Various Load Balancing Techniques in Cloud Computing

#Rashmikant R. Chaudhari, *Upendra R. Bhoi
Student, Assistant Professor
Parul Institute of Technology, Vadodara.Gujarat,India

***Abstract*- Cloud computing is arising as a new model of large - scale distributed computing. There are different number of issues which can be researched out for the right allocation and higher utilization of the resources using scheduling. The biggest drawback of it is load unbalanced, which is one of the middlemost issues for cloud providers. In cloud computing, load balancing is needed to distribute the dynamic local workload evenly across all the nodes. A few existing scheduling algorithms can maintain load balancing and provide higher strategies through efficient task scheduling and resource allocation techniques as well. In order to gain maximum benefits with optimized load balancing algorithms, it is needed to utilize resources efficiently. This paper discusses some of various existing load balancing algorithms in cloud computing and also their challenges.**

***Keywords* - Cloud Computing; Load Balance; Makespan; Cloud Task Scheduling.**

I. INTRODUCTION

The cloud computing is a large group of interconnected computers and cloud symbol represents a group of systems or complicated networks. Cloud computing is one way of communication among the various system in the network with the help of internet [1].

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 [4].

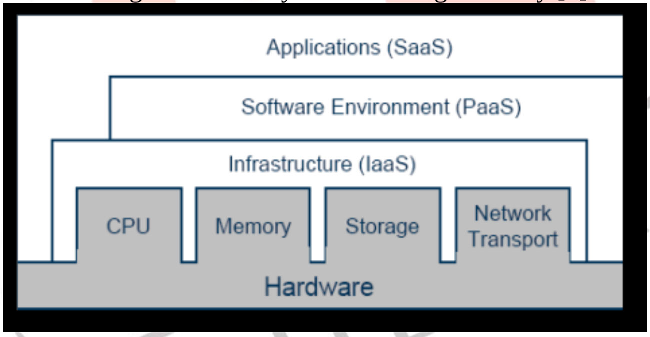

Fig. 1 Cloud computing over view [9]

Cloud Computing refers three services as Iaas, Paas, Saas. Infrastructure as a service refers to the sharing of hardware resources for executing services, typically using virtualization technology. With this so-called Infrastructure as a Service (IaaS) approach, potentially multiple users use existing resources. The resources can easily be scaled up when demand increases, and are typically charged for on a per-pay-use basis. In the Platform as a Service (PaaS) approach, the offering also includes a software execution environment, such as an application server. In the Software as a Service approach (SaaS), complete applications are hosted. on the hat e.g. your word processing software isn't installed locally on your PC anymore but runs on a server in the network and is accessed through a web browser [6].

Cloud task scheduling is an NP-complete problem in general . In the typical cloud scenario, cloud users submit their tasks to cloud scheduler. The Cloud scheduler firstly queries the Cloud Information Service for the availability of resources and to know their properties, and then scheduling the tasks on the resources that match tasks' requirements. After execution of tasks, the results are sent back to the users. How to schedule tasks in such cloud environment efficiently is a new challenge because of its nature of high heterogeneity in operating systems, architecture, resource providers and resource consumers [4].

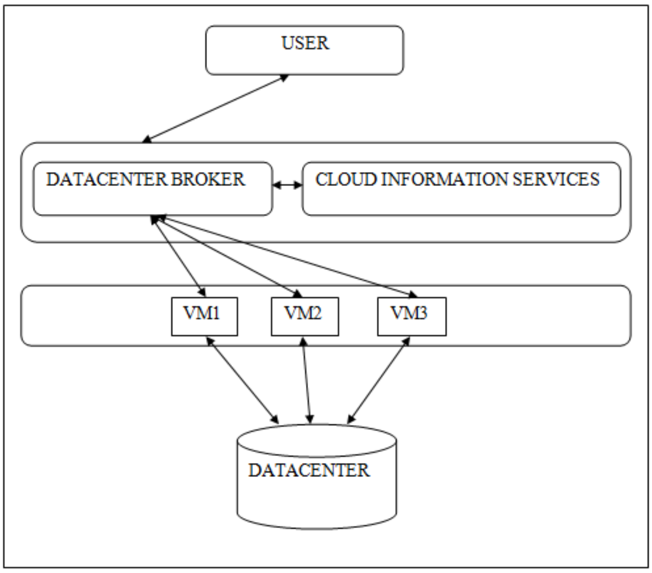

Fig. 2 Scheduling in Cloud [1]

Scheduling process in cloud can be generalized into three stages namely–

- **Resource Discovering And Filtering** – Datacenter Broker discovers the resources present in the network system and collects status information related to them.
- **Resource Selection** – Target resource is selected based on certain parameters of task and resource. This is deciding stage.
- **Task Submission** -Task is submitted to resource selected [14].

The simplified scheduling steps mentioned above are shown in Figure 2.

