Implementation and evaluation Efficient Load Balancing and Fault tolerance Mechanism for Cloud Environment

Pooja R. Kathalkar[1], Prof. A. V. Deorankar[2]
[1]*P. G. Student, Department of Computer Science, Government college of Engineering , Amravati, Maharashtra, India*
[2]*Assistant Professor, Department of Computer Science, Government college of Engineering, Amravati, Maharashtra, India*

***Abstract*- Load balancing in the cloud computing environment has an important impact on the performance. Good load balancing makes cloud computing more efficient and improves user satisfaction .Efficient task scheduling and resource management is a challenging problem of distributed computing but it is still in its infant stage in spite of exhaustive research in recent years. In this paper we proposed a hybrid load balancing algorithm for task scheduling. The propose system also provide efficient fault tolerance mechanism.**

I. INTRODUCTION

Cloud computing is the emerging technology in distributed environment consisting of several data centers, servers, virtual machines, load balancers etc. which are connected intelligently. Further, the cloud deals with many things such as storing and retrieving of documents, sharing of multimedia, lending the related resources on pay-as-you go model and much more. Even though there is much advancement in the era of computers and Internet of Things (IoT) with respect to responsiveness, reliability and flexibility, still there is a room for improvement in scheduling, optimal resource allocation and management algorithms since these algorithms come under NP-hard and NP-complete complexity classes. Hence, there is a need to address these set of challenging problems using different techniques. Efficient task scheduling and resource management is a challenging problem of distributed computing but it is still in its infant stage in spite of exhaustive research in recent years.

Load balancing is used to distribute a larger processing load to smaller processing nodes for enhancing the overall performance of system. A load balancer is a expedient that acts as a contrary proxy and allocates system or submission load across a quantity of attendants. Now a day the performance of cloud computing is major issue. For the improvement of the performance of the cloud environment uses load balancing and job scheduling technique.

Scheduling in Cloud Computing

Task scheduling is the process of allocating the resources to the particular job in specific time. The main objective of scheduling is to maximize the resource utilization. Minimizing the waiting time is the goal of scheduling. A good scheduling algorithm yields good system performance. In the cloud there are numerous and distinct resources available. The cost of performing tasks in cloud depends on which resources are being used so the scheduling in a cloud environment is different from the traditional scheduling. In a cloud computing environment task scheduling is a biggest and challenging issue. Aim of load balancing is to clearly understand the consumer requirements, the data and information can be sent and received without taking more time.

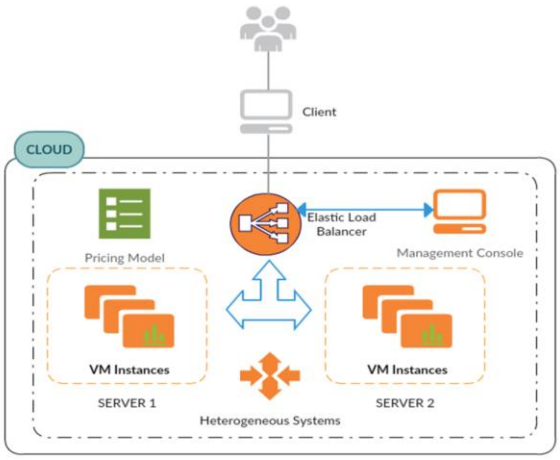

Fig No. 1-The System Model

II. SYSTEM IMPLEMENTAION

Objectives:

The main objective of this dissertation is to implement bio-inspired algorithm in order to provide better load balancing mechanism in cloud environment. Thus, main objectives of this work are to:

- Improve Task response time while also avoiding a situation where some of the nodes are having a huge amount of load while other nodes are doing nothing or idle.To implement Modified AES algorithm for document security.
- Efficient Task Scheduling for Load Balancing By designing a novel Modified Particle Swarm Optimization algorithm, the tasks can be efficiently scheduled with balanced load on VMs.
- Design a backup plan in case system fails during task performance.

Implementation:

In proposed system, Every request coming in for the transaction servers will be first received (Not processed) by a centralized redirection server. Which looking at the current status of load log present with the Redirection server load database the request will be redirected to the least loaded transaction server. The load database is updated with the new request and redirection. The user is then connected to the transaction server to process the request. . Redirection server maintains different kinds of log records in its database -Request log, Redirection log, Login/Logout log, Security log. . System can be reconfigured with additional transaction servers but it will require human interaction with the internal system. Using java EE technologies this system will be built.

