ISSN 2395-1621

A Survey on Load Balancing in Multi-Cloud Environment

[#1]Karishma Gidge, [#2]Prof. S.B. Rathod

[1]karishmagidge@gmail.com
[2]sbrathod.sae@sinhgad.edu

[#12]Department of Computer Engineering,

SAOE, Pune,
Savitribai Phule Pune University

ABSTRACT

Multi-Cloud computing has many attractive features like continuous availability, scalability, simplicity, low cost etc. It helps the user to access the ubiquitous data by using pay-as-you-go cost model. User is charged on virtual machine hours. So the effective use of resources will result in maximum throughput of system and reduce the cost of user on a large scale. Load balancing plays a vital role as it ensures equal load on all the Virtual Machines. There are many algorithms developed to ensure load balancing in multi-cloud computing environment. To improve the global throughput of cloud resources, effective and efficient load balancing algorithm is necessary. In this paper we have presented a survey on various load balancing algorithms for the ease of understanding we have grouped the algorithms.

ARTICLE INFO

Article History

Received: 20th December 2016

Received in revised form :

20th December 2016

Accepted: 25th December 2016

Published online :

26th December 2016

Keywords— cloud computing, load balancing, multi-cloud, dynamic algorithms

I. INTRODUCTION

According to National Institute of Standards and technology (NIST) cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can rapidly provisioned and released with minimal management effort or service provider interaction. Since its inception it is widely being used in each and every field like online data storage, social media, inline gaming etc. It is sharing of resources rather than having a personal one. It provides fast and easy access to files once stored on server. When an application is hosted on cloud or a file is stored on it can be accessed from all around the globe and from any device connected to internet, and hence it reduces the cost of purchasing the hardware and maintenance cost of infrastructure. Services provided by cloud computing are Software as a Service example Gmail, Platform as a Service example AZURE and Infrastructure as a Service example AWS [2].

Some of the challenges in cloud computing are resource allocation and task scheduling [1]. If the resources of cloud computing are utilized efficiently they can benefit us on a large scale. Load balancing is major issue in cloud computing. It has a major impact on cloud computing. Load balancing is distribution of loads equally among available Virtual Machine(VM) [3] and is generally applied on huge data. Load can be memory, CPU capacity, network usage. It is necessary because all nodes in network have different processing capacities. When a file is present on server and many users across globe try to access the file at a same time, this results in conjunction at server due to large number of
requests resulting into a major issue [5]. This results in slowing down of server. Besides, in cloud computing services are provided to user based on pay-as-you-go cost model. So an efficient use of resource will reduce the time for which the resource is being utilized and hence can save a lot of cost [1]. To overcome these issues there are various algorithms developed which make the efficient use of resources and increase the throughput. The algorithm has the job to assign each task to node and efficient processing of requests taking deadline into consideration [4].

This paper discusses the different types of load balancing algorithms in multi-cloud computing environment. They are broadly classified into Static and Dynamic. Static are used for systems with homogeneous environments. Dynamic are used for systems which take runtime state of system into consideration.

We provide a problem formulation for the security-aware virtual server migration with an objective to minimize the migration cost, and prove the proposed problem is NP-hard by reducing the well-known Bin-Packing problem to a special case of our problem;

We propose a greedy algorithm that can provide an approximate solution to minimize the migration cost and guarantee the security needs for a single enterprise migration problem, and prove its approximate factor;

We also develop a genetic approach to a realistic migration problem in the context of multiple enterprises;

We perform an extensive simulation study to evaluate the performance of the proposed solution under various settings.

Our simulation results demonstrate that our algorithms out-performs the classic random assignment algorithm

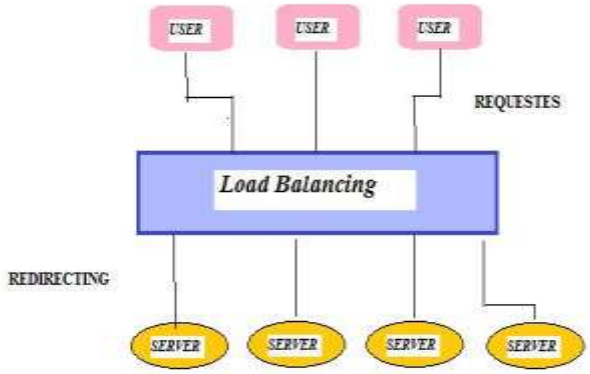

Fig. 1 shows the significance of load balancing in cloud computing environment. The way in which users request and how it is assigned to VM.

