Review Article

A Survey Paper on Data Load Balancing in Cloud Computing

Ranjna Singh[1], Prof. Pratiksha Asati[2]

[1] M.Tech. Scholar, Department of CSE, Vindhya Institute of Technology and Science (VITS) Satna (M.P.), INDIA
[2]Assistant Professor, Department of CSE, Vindhya Institute of Technology and Science (VITS) Satna (M.P.), INDIA

ABSTRACT

Since it offers a flexible and simple way to store and retrieve data and files over the network, cloud computing has grown in popularity over the past several years. The largest challenge in the field of cloud computing is managing the billions of dynamically incoming requests from end users. Various load balancing strategies to disperse the load equally among the cloud nodes have been presented in recent years to handle such requests effectively and efficiently. The virtual machines are evenly distributed among the incoming jobs by the conventional throttled load balancing mechanism. However, these algorithms only perform effectively in situations with uniform Virtual Machines. We have suggested an approach with cluster-based load balancing that can function well in an environment of heterogeneous nodes to address this problem. To reduce the scanning cost, this technique clusters the devices.

KEYWORDS

Virtual Machines (VMS).

1. INTRODUCTION

A completely new technology, cloud computing is based on parallel, distributed, and grid computing. It is also an extension of virtualization and a combination of these three types of computing [1]. In a form of computer programme outsourcing known as cloud computing, users have access to software and apps from any location because the programmes are hosted by a third party and are stored in the cloud. In order to provide computing resources and services on demand, cloud computing centralises server resources on a scalable platform. This architecture is dispersed. Because of features including lower initial investment, ease of management, quicker deployment, independence from location, device independence, and security, business owners are drawn to the cloud computing concept [2].

Sharing processing power, storage space, bandwidth, memory, and software via the internet is simple with cloud computing [2]. A versatile and convenient method of storing and retrieving data and files is provided by cloud computing. Numerous approaches to improve and streamline processes as well as user performance are needed to make massive data sets and files accessible to a large number of users worldwide [3]. One of the most crucial concerns in cloud computing is load balancing. Good load balancing is required to increase the effectiveness of cloud computing and to increase user satisfaction[4]. To enhance resource usage, minimise reaction times, and ensure maximum user satisfaction, load balancing involves distributing incoming loads or tasks equally among cloud nodes [5]. To enhance resource usage and task response time while preventing a situation where some nodes are overloaded or others are underloaded, general system work load is redistributed among all nodes of the distributed system during load balancing [6]. Dynamic load balancing methods are more appropriate for unstable and homogeneous environments than static load balancing techniques. However, in dynamic and heterogeneous situations, dynamic algorithms yield higher outcomes [3].

The majority of standard load balancing algorithms do not take into account the heterogeneity of virtual machines or the resource-specific requirements of the jobs. In a cloud setting, this could cause an issue. This study established four primary policies, including transfer policy, selection policy, information policy, and placement policy, to address the issues with conventional load balancing algorithms.

In order to handle more user requests, virtualization—the act of building several virtual machine instances for a single host machine—is a key component of cloud computing. It is feasible to execute numerous operating systems and different applications simultaneously on the same server node by using virtual machines (VMs) [5].

The dynamic load balancing approach used in this research will support heterogeneous settings and take into account the task-specific resource requirements by utilising virtual machines with k-means clustering. To increase throughput, execution time, reaction time, and turnaround time, VMs have been clustered using the k-means clustering approach.

2. LITERATURE SURVEY

Some recent contributions to the field of cloud computing are included in this section. Numerous academics employed various load balancing algorithms to address issues in the field of cloud computing.

Research topics, difficulties, architecture, platforms, and applications in the field of cloud computing are presented by Santosh Kumar, R. H. Goudaret, et al. Additionally, cloud computing and grid computing are contrasted in this essay. Cloud computing is created to suit generic application needs, whereas grid is built to finish a specific purpose. They have discussed cloud computing concerns such user data privacy, server dependability, cloud interoperability, and compliance. Users of cloud computing are not free to physically possess the data storage; instead, cloud providers have the data storage and control. The issues in cloud computing include data security, costing and charging models, and service level agreements between providers and customers [1].

Researchers Klaithem Al Nuaimi, Nader Mohamed, and others have suggested a study on the difficulties with load balancing algorithms in cloud computing. This paper provides an overview of the most recent load balancing techniques created especially for cloud computing settings. Because of the speed of the network links between the nodes, the distance between the client and the nodes processing the task, and the distances between the nodes involved in providing the service, designing a load balancing algorithm that can work for spatially distributed nodes is a very difficult task. Complexity of the algorithm, point of failure, and replication are difficult problems to solve while building a load balancing algorithm. Static algorithms and dynamic algorithms are the two categories into which load balancing algorithms are divided. The duty in a static load balancing mechanism is given to the node that can handle fresh requests. Based on the attributes gathered and calculated, a dynamic algorithm assigns the jobs and may dynamically reassign them to the nodes. This study finds that the DDFTP (dual direction downloading method from FTP servers) technique is more effective in terms of storage use and better suited for cloud environments [3].

For the public cloud, GaochaoXu, Junjie Pang, Xiaodong Fu, et al. have looked into a load balancing paradigm based on cloud partitioning. The public cloud was introduced in this study, which consists of multiple nodes with distributed computing capabilities spread across a wide geographic area. This model creates several cloud partitions from the public cloud. This technique has increased efficiency in the public cloud environment by using game theory to the load balancing strategy [4].

