Scientific Journal Impact Factor (SJIF): 3.518
International Journal of Modern Trends in Engineering and Research
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

"MODIFIED FUZZY LOGIC AND ADVANCE PARTICLE SWARM OPTIMIZATION MODEL FOR CLOUD COMPUTING"

Umesh Lilhore[1] and Dr Santosh Kumar[2]
[1,2] PhD Scholar MUIT Lucknow, Department of Computer Science & Engg

Abstract-In cloud computing better performances of computing resources are always a desirable task for cloud researchers. In this research paper we are presenting a new load balancing algorithm for cloud performance improvement, is introduced which based on "Modified fuzzy Logic and Advance Particle Swarm Optimization Model (MFL-APSOM)", to optimize the total execution time of tasks in the workflow applications. The key objective of applying the MFL-APSOM method is to minimize the total tasks execution time by verifying the load fluctuations of the interconnected tasks. The variance of the algorithm considers factors such as load variations and optimization of the data retrieval time. The proposed MFL-APSOM model is validated by applying various workflow structures with different data block sizes. The results are compared with existing HEFT (Heterogeneous Earliest Finish Time) algorithm and Scalable Heterogeneous Earliest Finish Time methods (SHEFT).Simulation results clearly shows that proposed method performs outstanding in terms of cloud performance parameters over existing methods.

Keywords- Cloud Computing, Performance of Cloud computing, MFL-APSOM and Fuzzy logic

I. INTRODUCTION

Cloud computing is a distributed computing paradigm that focuses on providing a wide range of users with distributed access to scalable, virtualized hardware and/or software infrastructure over the internet [2]. Potentially it can make the new idea of 'computing as a utility' in the near future. Despite this technical definition cloud computing is in essence an economic model for a different way to acquire and manage IT resources. An organization needs to weigh cost, benefits and risks of cloud computing in determining whether to adopt it as an IT strategy[1,11]. The availability of advance processors and communication technology has resulted the use of interconnected, multiple hosts instead of single high-speed processor which incurs cloud computing. Recently, public cloud is made available as a pay per usage model while private cloud can be built with the infrastructure of the organization itself. Web Services, Google AppEngine, and Microsoft Azure are examples of public cloud. The service provided by the public cloud is known as utility computing. As benefit, users can access this service "anytime, anywhere", share data and collaborate more easily, and keep data safely in the infrastructure. Although there are risks involved with releasing data onto third party servers without having the full control of it[4,6,9].

II. LOAD BALANCING IN CLOUD COMPUTING

Load balancing is the process of distributing the load among various resources in any system. Thus load need to be distributed over the resources in cloud-based architecture, so that each resources does approximately the equal amount of task at any point of time. Basic need is to provide some techniques to balance requests to provide the solution of the application faster[6,8].

III. PROBLEM FORMULATION

In Cloud Computing, load balancing is one of the major challenges that play an important role in defining the performance of the Cloud system. Without having an effective load balancer, some resources will be under-utilized and some of them will be over-utilized. Hence, to design a competent balanced system, main elements such as load estimation, load comparison and interaction between tasks and resources should be considered.

IV. PROPOSED "MFL-APSOM" MODEL

In the proposed method MFL-APSOM, each task considers as a single particle in a larger population. Chosen particles are controlled by their velocity, direction and magnitude. The novelty of the proposed model can be highlighted by projecting the load fluctuations between the parent tasks and the child tasks in a workflow application. The dynamic load balancing algorithm is applied either as a distributed or a nondistributed manner. Dynamic load balancing method have a great advantages over other method, is that if any node fails, it will not halt the system that can totally affect the overall system performance. The proposed MFL-APSOM, algorithms firstly computes the current load for all the computing resources of a virtual machine of cloud servers. Once the load is computed for all the resources, the load balancing effectively uses the resources dynamically in the process of assigning resources to the corresponding node to reduce the load value. So, for assigning of resources to properly VMs with fuzzy rules, an MFL-APSOM is suggested for load balancing.

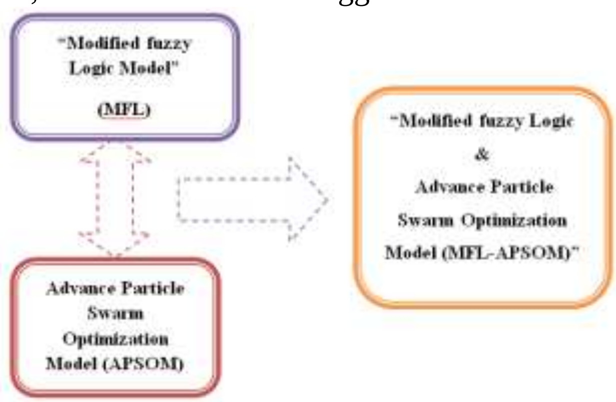

Figure 4 Proposed MFL-APSO models for cloud

V. STEPS FOR MFL-APSO MODEL

Following steps are used in designing of proposed model-

Step 1-The load balancer gets the data from CSP. The load balancer is a middleware intermediate to CSP and cloud data centre.

