ISSN 2277-3061

ENHANCING JOB SCHEDULING IN CLOUD ENVIRONMENT: A REVIEW

Mrs. Amita Rani [1], Dr. Mohita Garg [2]
[1] Research Scholar, Department of Computer Engineering, NWIET, Moga
amu.happy@ymail.com
[2] Associate Professor, Department of Computer Engineering, NWIET, Moga
mohita_cse@northwest.ac.in

ABSTRACT

Cloud computing is Internet based development and use of computer technology. It is a style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet. Users need not have knowledge of, expertise in, or control over the technology infrastructure "in the cloud" that supports them. Scheduling is one of the core steps to efficiently exploit the capabilities of heterogeneous computing systems. The problem of mapping meta-tasks to a machine is shown to be NP-complete. The NP-complete problem can be solved only using heuristic approach. There are a number of heuristic algorithms that were tailored to deal with scheduling of independent tasks. Different criteria can be used for evaluating the efficiency of scheduling algorithms. The most important of them are makespan, flowtime and resource utilization. In this paper, a new heuristic algorithm for scheduling meta-tasks in heterogeneous computing system is presented. The proposed algorithm improves the performance in both makespan and effective utilization of resources by reducing the waiting time.

Keywords

Cloud Computing, Load Balacing, Virtual Machine, Data Center, Data Center Broker.

Council for Innovative Research
Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol.14, No.3

www.ijctonline.com, editorijctonline@gmail.com

ISSN 2277-3061

INTRODUCTION

Internet has been a driving force towards the various technologies that have been developed since its inception. Arguably, one of the most discussed among all of them is Cloud Computing. Over the last few years, Cloud computing paradigm has witnessed an enormous shift towards its adoption and it has become a trend in the information technology space as it promises significant cost reductions and new business potential to its users and providers [1]. Cloud computing can be defined as "Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers".

Load Balancing[2] is an emerging computer paradigm where data and services placed massively in the cloud and which can be accessed from any connected devices over the internet. It is known as provider of dynamic services using very large scalable and virtualized resources over the internet. Load Balancing is a computer networking method to distribute workload across multiple computer clusters, network links or other resources to achieve optimal resource utilization, maximize throughput, minimize response time and avoid overload. It is a mechanism that distributes the dynamic local work load[3] evenly across all the nodes in the whole cloud to avoid a situation where some nodes are heavily loaded while others are idle. Its goal is to improve the overall performance and resource utility of the system.

A. CLOUD COMPUTING: AN OVERVIEW

Computing can be described as any activity of using and/or developing computer hardware and software. It includes everything that sits in the bottom layer, i.e. everything from raw compute power to storage capabilities. Cloud computing [1] ties together all these entities and delivers them as a single integrated entity under its own sophisticated management.

Cloud is a term used as a metaphor for the wide area networks (like internet) or any such large networked environment. It came partly from the cloud-like symbol used to represent the complexities of the networks in the schematic diagrams. It represents all the complexities of the network which may include everything from cables, routers, servers, data centers and all such other devices.

Computing started off with the mainframe era. There were big mainframes and everyone connected to them via "dumb" terminals. This old model of business computing was frustrating for the people sitting at the dumb terminals because they could do only what they were "authorized" to do. They were dependent on the computer administrators to give them permission or to fix their problems. They had no way of staying up to the latest innovations. The personal computer was a rebellion against the tyranny of centralized computing [4] operations. There was a kind of freedom in the use of personal computers. But this was later replaced by server architectures with enterprise servers and others showing up in the industry. This made sure that the computing was done and it did not eat up any of the resources that one had with him. All the computing was performed at servers. Internet grew in the lap of these servers. With cloud computing we have come a full circle. We come back to the centralized computing infrastructure. But this time it is something which can easily be accessed via the internet and something over which we have all the control.

B. SERVICE MODELS

There are different types of services are provides by cloud models like: Software as a Service(SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) [6] which are deployed as public cloud, private cloud, community cloud and hybrid clouds.