II. GUIDELINES OF LOAD BALANCING

Load Balancing is a computer networking method to distribute workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources, to achieve optimal resource utilization, maximize throughput, minimize response time, and avoid overload. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy. The load balancing service is usually provided by dedicated software or hardware, such as a multilayer switch or a Domain Name System server. Load balancing is one of the central issues in cloud computing [10]. It is a mechanism that distributes the dynamic local workload evenly across all the nodes in the whole cloud to avoid a situation where some nodes are heavily loaded while others are idle or doing little work. It helps to achieve a high user satisfaction and resource utilization ratio, hence improving the overall performance and resource utility of the system. It also ensures that every computing resource is distributed efficiently and fairly [11].

The goal of load balancing is improving the performance by balancing the load among these various resources (network links, central processing units, disk drives.) to achieve optimal resource utilization, maximum throughput, maximum response time, and avoiding overload. To distribute load on different systems, different load balancing algorithms are used.

III. 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 [14].
- **Resource Utilization** is used to check the utilization of resources. It should be optimized for an efficient load balancing [14].
- **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 [14].
- **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 [14].
- **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 inter process communication. This should be minimized so that a load balancing technique can work efficiently [14].

The objective and motivation of this survey is to give a systematic review of existing load balancing algorithms in cloud computing and encourage the amateur researcher in this field, so that they can contribute in developing more efficient load balancing algorithm. This will benefit interested researchers to carry out further work in this research area.

IV. VARIOUS LOAD BALANCING TECHNIQUES IN CLOUD

Following Task scheduling techniques are currently prevalent in clouds.

1) ***Min-Min Algorithm*****:** T. D. Braun, H. Jay Siegel, N. Beck, L. L. Boloni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, and B. Yao [3] proposed the Min-Min algorithm is simple and still basis of present cloud scheduling algorithm. It starts with a set of all unmapped tasks. Then the resource which has the minimum completion time for all tasks is found. Next, the task with the minimum size is selected and assigned to the corresponding resource (hence the name Min-Min). Finally, the task is removed from set of all unmapped tasks and the same procedure is repeated by Min-Min until all tasks are mapped.

The expectation is that a smaller makespan can be obtained if more tasks are assigned to the machines that complete them the earliest and also execute them the fastest Min-min algorithm has running time complexity of O(mn^2), where m is the number of resources currently in the system and n is the number of submitted tasks which should be scheduled [2].

2) ***Max-Min Algorithm*****:** M. Maheswaran, Sh. Ali, H. Jay Siegel, D. Hensgen, and R. F. Freund [7] proposed the Max-min algorithm is commonly used in distributed environment which begins with a set of unmapped tasks. Then calculate the expected execution matrix and expected completion time of each task on the available resources. Next, choose the task with overall maximum expected completion time and assign it to the resource with minimum overall execution time. Finally, the task is removed from set of all unmapped tasks and the same procedure is repeated by Max-Min until all tasks are mapped. [6].

Max-min algorithm has running time complexity of O(mn^2), where m is the number of resources currently in the system and n is the number of submitted tasks which should be scheduled [2].

3) ***Resource Aware Scheduling Algorithm*****:** Saeed Parsa and Reza Entezari-Maleki [8] proposed a new task scheduling algorithm RASA. It is composed of two traditional scheduling algorithm; Min-min and Max-min. RASA uses the advantages of Min-min and Max-min algorithms and covers their disadvantages. Though the deadline of each task, arriving rate of the tasks, cost of the task execution on each of the resource, cost of the communication are not considered.

Based on experimental results, if the number of available resources in grid system is odd it is highly preferred to start by Min-min in first round otherwise recommended starting by Max-min algorithm [12]. For next rounds just assign resources to task using a strategy different from last round ignoring waiting time of the small tasks in Max-min algorithm and the waiting time of the tasks in Min-min algorithm.

RASA has no time consuming; it has time complexity like Max-min and Min-min, O (mn^2) where is the total number of resources and n is the number of tasks.

4) ***Load Balanced Min-Min Algorithm*****:** T Kokilavani , GA DI [5] proposed load balancing algorithm. Its aims to increase the utilization of resources with light load or idle resources thereby freeing the resources with heavy load. The algorithm tries to distribute the load among all the available resources. At the same time, it aims to minimize the makespan with the effect utilization of resources.

The LBMM algorithm is start by executing Min-Min algorithm at the first step. At the second step it chooses the smallest size task from the most heavy load resource and calculates the completion time for that task on all other resources. Then the maximum completion time of that task is compared with the makespan produced by Min-Min. If it is less than makespan then the task is reassigned to the resource that produce it, and the ready time of both resources are updated. The process repeats until no other resources can produce less completion time for the smallest task on the heavy load resource than the makespan. Thus the heavy load resources are freed and the light load or idle resources are more utilized.

5) ***Load Balanced Improved Min-Min Algorithm*****:** Huankai Chen, Professor Frank Wang,Dr Na Helian,Gbola Akanmu [13], proposed LBIMM algorithm is start by executing Min-Min algorithm at the first step. At the second step it chooses the smallest size task from the most heavy load resource and calculates the completion time for that task on all other resources. Then the minimum completion time of that task is compared with the makespan produced by Min-Min. If it is less than makespan then the task is reassigned to the resource that produce it, and the ready time of both resources are updated. The process repeats until no other resources can produce less completion time for the smallest task on the heavy load resource than the makespan. Thus the heavy load resources are freed and the light load or idle resources are more utilized.

