of allocations from remote processors. A Time Quantum is used to distribute the task in the limit of time slice. Advantage of Round Robin algorithm is that it does not require inter-process communication. In the First come first serve technique the request that comes to the ready queue first and get the chance to execute first. The drawback of FCFS is the highest average completion time.

Example of Dynamic Algorithm that we are going to describe is Honeybee Foraging (HF). There is a class of bees called the forager bees which search for food sources, on finding they come back to the beehive to inform this by performing a dance called waggle dance. The display of this dance, gives the idea of the quality or quantity of food and also its distance from the beehive. Scout bees, another type of bees, then follow the foragers to the location of food and then began to reap it. They then return to the beehive and do a waggle dance, which gives an idea of how much food is remaining. As in [7], each server processing a request from its queue calculates a profit or reward, which is analogous to the quality that the bees show in their waggle dance. One measure of this reward can be the amount of time that the CPU spends on the processing of a request. The dance floor in case of honey bees is analogous to an advert board here. A server serving a request, calculates its profit. If this profit

was high, then the server stays at the current virtual server, it completes the task. If it was low, then the server returns to the ready queue and search for another task like the forager.

III. FUZZY LOGIC CONTROL SYSTEM

Fuzzy logic is a mathematical discipline that we use every day and helps us to reach the structure in which we interpret our own behaviors. Fuzzy logic is a good logic to treat random uncertainty, i.e., when the prediction of a sequence of events is not possible. Its basis is formed by "true" and "false" values and Fuzzy Set Theory (FST) through which the values in between –"partially true",

"Partially false"- are determined. Every element has a degree of membership to a set of 0 and 1. In short, Fuzzy Logic is basically a multivalued logic that allows intermediate values to be defined between conventional evaluations like yes/no, true/false, black/white, etc. and a continuous range of truth values in the interval [0, 1]. As discussed in [2], Fuzzy consists of two stages:

A) Fuzzyfication:

In this step, the degree of membership of the input values is assigned to fuzzy sets. The degree of membership is given by a function, μ: X→ [0, 1], where X is the set of input values.

B) De-Fuzzyfication: As [9], It is a process of "rounding it off" from its location in the unit hypercube to the nearest vertex. In this step, a numerical output value is generated from the output set.

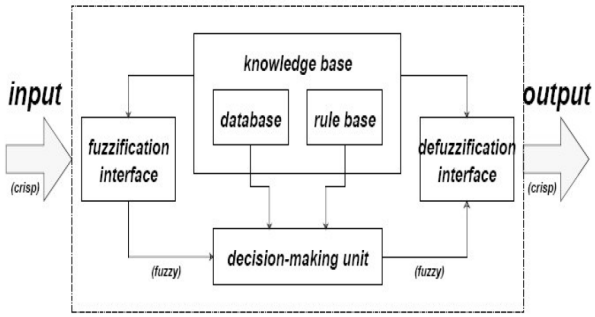

Fig. 2 Fuzzy Control System

A fuzzy control system [8] 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. Fig. 2 shows a fuzzy control system that contains the fuzzy rules that produce an interface engine,i.e., decision making unit that takes two inputs from the server using fuzzifier function and by deffuzification produce a single output.

IV. PROPOSED SYSTEM

This paper proposes a dynamic scheduling algorithm, utilizing the fuzzy logic controller. The controller takes two inputs, the number of requests that received in host from virtual machines and the time taken to complete the tasks in each virtual machine. The number of request as REQ and the time required for the tasks named as BT. These two input produces an output of 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 virtual machine are named as Small, Medium and Big. The time the processes i.e., value of BT are given as 25 to 75.

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

Step 2: Each variable used for example, completion time (the lowest time taken by the CPU for processing) to the host set to 0,1,2 i.e, three virtual machine as small, medium and big.

Step 3: The fuzzy rule based on the given designs. These rules find out the action take place in different input condition. The format is shown in table 1 by method of If-Then.

Step 4: The virtual machine id with less time of completion time and higher profit or reward is selected for the output by using deffuzification to access the host.

In the table 1, we see that when the completion time is less, the first machine get the response- low. As the machine increases, the response is also increases. In the case of moderate time, the second machine gets the advantage and the response get very high. At last, when the completion time is very high, the first machine completes its work fully.

Table 1. Rule Base Fuzzy based approach for scheduling algorithm.

Completion Time	Resources		
	Small	Medium	Big
75	VERY HIGH	LOW	LOW
50	MEDIUM	VERY HIGH	LOW
25	LOW	MEDIUM	VERYHIGH

V. TEST RESULTS

Fuzzy logic can be considered as a way of mapping a real world parameter with any given system. The fuzzy logic mechanism processes on dividing the ranges of parameters into different classes, with each class having a value

between [0, 1]. In our system, it has been observed that comparing with static algorithms (FCFS and Round Robin) and dynamic algorithm (Honeybee Foraging), our proposed algorithm that using fuzzy logic is efficient. Here the data taken for the experimental setup is as present in the following table 2.

Table 2: Experimental data taken against various scheduling techniques.