1) Software as a Service (SaaS):- The capability provided to the consumer is to use the some applications which is running on a cloud infrastructure. The applications are accessible from many devices through an interface such as a web browser (e.g., web-based email). The consumer does not control the cloud infrastructure which includes network, and servers, all operating systems, and provides storages.

2) Platform as a Service (PaaS):- PaaS [5] provides all the resources that are required for implementation of applications and all services completely from the Internet. In this no downloading or installing is required of any software. The capability provided to the consumer is to deploy onto the cloud infrastructure .Consumer uses all the applications by using different programming languages and tools which are provide by the provider. Any consumer has not any control on cloud infrastructure including all networks, servers and operating systems, but has control over the applications which they deployed.

3) Infrastructure as a Service (IaaS):- The capability provided to the consumer is to access all the processing, storage , networks and other many fundamental computing resources . Consumer [5] [6] is able to deploy arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage ,deployed application ,and possibly limited control of select networking components

C. DEPLOYMENT MODELS

Depending on infrastructure ownership, there are four deployment models of cloud computing [6].

1) Public Cloud: - Public cloud [9] allows users to access the cloud publicly. It is access by interfaces using internet browsers. Users pay only for that time duration in which they use the service, i.e., pay-per-use.

2) Private Cloud:- A private clouds [10] operation is with in an organization's internal enterprise data center. The main advantage here is that it is very easier to manage security in public cloud. Example of private cloud in our daily life is intranet.

3) Hybrid Cloud: - It is a combination of public cloud [11] and private cloud. .It provide more secure way to control all data and applications .It allows the party to access information over the internet. It allows the organization to serve its needs in the private cloud and if some occasional need occurs it asks the public cloud for some computing resources.

4) Community Cloud:-When cloud infrastructure construct by many organizations jointly, such cloud model is called as a community cloud. The cloud infrastructure could be hosted by a third-party provider or within one of the organizations in the community.

LOAD BALANCING

It is the mechanism of distributing the load among various nodes of a distributed system to improve both resource utilization and job response time while also avoiding a situation where some of the nodes are heavily loaded while other nodes are idle or doing very little work. It also ensures that all the processor in the system or every node in the network does approximately the equal amount of work at any instant of time. Load Balancing [5] is done with the help of load balancers where each incoming request is redirected and is transparent to client who makes the request. Based on predetermined parameters, such as availability or current load, the load balancer uses various scheduling algorithm to determine which server should handle and forwards the request on to the selected server. To make the final determination, the load balancer retrieves information about the candidate server's health and current workload in order to verify its ability to respond to that request. Load balancing solutions can be divided into software-based load balancers and hardware-based load balancers. Hardware-based load balancers are specialized boxes that include Application Specific Integrated Circuits (ASICs) [32] customized for a specific use. They have the ability to handle the high speed network traffic whereas Software-based load balancers run on standard operating systems and standard hardware components.

Some load balancers provide a mechanism for doing something special in the event that all backend servers are unavailable. This might include forwarding to a backup load balancer, or displaying a message regarding the outage. Load balancing gives the IT team a chance to achieve a significantly higher fault tolerance. It can automatically provide the amount of capacity needed to respond to any increase or decrease of application traffic.

It is also important that the load balancer itself does not become a single point of failure. Usually load balancers are implemented in high-availability pairs which may also replicate session persistence data if required by the specific application.

• O.M.elzeki, et.al,(2012): discusses in Improved Max-Min Algorithm in Cloud Computing that focuses on the cloud computing which further deals with the allocation of the tasks to the resources while observing different parameters like Waiting time, Average waiting time, Turn Around time, Processing cost. So, an algorithm named as Max-Min in improved manner from load balancing has been shown to overcome such kinds of problems. The algorithm calculates the expected completion time of the submitted tasks on each resource. Then the task with the overall maximum expected execution time is assigned to a resource that has the minimum overall completion time.