TABLE I .COMPARISON OF LOAD BALANCING ALGORITHM BASED ON DIFFERENT FACTORS

Algorithm			Parameters						Recommended in
Name	Method	Factor	Response Time	Resource Utilization	Scalability		Performance	Overhead	
					Dynamic	Static			
Min-min Algorithm	Batch Mode	Meta tasks	√	√	×	√	√	√	Grid
Max-min Algorithm		Meta tasks	√	√	×	√	√	√	Grid
Resource Aware Scheduling Algorithm		Meta tasks	√	√	×	√	√	×	Grid
Load Balanced Min-min Algorithm		List of heavy load resources	×	√	×	√	√	×	Grid
Load Balanced improved Min-min		List of heavy load resources	×	√	×	√	√	×	Cloud

Algorithm									

CONCLUSION

Load balancing is one of the main issues in cloud computing and resource utilization percentage by making sure that every computing resource is distributed efficiently and fairly. In this paper, I have surveyed the various existing Load balancing algorithm in cloud computing and tabulated their various parameters such as Cost, Makespan, Resource utilization, Scalability and so on. It is required to distribute the dynamic local workload evenly across all the nodes to achieve a more user satisfaction.

ACKNOWLEDGEMENT

Rashmikant.R.Chaudhari would like to thank to my thesis guide Asst.Prof. Upendra.R.Bhoi for his great effort and instructive comments in this paper work. Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of this paper.

REFERENCES

[1] Monika Choudhary and Sateesh Kumar Peddoju “A Dynamic Optimization Algorithm for Task Scheduling in Cloud Environment” International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 Vol. 2, Issue 3, May-Jun 2012, pp.2564-2568.

[2] R. F. Freund, M. Gherrity, S. Ambrosius, M. Campbell, M. Halderman, D. Hensgen, E. Keith, T. Kidd, M. Kussow, J. D. Lima, F. Mirabile, L. Moore, B. Rust and H. J. Siegel, "Scheduling Resource in Multi-User, Heterogeneous, Computing Environment with SmartNet,"In the Proceeding of the Seventh Heterogeneous Computing Workshop, 1998.

[3] T. D. Braun, H. Jay Siegel, N. Beck, L. L. Boloni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, and B. Yao, "A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems, "Journal of Parallel and Distributed Computing, Vol. 61, pp. 810-837, 2001.

[4] V. Venkatesa Kumar and K. Dinesh “Job Scheduling Using Fuzzy Neural Network Algorithm in Cloud Environment” Bonfring International Journal of Man Machine Interface, Vol. 2, No. 1,March 2012.

[5] T. Kokilavani and Dr. D. I. George Amalarethinam, "Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing", International Journal of Computer Applications, Vol. 20, No. 2, pp. 43-49, 2011.

[6] Yu, Xiaogao, and Xiaopeng Yu. "A new grid computation-based Min-Min algorithm", Fuzzy Systems and Knowledge Discovery, FSKD'09 Sixth International Conference on, Volume 1, Pages 43 – 45, IEEE, 2009

[7] M. Maheswaran, Sh. Ali, H. Jay Siegel, D. Hensgen, and R. F. Freund, "Dynamic Mapping of a Class of Independent Tasks onto Heterogeneous Computing Systems, Journal of Parallel and Distributed Computing, Vol. 59, pp. 107-131, 1999.

[8] Saeed Parsa and Reza Entezari-Maleki , "RASA: A New Grid Task Scheduling Algorithm", International Journal of Digital Content Technology and its Applications,Vol. 3, pp. 91-99, 2009.

[9] Wu, W. Shu and H. Zhang, Segmented Min-Min: A Static Mapping Algorithm for Meta-Tasks on Heterogeneous Computing Systems, in Proc. of the 9th Heterogeneous Computing Workshop (HCW'00), pp. 375--385, Cancun, Mexico, May 2000.

[10] Preeti Agrawal, Yogesh Rathore,"Resource Management In Cloud Computing With Increasing Dataset ",International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, Volume 2, Issue 6, June 2012).

[11] B. P. Rima, E. Choi, and I. Lumb, “A Taxonomy and Survey of Cloud Computing Systems”, Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Korea, August 2009, pages 44-51.

[12] A. M. Alakeel, “A Guide to dynamic Load balancing in Distributed Computer Systems”, International Journal of Computer Science and Network Security (IJCSNS), Vol. 10, No. 6, June 2010, pages 153-160.

[13] Huankai Chen,Professor Frank Wang,Dr Na Helian,Gbola Akanmu,"User-Priority Guided Min-Min Scheduling Algorithm For Load Balancing in Cloud Computing", IEEE 2013.

[14] Amandeep Kaur Sidhu, Supriya Kinger,"Analysis of Load Balancing Techniques in Cloud Computing",International Journal of Computers & Technology Volume 4 No. 2, March