Step 2- The load balancer processes the data centre and identifies the number of VMs with number of processors in each VM. The load balancer gets properties of VMs such as total memory, used free memory, available memory, latency rate, response Time and execution time is calculated.

Step 3- Resource level percentage is calculated for each VM using this property.

Step 4-By applying the Fuzzy rules, the resource level status is identified as slower, medium and Higher.

Step 5-The load is allocated based on the percentage identified by the resource level percentage metrics. This RLP is an efficiency parameter in this proposed methodology to increase the efficiency of the cloud computing.

International Journal of Modern Trends in Engineering and Research (IJMTER)
Volume 03, Issue 08, [August– 2016] ISSN (Online):2349–9745; ISSN (Print):2393-8161

VI. SIMULATION RESULTS

To evaluate the performance of our proposed algorithm Java programming has been used to extend the Cloudsim simulation tool. Cloudsim is an open source simulation package developed in Java language by the University of Melbourne. It is mainly developed to study the effectiveness of various optimization and performance improvement methods in Cloud Computing. In Clouds, we have several sites containing several virtual machines and storage elements. The proposed algorithm uses 300 VMs in a single data centre. The memory for each VM can share the physical resources on a server. The different type of file size is given and the files allocated based on the resource level percentage taken in allocated VM.

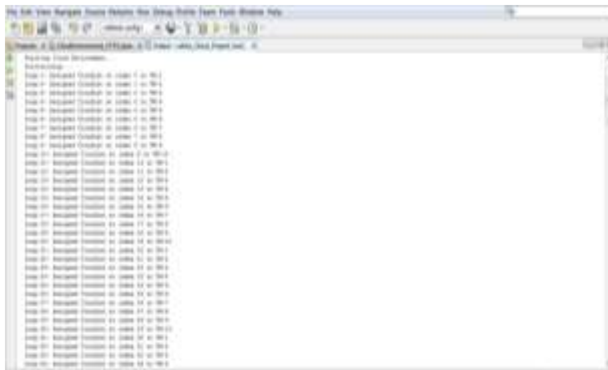

Figure 6- Creating VMs for proposed method

6.1 Response time Analysis- In cloud the response time is calculated by the time required for one packet to travel from load balance server to a virtual machine.

Virtual Machine VM	Average Response Time (MS)		
	Existing HEFT Method	Existing SHEFT Method	Proposed MFL-APSOM Method
10	37713.2	36047.2	35077.1
20	28956.8	25474.1	21365.2
30	18665.2	17878.4	16342.3
50	14345.6	13245.6	12560.3

Table 6.1 Response time

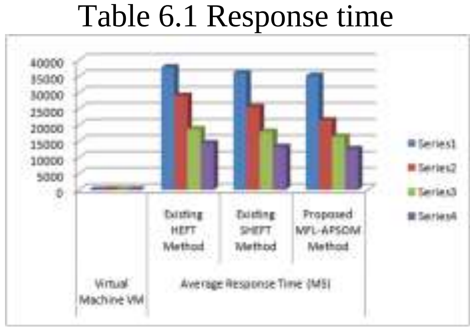

Figure 6.1 Response time

6.2 Execution Time- can be estimated by evaluating the time required for completing the operation time of virtual machines in the cloud data centre. Less execution time shows better performance.

Virtual Machine VM	Average Execution Time (MS)		
	Existing HEFT Method	Existing SHEFT Method	Proposed MFL-APSOM Method
10	30789.98	29877.45	28745.22
20	22478.45	21447.25	19657.25
30	18477.44	17887.47	16447.33
50	14878.57	13554.25	12441.14

Table 6.2 Execution Time

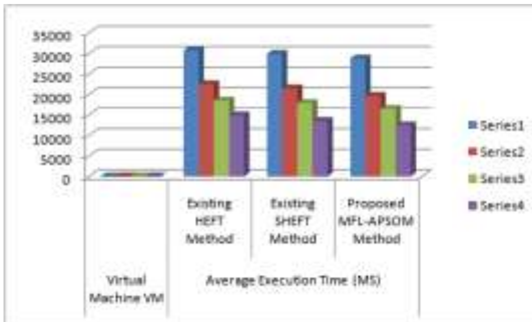

Figure 6.2 Execution Time

6.3 Memory Rate-In this experiment, the maximum memory rate with different values has been set to evaluate the performance of our proposed scheduling model.

Work Flow Data Size in MB	Memory Rate (in MS)		
	Existing HEFT Method	Existing SHEFT Method	Proposed MFL-APSOM Method
64	565	578	661
128	325	345	365
256	778	789	798
512	445	478	487
1024	678	689	702

Table 6.3 Memory Rate

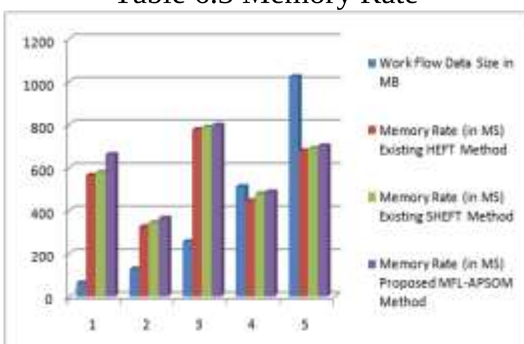

Figure 6.3 Memory Rate

6.4 Workflow Soft Error Rates-To analyze the fault tolerance of the proposed algorithm we have applied soft errors on system's memory to evaluate the output results of the proposed MFL-APSOM experiment.