• Amandeep Kaur Sidhu, (April-2013) discussed in Analysis of load balancing techniques in cloud computing that aims to share of data, calculations and resources transparently over a scalable network of nodes.

• Gytis Vilutis* et al,(2012) discussed that it is complicated to determine the quantity of resources in order to satisfy work load with peaks. Some projects are lost because of under provisions of cloud resources which leads to postponed work and that can reduce the probability of projects not to be executed. The author discussed two problems: to deploy maximum quantity of servers wishing to satisfy all its users requirement and to keep minimum quantity of servers in full usage even the users load is at minimum level.

• Upendra Bhoi* et al,(April2013) Discussed that in enhanced Max-Min Task Scheduling Algorithm in cloud computing helps in supplying a high performance computing based on protocols which allowed shared computation and storage over long distances. It depends upon expected execution time instead of completion time. Max-Min algorithm assign task with maximum execution time to resource produces minimum completion time while Enhanced Max-min assign task with average execution time to resource produces minimum execution time.

• Klaitham Al Nuaimi* et al, (2012) discuss about the overall approach to enhance the performance of cloud. Cloud provides a flexible and easy way to keep and retrieve data and files. Especially for making large data sets and files. In Load Balancing algorithm are classified as static and dynamic algorithm. Static algorithm is for stable and homogenous environments whereas dynamic are more flexible and can adapt to various changes by providing better results.

• Tushar Desai* et al,(Nov 2013) discusses about the imerging technology i.e a new standard of large scale distributed computing and parallel computing. It provides shared resources, information or other resources as per clients requirements at specific times.For better management of available good load balancimg techniques are required. And through beter load balancing in cloud , performance is increased and user gets better services. So in this author has discussed many different load balancing techniques used to solve the issue in cloud computing environment.

• Haozheng Ren* et al, (2012) explains that The load balancing algorithm is an important means to achieve efficient utilization of resources. This paper presents a dynamic load balancing algorithm based on virtual machine migration under cloud computing environment .It Minimizes the allocation time of user requests and Maximize system throughput.

METRICS FOR LOAD BALANCING IN CLOUD

Various metrics considered in existing load balancing techniques in cloud computing are discussed below-

- **Scalability** is the ability of an algorithm to perform load balancing for a system with any finite number of nodes. This metric should be improved.
- **Resource Utilization** is used to check the utilization of re-sources. It should be optimized for an efficient load balancing.
- **Performance** is used to check the efficiency of the system. This has to be improved at a reasonable cost, e.g., reduce task response time while keeping acceptable delays.
- **Response Time** is the amount of time taken to respond by a particular load balancing algorithm in a distributed system. This parameter should be minimized.
- **Overhead Associated** determines the amount of overhead involved while implementing a load-balancing algorithm. It is composed of overhead due to movement of tasks, inter-processor and interprocess communication. This should be minimized so that a load balancing technique can work efficiently.

LOAD BALANCING ALGORITHMS

In order to balance the requests of the resources it is important to recognize a few major goals of load balancing algorithms:

a) **Cost effectiveness**: primary aim is to achieve an overall improvement in system performance at a reasonable cost.

b) **Scalability and flexibility**: the distributed system in which the algorithm is implemented may change in size or topology. So the algorithm must be scalable and flexible enough to allow such changes to be handled easily.

c) **Priority**: prioritization of the resources or jobs need to be done on before hand through the algorithm itself for better service to the important or high prioritized jobs in spite of equal service provision for all the jobs regardless of their origin.

Following load balancing algorithms are currently prevalent in clouds:-

OLB : Opportunistic Load Balancing (OLB) assigns each task, in arbitrary order, to the next machine that is expected to be available, regardless of the task's expected execution time on that machine. The intuition behind OLB is to keep all machines as busy as possible [8, 9].

MET : In contrast to OLB, Minimum Execution Time (MET) assigns each task, in arbitrary order, to the machine with the best expected execution time for that task, regardless of that machine's availability. The motivation behind MET is to give each task to its best machine. This can cause a severe load imbalance across machines.

