International Journal of Research In Science & Engineering
Volume: 4 Issue: 2 March-April 2018

e-ISSN: 2394-8299
p-ISSN: 2394-8280

ANALYSIS OF DIFFERENT RESOURCE SCHEDULING ALGORITHM IN CLOUD COMPUTING

Snehil Agrawal[#1],Prof. Sumit Nigam[*2]

[#]CSE Department
SIMS, Indore,India

[1]snehil.agrawal@sims-indore.com

[3]sumit.nigam@sims-indore.com

Abstract **-Cloud Computing is providing a facility to organising and accessing application or services through internet from anywhere. We can also say that, in Cloud Computing scenario user don't need to install, Purchase a software and hardware at their end theyOnly need internet to use Cloud Computing Services. The main aim of Cloud Computing is to provide "Subscription Based Services" means it based on "pay-as-per-use". In this paper we provide different load balancing technique in cloud computing and terminology which is relevant to the load balancing.**

Keywords: Cloud Computing, SchedulingAlgorithms, Loadbalancing, Cloud Sim,cloud sim Architecture.

I. INTRODUCTION

Cloud computing is a computing paradigm, where a large pool of systems are connected in private or public networks, to provide dynamically scalable infrastructure for application, data and file storage. With the advent of this technology, the cost of computation, application hosting, content storage and delivery is reduced significantly. Cloud Computing delivers Infrastructure, platform and software (application) as services, which are made available as pay-per-use model to consumers.

It means it is providing services as subscription based. The Beauty of the Cloud Computing is that another company hosts your application. This means that they handle the costs of servers, they manage the software updates, and depending on how you craft your contract-you pay less for the services. Cloud Computing is made up by a combination of two terms in field of technology. First, is Cloud and second is Computing. Cloud is a heterogeneous resources, it is a mesh of huge infrastructure refers to both the application delivered to end users as services over the internet and hardware and software system in datacenters that is responsible for providing those services. In order to make efficient use of these resources and ensure their availability to the end users "Computing" is done based on certain criteria specified in SLA (Service Level Agreement) [1].

e-ISSN: 2394-8299
p-ISSN: 2394-8280

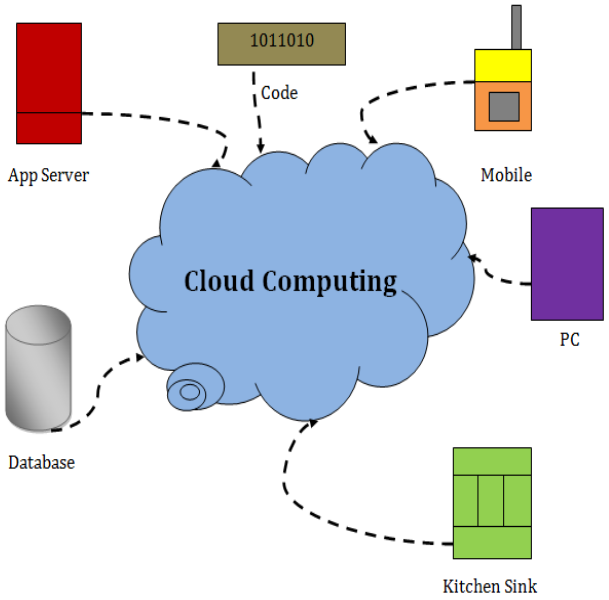

Figure 1: Cloud Computing Scenario [2]

*A. **Resource Scheduling in Cloud***

Resource Scheduling is a collection of techniques used to calculate the resources required delivering the work and when they will be required.Resource Scheduling in cloud is a challenging job and the scheduling of appropriate resources to cloud workloads depend on QoS requirements of cloud applications. In Cloud Environment, heterogeneity, uncertainty and dispersion of resources encounters problems of allocation of resources, which cannot be addressed with existing resource allocation policies. Researchers still face troubles to select the efficient and appropriate resource scheduling algorithm for a specific workload from existing literature of resource scheduling algorithm.Here trying to discuss existing scheduling algorithm and its challenges and advantages.