II. LITERATURE SURVEY

A. Static load balancing Algorithms

These are traditional type of load balancing algorithms. They are less intricate and are easy to implement. They are to be applied on system having low variation in loads. It simply divides the load among available VMs. It doesn't take into consideration the present state of VM while allocating the tasks to them. It requires the prior knowledge of the system resources. Processing power, memory and storage capacity and most recent communication performance are considered at beginning. They don't adapt to dynamic changes [4]. At the beginning based on processing power servers are assigned weights. Consequently, process with the higher weight receives more number of tasks for execution. After the execution begins nodes do not take note of dynamic changes that occur at runtime. One of the major drawbacks of these algorithms in tasks cannot be shifted from one node to another for load balancing.

1) Round Robin Load Balancing Algorithm.

Round Robin Load Balancing Algorithm uses the popular Round Robin technique for allocating jobs to servers. It is a technique of choosing all the elements equally and giving equal turns to each element. Its main motive is fairness and every process gets equal time slice for execution. The first processes to be allocated for execution is selected randomly and remaining all others are selected in round robin fashion. Here, priority of the tasks are not taken into consideration [11]. Data central controller assigns the requests to a list of VM's in Round Robin fashion. Advantage of this algorithm is all processes are put into execution and there is no starvation. As there is no process done in order to select VM for execution it is fast and easy to implement [11]. There are various limitations as each VM has different processing capability and hence results in some of them being idle. This has been overcome by weighted Round Robin Algorithm which assigns weights to each VM based on its processing capability. Besides, large processes need more time hence as a result small processes need to wait for more time. In spite of all these advantages this algorithm is not suited for load balancing in cloud computing environment.

2) Min-Min

Min-Min is simplest among all the algorithms. It is more efficient because it selects best task for execution which has least execution time and puts it into execution which results in greater speed [5]. For an instance there is a queue 'X' unallocated tasks, it finds the task with least execution time. Let us assume the minimum time for execution is 'T'. So it is taken from 'X' and is put into execution on a particular resource. This process is carried out till 'X' becomes empty.

Multitasking is carried out here. Here the smallest task is given the highest priority and so it is efficient for system having large number of small tasks. In case of system having more number of large tasks this algorithm is not efficient because it puts them into waiting state. Limitation of this is it does not consider the current weight on system while assigning the tasks to them. So proper load balancing is not achieved [10].

3) Max-Min

Max-Min is analogous to Min-Min Load Balancing Algorithm. In this all the tasks are arranged in same way as in Min-Min algorithm only difference is the tasks which has maximum execution time is given highest priority and is executed in beginning. Remaining tasks are arranged in similar way from maximum execution time to

minimum execution time [2]. This algorithm requires more time when compared with others.

4) Deadline Based Pre-emptive Scheduling (DBPS)

Deadline is time before which the process should be executed, if it is executed after the allotted time it is of no significance. Earliest Deadline First (EDF) is used to place the processes in the priority queue. When a process which is under execution is executed, the priority queue will search for the process having the earliest deadline. DBPS is performed every time when new task is to be executed. If while execution of a task there is an interrupt by other task then the task manager will compare the deadline of the tasks and assign a priority accordingly. If a soft real time process is under execution and hard real time process interrupts then it is pre-emptied and put into waiting state and hard real time process will be executed. Thus, DBPS gives highest priority to the task with earliest deadline [1].

B. Dynamic load balancing Algorithms

Dynamic algorithms make the decisions corresponding to the current state of the system [6].They take into account the prior information as well as current state of the system [3]. It takes into consideration the capabilities of the system like processing power, memory and storage capacity as well as network bandwidth into consideration. Implementation of these algorithms is quite intricate as it is quite tough to keep the updates of the current system at every point of time, but it is very easy to track down load on any server at the data centre. Main advantage here is tasks can easily be migrated from one server to another while they are being executed, for load balancing and thus they manage it in a very intelligent way among all the available servers [2].It is more accurate and efficient than static load balancing algorithms.

1) Ant Colony Algorithm

Ant Colony Algorithm is analogous to the real moments of ants, as they find they the best shortest path towards the food. As they travel they release a chemical substance known as pheromone which helps other ants to find the optimal path towards the food. Same concept in used in this algorithm. It has a table known Pheromone table which is used to keep track of all the ants, threshold of each node and its CPU utilization. A threshold value of CPU utilization is set and then ants move in forward direction. Accordingly each node is classified as overloaded or under loaded based on its CPU utilization. Further ants move in backward direction and checks for under loaded nodes. If it finds one then load is transferred from overloaded node to an under loaded node. This process is continued till all the nodes in network become balanced. Each time the migration takes place it is reported to pheromone table. This is an iterative process till all the nodes in network become balanced [7].

2) Throttled load balancing

This algorithm finds suitable VMs for execution of tasks.VM's are grouped according to requests they can manage [8]. There are two controllers namely, main controller and node controller. The node maintains the information of each partition system and updates it to the main controller. When a main controller receives the task it assigns the task to the suitable node controller which assigns the task to a suitable VM based on capabilities like processing power, memory etc. Node controller first checks the resources and then assigns tasks to them [1]. If no machines are idle then tasks need to wait and are queued. This algorithm is has a very high performance but task starvation is main drawback here, as well searching for node causes delay.