Resources	Time Taken	Fuzzy	Honeybee Foraging	Round Robin	FCFS
10	100	111	166.3	171	211.2
20	200	166.4	211	231.4	256.2
30	300	215.9	266.9	295.9	560.8
40	400	182.98	341.43	272.98	492.7
50	500	188.24	291.32	258.24	410.2

Fig. 3 (Comparing Completion time) shows that proposed algorithm gives best response time when compare with other traditional approaches. Fig. 3 shows the completion time and the number of datacenters (from 10 to 50) represents the total average completion time or best response time. The figure contains a bar chart for each number of datacenters. By using the proposed algorithm, we analyze that the best technique for choosing the response time is Fuzzy Logic.

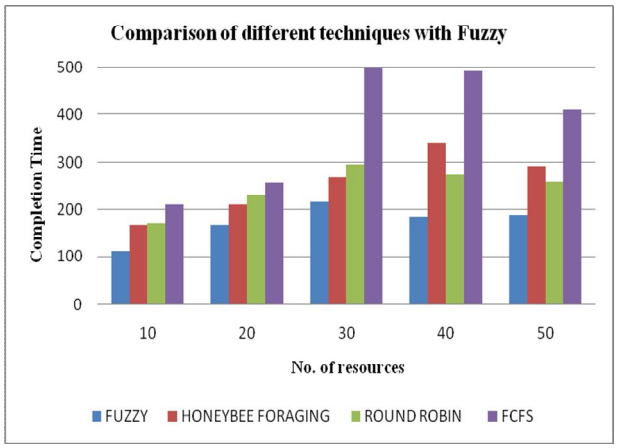

Fig. 3 Calculation of response time with fuzzy logic

VI. ANALYSIS OF DATA

A) Throughput:

Throughput is the amount of data moved successfully from one place to another in a prescribed time period. High throughput is necessary for overall system performance.

B) Fault Tolerance:

It measures the capability of an algorithm to perform uniform load balancing in case of any failure. The basic property of good load balancing algorithm is highly fault tolerable. It enables an algorithm to continue operating properly in the event of some failure. Even a small failure can cause total failure in load balancing.

C) Response Time:

Response time means the time taken to respond in response to a particular load. If it is low it will increase the overall performance. Static load balancing algorithms have shorter response time. Dynamic load balancing algorithms may have relatively higher response time as sometimes redistribution of processes takes place.

D) Resource Utilization:

It is used to ensure the proper utilization of all those resources, which comprised the whole system. If the algorithm is capable to utilize resources, they can be moved to under loaded processors more efficiently. Static algorithm has lower resource utilization as respect to dynamic.

E) Performance:

It is used to check, how ef ficient the system is. Performance is used to check the efficiency of the system. It can be achieved by reduce task response time and by achieving cost.

Techniques	FCFS	Round Robin	Honeybee Foraging	FUZZY
Throughput	NO	YES	YES	YES
Fault Tolerance	NO	NO	NO	YES
Response Time	YES	YES	NO	YES
Resource Utilization	YES	YES	YES	YES
Performance	NO	YES	YES	YES

The Table 3, represents all the comparison according to their Performance.

Table 3: Comparison on techniques according to performance

.VII. CONCLUSION

In this paper the proposed algorithm demonstrates the difference between FCFS, Round Robin, Honeybee Foraging and Fuzzy logic. This paper presented a new

approach for scheduling algorithm on Cloud, namely Fuzzy scheduling technique that provides the resources to the virtual machine id that take smallest number of completion time. It also shows the better performance than the traditional algorithm. It has been also viewed a new pattern that shows the difference in honeybee foraging and round robin.

VIII. REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility, Future generation system, 2009.

[2] Amin Mehranzadeh, Seyyed Mohsen Hashemi International Journal of Applied Information Systems (IJAIS): A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic, Volume 5– No.7, May 2013.

[3] Suriya Begum, Dr. Prashanth C.S.R, IJCSI International Journal of Computer Science Issues : Review of Load Balancing in Cloud Computing, Vol. 10, Issue 1, No 2, January 2013.

[4] N. S. Raghava and Deepti Singh, OPEN JOURNAL OF MOBILE COMPUTING AND CLOUD COMPUTING: Comparative Study on Load Balancing Techniques in Cloud Computing, Volume 1, Number 1, August 2014.

[5] Sandeep Sharma, Sarabjit Singh, and Meenakshi Sharma, "Performance Analysis of Load Balancing Algorithms" World Academy of Science, Engineering and Technology 14 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, 899-911, 2001.

[7] Asitha Micheal, Jalpa Mehta "Load Balancing Techniques in Cloud Computing", International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 2 Issue: 10.

[8] D.Driankov, H.Hellendoom and M.Reinfrank, An introduction to fuzzy control, Springer-Verlag, Berlin, New York, 1993.

[9] Book on "Fuzzy Logic- With Engineering Application" by Timothy J. Rose, Wiley Edition.

Anindita Kundu is an Assistant Professor of Sadabai Raisoni Women's College, Nagpur, affiliated to SNDT Mumbai University. She received the MCA and B.Sc (Computer Science) degree from RTMNU university Nagpur in 2009 and 2006 respectively. She is currently pursuing PHD in the topic of Cloud Computing. She has attended 4 national conferences and one workshop of Cloud Computing. She is a Counsellor of IGNOU. Also she is university exam paper setter as well as moderator. Her current research interest includes Cloud Computing and Fuzzy Logic.