B. Advantages of Resource Scheduling

- ✓ Reduce costs and energy use which adds to an organisation's bottom line.
- ✓ Open up control of resources to the people who use them, empowering them to work in the best possible ways and so further meet many of the key workplace objectives of the organisation.
- ✓ Support the agile workforce such as hospitality management, videoconferencing and sophisticated room booking systems.
- ✓ Able to generate data in real-time that shows how resources are used, it is also an important tool for strategic decision making.
- ✓ Awareness of how specific resources are used including how often, in what way and by whom.
- ✓ Minimize the need for costly and disruptive relocations and refurbishments by utilising existing space and resources better.
- ✓ Help to tailor both the physical and technological working environment to the precise needs of individuals and teams.

C. Challenges in Efficient scheduling

Based on the survey a scheduling fabric can be enforced by utilizing the different parametric quantities. The efficiency of the energy usage is main issue that took a lot concern. But Scheduling is unmatchable issues in the management of diligence performance in a cloud environment. It must focus on cost, time, energy efficiency and load balancing of the data centres. Paleness imagination allotment sets a critical use in scheduling.

II. BACKGROUND

A. Simulation in Cloud: Cloudsim

CloudSim: In CloudSim, cloud computing infrastructures and application services allowing its users to focus on specific system design issues that they want to investigate Simulation in a CloudSim means implementation of actual environment towards benefit of research. The users or researcher actually analyse the proposed design or existing algorithms through simulation. Resources and software are shared on the basis of client's demand in cloud environment. Essentially, dynamic utilization of resources is achieved under different conditions with various previous established policies. Sometime it is very much difficult and time consuming to measure performance of the applications in real cloud environment. In this consequence, simulation is very much helpful to allow users or developers with practical feedback in spite of having real environment. In this research work, simulation is carried out with a specific cloud simulator, CloudSim [1]. Figure 2 shows Layered CloudSim architecture.

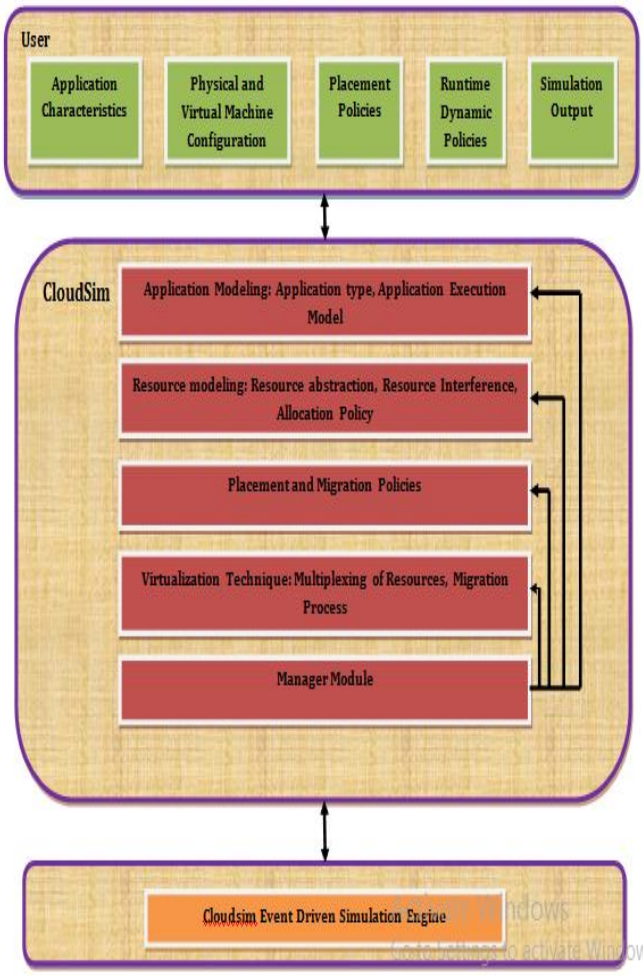

Figure 2: Block diagram of CloudSim [2]