MCT : Minimum Completion Time (MCT) assigns each task, in arbitrary order, to the machine with the minimum expected completion time for that task. This causes some tasks to be assigned to machines that do not have the minimum execution time for them [1].

Min-min : Min-min heuristic uses minimum completion time (MCT) as a metric, meaning that the task which can be completed the earliest is given priority. This heuristic begins with the set U of all unmapped tasks. Then the set of minimum completion times (M), is found.

Max-Min : The Max-min heuristic is very similar to Min-min and its metric is MCT too. It begins with the set U of all unmapped tasks. Then, the set of minimum completion times (M) is found as mentioned in previous section. Next, the task with the overall maximum completion time from M is selected and assigned to the corresponding machine and the workload of the selected machine will be updated. And finally the newly mapped task is removed from U and the process repeats until all tasks are mapped [1, 9].

LJFR-SJFR : LJFR-SJFR heuristic begins with the set U of all unmapped tasks. Then the set of minimum completion times is found the same as Min-min. Next, the task with the overall minimum completion time from M is considered as the shortest job in the fastest resource (SJFR). Also the task with the overall maximum completion time from M is considered as the longest job in the fastest resource (LJFR). At the beginning, this method assigns the m longest tasks to the m available fastest resources (LJFR). Then this method assigns the shortest task to the fastest resource, and the longest task to the fastest resource alternatively [4, 11].

Sufferage : In this heuristic for each task, the minimum and second minimum completion time are found in the first step. The difference between these two values is defined as the sufferage value. In the second step, the task with the maximum sufferage value is assigned to the corresponding machine with minimum completion time [4, 12].

ISSN 2277-3061

PROBLEM DESCRIPTION

Although cloud computing has been widely adopted. Research in cloud computing is still in its early stages, and some scientific challenges remain unsolved by the scientific community, particularly load balancing challenges [15].

Some load balancers provide a mechanism for doing something special in the event that all backend servers are unavailable. This might include forwarding to a backup load balancer, or displaying a message regarding the outage. Load balancing [21] gives the IT team a chance to achieve a significantly higher fault tolerance. It can automatically provide the amount of capacity needed to respond to any increase or decrease of application traffic.

It is also important that the load balancer itself does not become a single point of failure. Usually load balancers are implemented in high-availability pairs which may also replicate session persistence data if required by the specific application. Load balancing is dividing the amount of work that a computer has to do between two or more computers so that more work gets done in the same amount of time and, in general, all users get served faster. Load balancing can be implemented with hardware, software, or a combination of both. Typically, load balancing is the main reason for computer server clustering.

• **Automated service provisioning:** A key feature of cloud computing is elasticity, resources can be allocated or released automatically. How then can we use or release the resources of the cloud, by keeping the same performance as traditional systems and using optimal resources?

• **Virtual Machines Migration**: With virtualization, an entire machine can be seen as a file or set of files, to unload a physical machine heavily loaded, it is possible to move a virtual machine between physical machines.

• In the paper titled “Improved Max-Min Algorithm in cloud computing“, the author is trying out to allocate the task with maximum execution time to the resource with minimum completion time.

• In this approach, if we are having more no of tasks(lets say 10,000), then the average turn-around time of the tasks will be very high which will decrease the efficiency of the entire system.

• And if the average turnarounds time will be high then the processing cost as well as waiting time will also be increased.

• Thus Load balancing is improving the performance by balancing the load among the resources like network links, CPU, disk and even on cloud and other storage devices.

OBJECTIVES

Users could experience many problems without Load balancing like delays, timeouts and long system responses.

A. LOAD BALANCING CLASSIFICATION:-

This is mainly divided into two categories: static load balancing algorithm and dynamic load balancing algorithm:

1) **Static approach**: - This approach is mainly defined in the design or implementation of the system. Static load balancing algorithms divide the traffic equivalently between all servers.