 ISSN (Online):2349–9745; ISSN (Print):2393-8161

Work Flow Data Size in MB	Soft error, Memory Rate For Proposed MFL-APSOM (in %)	
	5%	10%
64	0.61	0.52
128	0.42	0.39
256	0.48	0.44
512	0.42	0.38

Table 6.4 Work flow Soft error Rates %

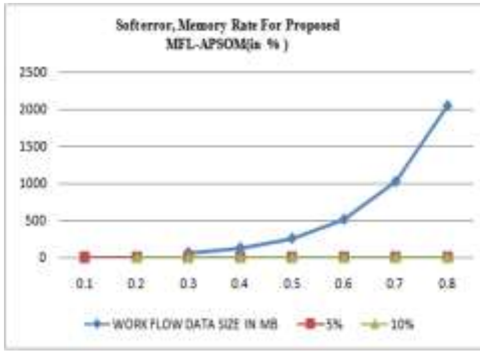

Table 6.5 Work flow Soft error Rates %

Result Analysis- The above experimental results clearly shows that proposed MFL-APSOM method performs outstanding over existing HEFT and SHEFT methods.

VII. CONCLUSIONS

Load balancing is one of the key factors in increasing the performance of the Cloud Computing. To address this shortcoming, a heuristic scheduling algorithm has been presented in this chapter which is designed based on “Modified fuzzy Logic and Advance Particle Swarm Optimization Model (MFL-APSOM)” Model. An experimental result clearly shows that proposed MFL-APSOM method performs outstanding over existing HEFT and SHEFT methods in terms of response time, execution time, error rate and memory rate.

REFERENCES

[1] Alicherry.M. and Lakshman, T.V, “Optimizing data access latencies in cloud systems by intelligent virtual machine placement“, IEEE Communications-Pages: 647- 655-April 2013.
[2] Amit Kumar Das, Tamal Adhikary, Md. Abdur Razza que, Chong Seon Hong, "An intelligent approach for virtual machine and QoS provisioning in cloud computing” PP 462-467-International Conference on Information Networking (ICOIN), 2013.
[3] Brototi Mondala, Kousik Dasgupta and Paramartha Duttab,“Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach” Science Direct, Pages: 783 -789, May 2012.
[4] Chao-Tung Yang, Wei- Li Chou, Ching-Hsien Hsu and Cuzzocrea, A., “On Improvement of Cloud Virtual Machine Availability with Virtualization Fault Tolerance Mechanism”, IEEE Cloud Computing, Pages: 122-129, Dec. 1st 2011.
[5] Dong Oh Son, Hong Jun Choi, Jae Hyung Park and Cheol Hong Kim ,“ Analysis of Memory Management Policies for Heterogeneous Cloud Computing”,IEEE Transactions, Pages:1- 3-June 2013.
[6] Faragardi, H.R., Shojaee, R., Tabani, H. and Rajabi, A.,” An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements“, IEEE Transactions, Pages 315-321, June 2013.
[7] Huankai Chen, Wang, F., Helian,N. and Akanmu, G.“User- priority guided Min- Min scheduling algorithm for load balancing in cloud computing”, IEEE Parallel Computing Technologies (PARCOMPTECH-13), Pages: 1- 8, Feb. 2013.
[8] Myunghoon Jeon, Kwang Ho-Lim, Hyun Ahn and Byoung-Dai Lee,” Dynamic Data Replication Scheme in the Cloud Computing Environment “ ,IEEE Transactions, Pages 40-47, Dec 2012.
[9] Nawsher Khan, A. Noraziah and Tutut Herawan,”A Cloud Architecture with an Efficient Scheduling Technique”, Springer pp 381-388, 2012.

[10] Pawar C.S. and Wagh R.B., "Priority Based Dynamic Resource Allocation in Cloud Computing" , IEEE Cloud and Services Computing(ISCOS), International Symposium, Dec 2012.

[11] Punit Gupta, Mayank Kumar Goyal, Prakash Kumar and Alok Agarwal, "Trust and Reliability Based Scheduling Algorithm for Cloud IaaS", Springer, pp 603-607, 2013.

[12] Xu, Gaochao, Pang, Junjie and Fu, Xiaodong," A load balancing model based on cloud partitioning for the public cloud", IEEE Transactions, Pages-34-39, Feb 2013.

[13] Yaojun Han and Xue mei Luo, "Hierarchical Scheduling Mechanisms for Multilingual Information Resources in Cloud Computing", Springer, Pages 268-273, year 2013.

[14] Zhongyuan Lee and Ying Wang, " A Dynamic Priority Scheduling algorithm on service request scheduling in Cloud Computing", published in IEEE Electronic and Mechanical Engineering and Information Technology (EMEIT-2011), Pages 4665-4669, Aug. 2011.