Figure 2 shows the layered implementation of the CloudSim software framework and architecturalcomponents. At the lowest layer is the SimJava discrete event simulation engine [3] that implements the core functionalities required for higher-level simulation frameworks such as queuing and processing of events, creation of system components (services, host, data center,broker, virtual machines), communication between components, and management of the simulation clock. Next follows the libraries implementing the GridSimtoolkit [3] that support high level software components for modeling multiple Grid infrastructures, including networks and associated traffic profiles, and fundamental Grid components such as the resources, data sets, workload traces, and information services. The CloudSim is implemented at the next level by programmatically extending the core

functionalities exposed by the GridSim layer. CloudSim provides novel support for modelling and simulation of virtualized Cloudbased data centre environments such as dedicatedmanagement interfaces for VMs, memory, storage, and bandwidth. CloudSim layer manages the instantiation and execution of core entities (VMs, hosts, data centres, application) during the simulation period. This layer is capable of concurrently instantiating and transparently managing a large scale Cloud infrastructure consisting ofthousands of system components. The fundamental issues such as provisioning of hosts to VMs based on user requests, managing application execution, and dynamic monitoring are handled by this layer. A Cloud provider, who wants to study the efficacy of different policies in allocating its hosts, would need to implement his strategies at this layer by programmatically extending the core VM
Provisioningfunctionality. There is a clear distinction at this layer on how a host is allocated to different competing VMs in the Cloud. A Cloud host can be concurrently shared among a number of VMs that execute applications based on user-defined QoS specifications. The top-most layer in the simulation stack is the User Code that exposes configuration related functionalities for hosts (number of machines, their specification and so on), applications (number of tasks and their requirements), VMs, number of users and their application types, and broker scheduling policies. A Cloud application developer can generate a mix of user request distributions, application configurations, and Cloud availability scenarios at this layer and perform robust tests based on the custom Cloud configurations already supported within the CloudSim. As Cloud computing is a rapidly evolving research area, there is a severe lack of defined standards, tools and methods that can efficiently tackle the infrastructure and application level complexities. Hence in the near future there would be a number of research efforts both in academia and industry towards defining core algorithms, policies, application benchmarking based on execution contexts. By extending the basic functionalities already exposed by CloudSim, researchers would be able to perform tests based on specificscenarios and configurations, hence allowing the development of best practices in all the critical aspects related to Cloud Computing.

III. LITERATURE SURVEY

To understand any technology or system a full survey is required to get the information about the whole system means what is the phenomenon inside that technology?, what are the issues in that technology? And what are the pros and cons of that? These are the basic things which we need to understand before doing any work. [9]Provide an extensive survey of cloud computing and highlight the benefits of cloud computing. The survey also discusses the main issues in particular area and provide different approach to handle these issues. They also explain what are the similarity or dissimilarity between cloud computing and mobile cloud computing. The main issues for cloud computing are security, power management, data management and load balancing. The main focus for this research is to identify the issue of load distribution in cloud computing.

[21]Proposed an article to define an improved load distribution algorithm model for the public cloud. This model isbuilt on the cloud distribution concept having a switch mechanism to respond to various situations using different terminology. This algorithm of load balancing uses the game theory to improve the efficiency in the public cloud environment. The public cloud has varioussystems with distributed computing resources in various different areas. Thus, in this architecture the public clouds divided in too many cloud partitions, which helps them in simplifying the load balancing even when the area is very large and complex. The cloud contains the main organizer that select the suitable subdivision for coming request on the other side the balance of every cloud subdivision selects the best load Distributionstrategy. The Round Robin algorithm is used here because it is simple; it is very simple technique because it used the phenomenon of time slices. In this the time divided in too many slots and every system is having a particular time slot, it follows the principle of time scheduling.

[11]Describe the use of load distribution. The aim of doing load distributionis to reduce energy consumption and achieve maximum resource utilization this can reduce the number of job rejection. Load balancing in cloud computing can improve and maintain the performance of cloud system. The main goals of load distributionis improve the system performance, stabilitymaintenance of the system, Load balancing can increase flexibility of the system Means a new node can easily be add or remove in the network; load balancing creates backups by which system become fault tolerant.