3) Neighbour Aware Random Sampling (NARS)

This algorithm is used to improve the selection process of nodes for random walk around the network. The number of nodes are approximated so nodes can leave or join this process randomly. There are certain assumptions made in this algorithm that every node is aware about the other node in network, duration for which resource is going to be utilized is known, and each node is aware about the computational capacity of other nodes.

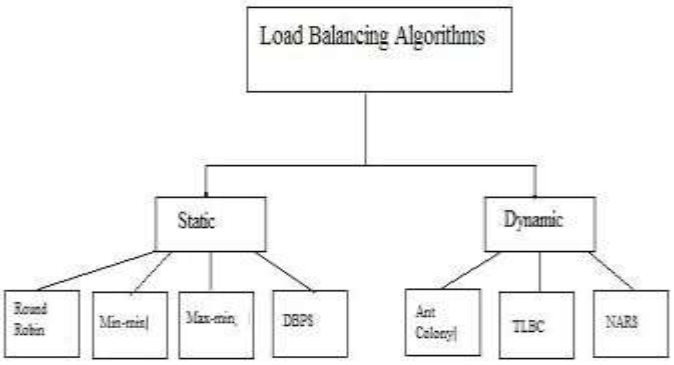

Fig. 2 Classification of load balancing algorithms

III. CONCLUSION

As we have seen above load balancing is paramount in multi-cloud computing. In this paper various algorithms have been discussed which can be used to increase the efficiency. As users are charged on pay as you go basis their cost can be substantially reduced if time consumed by servers to process data is reduced. By considering all the algorithm and taking their limitations into consideration well suited algorithm for a particular use must be selected. A generalised better algorithm needs to be developed which will manage all tasks large as well as small efficiently by balancing load on the servers.

REFRENCES

[1] M. R. Sumalatha, C. Selvakumar, T. Priya, R. Titus Azariah And P. Murali Manohar "Clbc - Cost Effective Load Balanced Resource Allocation For Partitioned Cloud System", 2014 International Conference On Recent Trends In Information Technology.

[2] Sidra Aslam And Munam Ali Shah," Load Balancing Algorithms In Cloud Computing: A Survey Of Modern Techniques", 2015 National Software Engineering Conference (Nsec 2015).

[3] Tushar Desai And Jignesh Prajapati," A Survey Of Various Load Balancing Techniques And Challenges In Cloud Computing", International Journal Of Scientific & Technology Research Volume 2, Issue 11, November 2013 Pp 158-161.

[4] Klaithem Al Nuaimi, Nader Mohamed, Mariam Al Nuaimi And Jameela Al-Jaroodi," A Survey Of Load Balancing In Cloud Computing: Challenges And Algorithms", 2012 Ieee Second Symposium On Network Cloud Computing And Applications Pp 137-142.

[5] Divya Chaudhary And Bijendra Kumar, "Analytical Study Of Load Scheduling Algorithms In Cloud Computing", 2014 International Conference On Parallel, Distributed And Grid Computing, Pp 7-12.

[6] Nadeem Shah And Mohammed Farik ," Static Load Balancing Algorithms In Cloud Computing: Challenges & Solutions", International Journal Of Scientific & Technology Research Volume 4, Issue 10, October 2015 Pp 365-367.

[7] Shagufta Khan And Niresh Sharma, "Effective Scheduling Algorithm For Load Balancing (Salb) Using Ant Colony Optimization In Cloud Computing" , International Journal Of Advanced Research In Computer Science And Software Engineering, February 2014 Pp 966-973.

[8] Priyadarashini Adyasha Pattanaik, Sharmistha Roy And Prasant Kumar Pattanaik, "Performance Study Of Some Dynamic Load Balancing Algorithms In Cloud Computing", 2nd International Conference On Signal Processing And Integrated Networks (Spin), 2015,Pp 619-624.

[9] Ariharan V And Sheeja S Manakattu, "Neighbour Aware Random Sampling (Nars) Algorithm For Load Balancing In Cloud Computing", 2015 Ieee.

[10] Subhadra Bose Shaw And Dr. A.K. Singh, "A Survey On Scheduling And Load Balancing Techniques In Cloud Computing Environment", 5th International Conference On Computer And Communication Technology (Iccct), 2014, Pp 87-95.

[11] Nadeem Shah, Mohammed Farik, "Static Load Balancing Algorithms In Cloud Computing: Challenges & Solutions", International Journal Of Scientific & Technology Research Volume 4, Issue 10, October 2015, Pp 365-367.