2) **Dynamic approach**: - This approach considered only the current state of the system during load balancing decisions. Dynamic approach is more suitable for widely distributed systems such as cloud computing .

Dynamic load balancing can be divided in two types as distributed approach and non-distributed (centralized) Approach. It is defined as following:

a) Centralized approach: - In centralized approach, only a single node is responsible for managing and distribution within the whole system. Other all nodes are not responsible for this.

b) Distributed approach: - In distributed approach, each node independently builds its own load vector .They collecting the load information of other nodes. All decisions are made locally using local load vectors. Distributed approach is more suitable for widely distributed systems such as cloud computing.

In Max-Min algorithm, in cloud computing describes the solving of large tasks first and delay in small tasks. So the main objective is to improve the Max-Min Algorithm in cloud computing. Max-Min strategy resolves the priority system and selects the task with the maximum completion time and assigns it to the resource on which achieve minimum execution time.

• To improve execution time over the completion time of the task.

• To improve the Turn Around Time.

• Supplying high performance computing based on protocols which allow shared computation and storage over long distances.

METHODOLOGY

Cloud services provide computing on demand in real time. Number of users accessing cloud environment are always more than that were using it on previous day. Cloud has application areas for developing applications, providing and managing infrastructure, patching applications. Users and their requests for accessing cloud infrastructure are highly

dynamic and loading servers running in data center. We need efficient strategy to balance load on these servers so that the servers don't get crash and they can persist long. Precisely Objective is to achieve accuracy, performance of servers and the cloud environment can be maintained.

Steps:

1. Initialize the Cloud Sim in Java
2. Create the Datacenter with different number of hosts.
3. Each Host will have the different numbers of Virtual machines of different capacities.
4. Then we will create the Cloudlets of varying length and size.
5. The list containing the Virtual machines [18] and Cloudlets will be given to the Data Center Broker (DCB)
6. DCB will count the estimated finish of the cloudlet on each and every virtual machine
7. Check the Status of the virtual machine whether it is busy or idle.
8. Fetch that virtual machine which is having more power than another virtual machines and is under loaded.
9. Dispatch our cloudlet to that virtual machine and we will modify the rating of that particular virtual machine
10. Repeat the same procedure for all the cloudlets.

CLOUD SIM

The CloudSim simulation layer provides support for modeling and simulation of virtualized Cloud-based data center environments including dedicated management interfaces for VMs, memory, storage, and bandwidth. The fundamental issues, such as provisioning of hosts to VMs, managing application execution, and monitoring dynamic system state, are handled by this layer. A Cloud provider, who wants to study the efficiency of different policies in allocating its hosts to VMs (VM provisioning), would need to implement his strategies at this layer. Such implementation can be done by programmatically extending the core VM provisioning functionality. There is a clear distinction at this layer related to provisioning of hosts to VMs. A Cloud host can be concurrently allocated to a set of VMs that execute applications based on SaaS provider's defined QoS levels. This layer also exposes the functionalities that a Cloud application developer can extend to perform complex workload profiling and application performance study. The top-most layer in the CloudSim stack is the User Code that exposes basic entities 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. By extending the basic entities given at this layer, a Cloud application developer can perform the following activities: (i) generate a mix of workload request distributions, application configurations; (ii) model Cloud availability scenarios and perform robust tests based on the custom configurations; and (iii) implement custom application provisioning techniques for clouds and their federation.

CONCLUSION

This paper is based on cloud computing technology which has a very vast potential and is still unexplored. The capabilities of cloud computing are endless. Cloud computing provides everything to the user as a service which includes platform as a service, application as a service, infrastructure as a service.
One of the major issues of cloud computing is load balancing because overloading of a system may lead to poor performance which can make the technology unsuccessful. So there is always a requirement of efficient load balancing algorithm for efficient utilization of resources. Our paper focuses on the various load balancing algorithms and their applicability in cloud computing environment.

REFERENCES

[1] O.M. Elzeki . "Improved Max-Min Algorithm in Cloud Computing". International Journal of Computer Applications(0975-8887) Volume 50-No.12,july 2012.