They are using three algorithms for load balancing and comparing them on the basis of some parameters. Algorithms are Round robin, Throttled load distribution, equally spread current execution load. Basically there are two types of load distribution algorithms:-In static load distribution algorithm the system not have the knowledge of current situation of the system so the decision of dividing the load can be done without the knowledge of the current situation of the system. It requires knowledge about the applications and resources of the system. Dynamic: In the dynamic load balancing algorithm the current state of the system is necessary the shifting of work can be done only the basis of current state of the system.

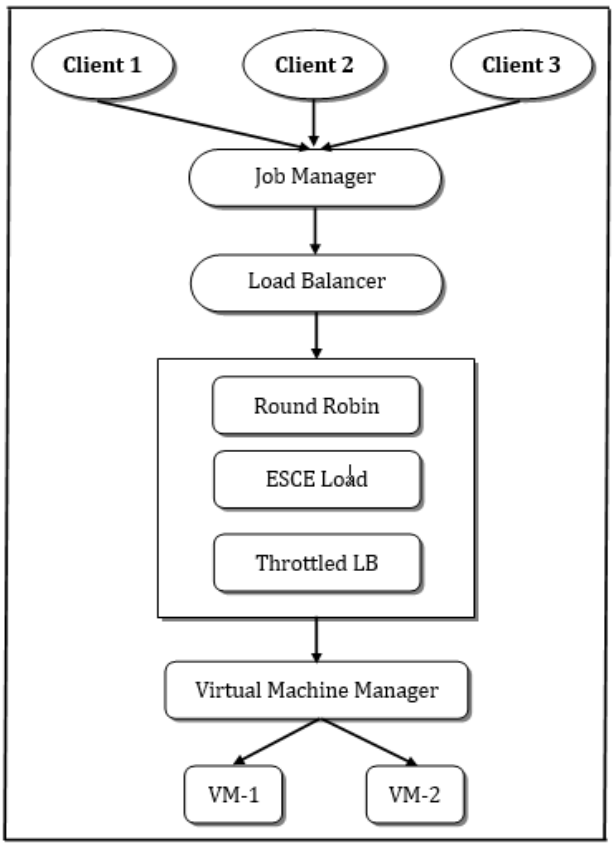

Figure 3: Load balancing algorithm execution [4]

Below figure represents the basic scenario of load distribution algorithm execution. When client request for a particular resource then the job manager manage the coming request and make a queue of the request then it passes the request in the cloud network for execution when their will be the flood of requests in the network then load balancer comes in action and divide the request among the servers with the help of load balancing algorithm.

[12] In the cloud computing,Distributed file systems are main building blocks. In these file systems, nodes continuously provide computing and storage functions, a file is divided into a number ofChunks or parts reside in different nodes so that Map Reduce tasks can be Performed in parallel over the nodes. However, in a cloud computing environment, failure can be occur, and nodes can be upgraded, replaced, and added and delete in the system. Files can be dynamically created, deleted, and appended. This can cause load imbalance in a distributed file system, the file chunks are not distributed as uniformly as possible among the nodes. If we provide a central controller to balance the load and divide the file chunks then there is a chance that if the work load increases then this central controller would be crash so there is bottleneck and single point of failure. So they proposed a fully distributed load rebalancing algorithm to handle the load imbalance problem. This algorithm is compared against a centralized approach in a Production system and a competing distributed solution presented in the literature. The solution of load balancing problem is solved by DHT (distributed hash table), in this chunks are divided in different parts then every chunk has controller which control the chunk movement. The algorithm is compared with centralized algorithm in the Hadoop HDFS

production system and gave a good performance in terms of load imbalance factor, movement cost, and algorithmic overhead. This algorithm provide fast convergence rate.