What actually load balancing application does ?

load balancing application will find under loaded and best suitable server for the request to be proceed.In other word task scheduling is done according to algorithm applied that is hybrid (Modified particle swarm optimization +Modified cat swarm optimization) Algorithm. This algorithm first find local best from all available server and then global best from local best. Task is assign to Global best system.

Modified Bio-Inspired Algorithm Receive request from user

1. Receive request from user.
2. Encrypt the request processing parameters using Caesar Algorithm
3. find existing loads and maximum load each servers can handle
4. find minimum of the loads by calculating difference between maximum load and current load on each server
5. i.e. for each server i, load(i) = max (maximum_load(i)-current_load(i))
6. first two servers are given priority for redirection. if they are found busy then request is redirected on third server
7. identify current time instance and calculate queuetime required to perform the redirection decision.
8. update current load of the selected server as current_load=current_load+1
9. redirect on Application server on respective Virtual Machine to perform transactions

Fault Tolerance algorithm

1. Enter search query and redirect on application server using proposed load balancing algorithm
2. Calculate the response time required by respective server to process a request and store in db
3. For every request processing evaluate the time required against average response time using response history
4. if time required is more than average time, fault is said to be occured and reported
5. the request is sent back to controller and redirected to the next least loaded server

II. EXPERIMENTAL EVALUATION

In this section, we present the experimental evaluation of the proposed scheme. We investigated the success of our scheme by comparing the result obtain by existing round-robin and propose MPSO.

From the evolution it is clear that although round-robin choose application server in less time but it does not choose efficient server for the task. Propose algorithm select most efficient application server for the task.

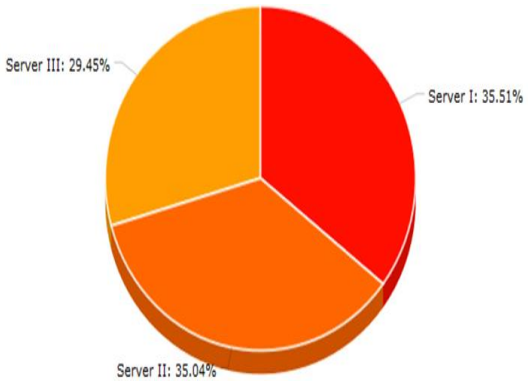

Fig No. 2-Server Performance Report

Mpso algorithm will assign the server to the task according to evalution i.e. in this case server 1 is best suitable so it will be selected while round robin can choose according to sequence.

III. CONCLUSIONS

The proposed Bio-Inspire HYBRID algorithms will provide efficient scheduling in a cloud environment and fault tolerance technique will substantialy decrease the failure rate.

REFERENCES

[1] Y. Shi and R. Eberhart, "A modified particle swarm optimizer", Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on. IEEE, (1998).

[2] Zhao, G. (2014). 'Cost-Aware Scheduling Algorithm Based on PSO in Cloud Computing Environment', International Journal of Grid and Distributed Computing. 7, 1, .33-42

[3] Rodriguez, M. A., &Buyya, R. (2014). Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds. Cloud Computing, IEEE Transactions on, 2(2), 222- 235.

[4] Awad, A. I., El-Hefnawy, N. A., &Abdel_kader, H. M. (2015). Enhanced Particle Swarm Optimization for Task Scheduling in Cloud Computing Environments. Procedia Computer Science, 65, 920-929.

[5] Pandey, S., Wu, L., Guru, S. M., &Buyya, R. (2010, April). A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments. In Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on (pp. 400-407). IEEE.

[6] Abdi, S., Motamedi, S. A., &Sharifian, S. (2014). Task scheduling using Modified PSO Algorithm in cloud computing environment. In International Conference on Machine Learning, Electrical and Mechanical Engineering (pp. 8-9).

[7] Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. IV. pp. 1942–1948.

[8] Binitha S, S Siva Sathya , "A Survey of Bio inspired Optimization Algorithms", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012

[9] Jyoti Thaman, Manpreet Singh, "Current Perspective in Task Scheduling Techniques in cloud Computing: a review", International Journal in Foundations of Computer Science & Technology (IJFCST) Vol.6, No.1, January 2016

[10] Shridhar Domanal, Ram Mohana Reddy Guddeti, Rajkumar Buyya," A Hybrid Bio-Inspired Algorithm for Scheduling and Resource Management in Cloud Environment", IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. X, NO. X, JULY 2016