[2] "A technical support seminar on cloud computing technology" by Prashant Gupta.

[3] Amandeep Kaur Sidhu. "Analysis of load balancing techniques in cloud computing". International Journal of computers & technology volume 4 No. 2, March-April, 2013, ISSN 2277-3061.

[4] Ektemal Al-Rayis. "Performance Analysis of load balancing Architectures in Cloud computing" 2013 European Modeling Symposium. 978-1-4799-2578-0/13$31.00@2013 IEEE.

[5] Haozheng Ren. "The load balancing Algorithm in cloud computing Environment" 2nd International Conference on computer science and network technology 2012.

[6] Tushar Desai. "A survey of various load balancing techniques and challenges in cloud computing" International Journals of scientific and technology research volume 2. Issue11,Nov2013.

[7] Upendra Bhoi. “Enhanced max-min Task scheduling Algorithm in cloud computing”. International Journal of Application or Innovation in Engineering & management(IJAIEM), April 2013.

[8] Klaithem Al Nuaimi, “A survey of load balancing in cloud computing challenges and algorithm”. 2012 IEEE second symposium on network cloud computing and applications.

[9] Gytis Vilutis, “Model of load balancing and scheduling in cloud computing”. Proceedings of the ITI 2012 34th Int.Conf. on Information Technology Interfaces, June 25-28,Cavat,Croatia.

[10] S. Banerjee, I. Mukherjee and P.K. Mahanti, Cloud Computing Initiative using Modified Ant Colony Framework, World Academy of Science and Technology, 56, pp. 221-224, 2009.

[11] Y. Li, A Bio-inspired Adaptive Job Scheduling Mechanism on a Computational Grid, International Journal of Computer Science and Network Security, 6(3B), pp. 1-7, 2006.

[12] S.C. Wang, K.Q. Yan, W.P. Liao and S.S. Wang, Towards a Load Balancing in a Three-level Cloud Computing Network, Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology, pp. 108-113, 2010.

[13] M. Salehi and H. Deldari, Grid Load Balancing using an Echo System of Intelligent Ants, Proceedings of the 24th IASTED International Conference on Parallel and Distributed Computing and Networks, pp. 47-52, 2006.

[14] M. Dorigo, V. Maniezzo and A. Colorni, Ant System: Optimization by a Colony of Cooperating Agents, IEEE Transactions on Systems, Man, and Cybernetics, PP. 29-41, 1996.

[15] C.W. Chiang, Y.C. Lee, C.N. Lee and T.Y. Chou, Ant Colony Optimization for Task Matching and Scheduling, IEE Proceedings on Computers and Digital Techniques, 153 (6), pp. 373- 380, 2006.

[16] M. Dorigo, M. Birattari and T. Stutzle, Ant Colony Optimization-Artificial Ants as a Computational Intelligence Technique, IEEE Computational Intelligence Magazine, pp. 1- 12. 2006.

[17] J. Sun, S. Xiong and F.M. Guo, A New Pheromone Updating Strategy In Ant Colony Optimization, Proceedings of the International Conference on Machine Learning and Cybernetics, pp. 620-625, 2004.

[18] K.u Ruhana, K. Mahamud, H. Jamal and A. Nasir, Ant Colony Algorithm for Job Scheduling in Grid Computing, Proceedings of the Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, pp. 40-45, 2010.

[19] H. Jamal, A. Nasir, K. Ruhana, K. Mahamud and A.M. Din, Load Balancing Using Enhanced Ant Algorithm in Grid Computing, Proceedings of the Second International Conference on Computational Intelligence, Modelling and Simulation, pp. 160-165, 2010.

[20] C. Gong, J. Liu, Q. Zhang, H. Chen and Z. Gong, The Characteristics of Cloud Computing, Proceedings of the 39th International Conference on Parallel Processing Workshops, pp. 275-279, 2010.

[21] http://searchcloudcomputing.techtarget.com