[6]Proposed a centralized hierarchical CMS system composed of a resource controller and number of server and clusters, every server cluster is manager by cluster head, and we assume that each servers in different server clusters gives different services. The procedure of CMS is as follows. when every time the clients requests is coming in the CMS for multimedia service tasks, the resource manager in the CMS transmit those requests to various server clusters according to the type of the requested resource. Basically, the cluster controller of every server cluster divides the assigned task to a server in the server cluster. Here dynamic multiservice scenario is used,in this every cluster of servers hold a particular kind oftask, and each client requests a particular kind of service at a different-differenttime. This scenario can becomes under an integer linear programming problem. This problem is further solved by an efficient genetic algorithm, which is suitablefor the dynamic problems. And the results shows that theproposed genetic algorithm is efficiently match with dynamicmultiservice load distribution in cloud based multimedia service. In this centralized hierarchical cloud-basedmultimedia system (CMS) is considered having acluster heads, resource manager, and server clusters. In this the resource coordinatordivide clients' requests in the server clusters according to the characteristics of the task, and after this every cluster head divide the assigned task among the servers inside its server cluster. There is challenge to design a good load balancing algorithmthat divide the multimedia service work load on servers with the minimal cost for sending Multimedia data across the server clusters and clients.

[8]Proposed a decentralized cluster-based algorithm in which dynamic load balancing is achieved in the cloud architecture. In this proposed algorithm they present the following important features: the algorithm supports heterogeneity, provide scalability, provide low network congestion and eliminates the bottleneck node because it is decentralized in nature. CloudSim is used to evaluate the simulation result. They used the master slave architecture in this master manage the slaves and distributes coming work load among the slaves. They used round robin algorithm for distribution of load from master to slave.

[24] Proposed two load-balancing protocols. They are using distributed hash table (DHT) For assigning request to nodes or peer in a P2P system, items and nodes are resides in a common address space, and nodes should store every items place close on the Address space. The first protocol manages the Distribution of the key address space among the nodes and

[16]Proposedrelative load distribution algorithms in which they compare three load balancing techniques that is biased random sampling, honeybee foraging and active clustering. They compare these algorithms on the basis of through put. Random samplingandActive clustering gives superior result when the processing nodes are increased.

[6]When there is change in the workload in the various servers because of delay in the network, thiswill influence the execution of the load distribution algorithm. In this paper, they perform a simple survey of the problem and provided two efficient delay adjustment schemes to solve this problem. They used global heat diffusion algorithm. They used both centralized and decentralized algorithm. The problem of uniform load distribution they use two techniques: adaptive adjustment and uniform adjustment scheme. Uniform adjustment scheme performs a uniform balancing of the load between the various servers. The adaptive scheme can communicate with neighbor serverswithout the need and can perform limited degree of user tracking but to. These schemes improve the performance of the load distributionalgorithm.

[11]Presentedan dynamic load distribution algorithm for cloud workload environment by which the load can be conveyed in an adjusted methodology, as well as it dispenses the load deliberately and consistently by checking certain parameters like number of request the server is taking care of as of now. It adjusts the load on the overloaded hub to under loaded hub with the goal

that reaction time from the server will reduction and execution of the framework is expanded. Existing Load Balancing methods/Algorithms that have been considered to a great extent concentrate on decreasing overhead, diminishing the relocation time and enhancing execution and so forth. However the reaction to demand proportion is once in a while considered. In proposed algorithm, the solicitation is allocated as right on time as could be allowed to the fitting data centre. In view of the dynamic way of the algorithm, there is no compelling reason to have the former information of the

Major Challenges –

- ✓ Throughput
- ✓ Associated Overhead
- ✓ Fault Tolerant
- ✓ Migration Time
- ✓ Response Time
- ✓ Resource Time
- ✓ Scalability
- ✓ Performance

IV. PROPOSED WORK

In this paper we discussed about algorithms for balancing load in cloud computing. And we will do comparison between round robin and POS algorithm to optimize response time in terms of same parameters.

V. CONCLUSION

In this paper we talked about static and dynamic algorithm and we found Each algorithm is observed and their scheduling criteria like average response time, data center service time and total cost of different data centers are found. Better response time can be expected if we apply some evolutionary algorithms such as PSO, ACO, and ABC instead of classical algorithms.

REFERENCES

[1] Devi, D. Chitra, and V. RhymendUthariaraj. "Load balancing in cloud computing environment using improved weighted round robin algorithm for nonpreemptive dependent tasks." The scientific world journal 2016 (2016).

[2] Rodrigo N. Calheiros and Rajiv Ranjan, "CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms. Software", Practice and Experience, Wiley Press, New York, USA, 2010 (in press, accepted on June 14, 2010)

[3] Foram F Kherani, Prof.Jagnesh Vania "Load Balancing in Cloud Computing" 2014 IJEDR Volume 2, Issue 1

[4] Berwal, M. and C. Kant (2015) "Load Balancing in Cloud Computing."

[5] Deng, Y. and R. W. Lau (2012)."On delay adjustment for dynamic load balancing in distributed virtual environments."Visualization and Computer Graphics, IEEE Transactions on**18** (4): 529-537.

[6] Desai, T. and J. Prajapati (2013). "A Survey Of Various Load Balancing Techniques And Challenges In Cloud Computing." International Journal of Scientific & Technology Research**2** (11): 158-161.

[7] Dhurandher, S. K., M. S. Obaidat, et al. (2014). A cluster-based load balancing algorithm in cloud computing. Communications (ICC), 2014 IEEE International Conference on, IEEE.

[8] Fernando, N., S. W. Loke, et al. (2013). "Mobile cloud computing: A survey." Future Generation Computer Systems**29** (1): 84-106.

[9] Godfrey, B., K. Lakshminarayanan, et al. (2004). Load balancing in dynamic structured P2P systems. INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE.

[10] Haryani, N. and D. Jagli (2014) "Dynamic Method for Load Balancing in Cloud Computing."

[11] Hsiao, H.-C., H.-Y.Chung, et al. (2013). "Load rebalancing for distributed file systems in clouds." Parallel and Distributed Systems, IEEE Transactions on**24** (5): 951-962.
[12] Kumar, S., D. S. Rana, et al. "Fault Tolerance and Load Balancing algorithm in Cloud Computing: A survey."
[13] Lin, C.-C., H.-H.Chin, et al. (2014)."Dynamic multiservice load balancing in cloud-based multimedia system."Systems Journal, IEEE**8**(1): 225-234.
[14] Mondal, B., K. Dasgupta, et al. (2012). "Load balancing in cloud computing using stochastic hill climbing-a soft computing approach." Procedia Technology**4**: 783-789.
[15] Randles, M., D. Lamb, et al. (2010).A comparative study into distributed load balancing algorithms for cloud computing. Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on, IEEE.
[16] Rao, A., K. Lakshminarayanan, et al. (2003). Load balancing in structured P2P systems. Peer-to-Peer Systems II, Springer**:** 68-79.
[17] Sharma, S., S. Singh, et al. (2008)."Performance analysis of load balancing algorithms."World Academy of Science, Engineering and Technology**38**: 269-272.
[18] Thomas, N., M. Thomas, et al. (2015). "Multimedia Streaming Using Cloud-Based P2P Systems."Procedia Computer Science**57**: 25-32.
[19] Xu, G., J. Pang, et al. (2013). "A load balancing model based on cloud partitioning for the public cloud." Tsinghua Science and Technology**18**(1): 34-39.
[20] Xu, Z. and L. Bhuyan (2006).Effective load balancing in p2p systems.Cluster Computing and the Grid, 2006.CCGRID 06.Sixth IEEE International Symposium on, IEEE.
[21] Aguilar-Garcia, A., R. Barco, et al. (2015). "Load balancing mechanisms for indoor temporarily overloaded heterogeneous femtocell networks." EURASIP Journal on Wireless Communications and Networking2015(1): 1-14.
[22] Lymberopoulos, Leonidas, ChariklisPittaras, Maria Grammatikou, SymeonPapavassiliou, and Vasilis Maglaris. "PLATON: Peer-to-Peer load adjusting tree overlay networks." Peer-to-Peer Networking and Applications 5, no. 2 (2012): 125-142.
[23] Karger, David R., and Matthias Ruhl. "Simple efficient load balancing algorithms for peer-to-peer systems." In Proceedings of the sixteenth annual ACM symposium on Parallelism in algorithms and architectures, pp. 36-43. ACM, 2004.
[24] Le, Viet Dung, Gilbert Babin, and Peter Kropf, "BALLS: a structured peer-to-peer system with integrated load balancing." In Annales des télécommunications, vol. 61, no. 11-12, pp. 1229-1281. Springer-Verlag, 2006.