Musical instrument identification: a pattern-recognition approach is represented by Joseph Doyle and Robert Short. This paper explains how huge public cloud infrastructure can use power produced by numerous power plants. For a given amount of energy produced, each power plant will emit a different amount of carbon dioxide. The service paradigm, according to the author, entails a cloud-based service provider (CBSP) that offers a sizable pool of computing and networking sources. The pool is used to distribute this to cloud users as needed. These assets are accessible to cloud users. In this study, the carbon intensity of power providers in various geographic areas is analysed to lessen the cloud's negative environmental effects. The cost of electricity, carbon emissions, and the typical response time are all examined by the author [7].

An effective distributed method for load balancing in cloud computing has been developed by AartiVig, Rajendra Singh Kushwahet al. The process of load balancing involves dividing the overall load among the system's multiple nodes for resource usage. This model has a number of local agents created by channel agents, and each local agent has information about it stored in a table. All of the local agents of the channel agent receive the pertinent neighbour information from the table. When the local agent cannot execute a process because there is no space available in the virtual machine, the local agent sends a mitigation agent to its neighbouring agents that includes the configuration of the virtual machine as well as the process id and local host id. When a neighbouring agent checks its table, it returns a true value for the virtual machine if it fits the configuration of the virtual machine that is provided, including the amount of available space. If not, the neighbouring agent passes the mitigation agent on to its neighbours. The mitigation agent determines if its local host is available when other agents receive If a mitigation agent is already working on it, the request is forwarded to the local agent for further consideration [8].Shridhar For load balancing in a cloud context, G. Domanal and G. Ram Mohana Reddy suggest an unique hybrid scheduling technique. The DCBT design process, which combines Divide-and-Conquer and throttled algorithms, was used to create this algorithm. It effectively distributes the incoming load to enhance resource usage and shorten job execution time in a cloud environment [9].

3. LOAD BALANCING ALGORITHMS WORKING PRINCIPLE

Cloud and computing are the two components that make up cloud computing. Cloud computing is nothing more than a collection of disparate services where resources like processing are mapped out. Load balancing is, as we all know, the main problem in the world of cloud computing. For load balancing, a number of strategies are available. Resource allocation and scheduling are done using load balancing [9][10]. The following types of load balancing strategies are categorised into:

Static load balancing (A).

In this type of cloud, nodes' capacity, processing speed, performance capabilities, etc., must be known beforehand.

Dynamic Load Balancing Is B.

The load in a dynamic environment can be modified while running. These are helpful in networks with heterogeneity.

- ❖ The use of centralised load balancing
- ❖ Information is updated and maintained by a single central node.
- ❖ Diffusion of load balancing
- ❖ In this case, numerous nodes are in charge of updating and maintaining the data.

- To properly balance the load, several nodes keep an eye on the network.
- Hierarchical load balancing, element
- Large homogeneous and heterogeneous networks use hierarchical load balancing.
- several cloud layers for load balancing.

There are numerous load balancing methods used in cloud computing already, such as CARTON. To raise the server's performance level, it employs two parameters called LB and DRL. Based on sampling, the Compare and Balance technique. In order to lower the cost of cloud services, it reduces the number of host machines. By mapping a task to a virtual machine and then the virtual machine to host resources, task scheduling based on the load balancing approach achieves good resource utilisation. A self-organizing and naturally inspired algorithm, the Honeybee Foraging Behavior Algorithm has improved performance by growing the system. System domain random sampling is used in the biassed random sampling approach. It succeeds in self-organization to balance the load throughout the entire system [3].

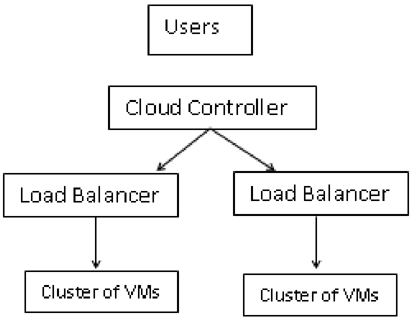
4. METHODOLOGY

The incoming jobs are evenly divided across the virtual machines (VMs) by the conventional Throttled load balancing technique. The main disadvantage is that while this approach works well in environments with homogeneous VMs, it incurs a cost every time a job needs to be completed by scanning the entire list of VMs. This system's objective is to distribute the load equitably by allocating tasks to suitable nodes. The node that has space to spare processes the input file; otherwise, it is moved to the other node. K-mean virtual machine clustering was used in its creation. A physical machine is turned into a virtual machine, which is then given to the consumer, giving him the impression that he is actually using the physical equipment [10][11].

Through the use of virtual machine clustering, this solution resolves the load balancing problem. The system's objective is as follows:

- I Distributing the load equitably by assigning the user's request to the appropriate nodes.
- ii) Reassigning VMs to handle the request

Figure 1 depicts the system design, which includes the users, cloud controller, and load balancer.



A. Users

Image 1: System Architecture

A user submits a cloud-based request to process a file, programme, etc.

The controller, B.

Processing of tasks is controlled by a cloud controller. It decides which load balancer or node controller should be chosen for the following allocation.

C. Clustering

This system divides VMs with comparable capacity into clusters using a clustering method. Using k-means clustering, it separates n virtual machines into k groups.

Load Balancing

A list of all the clusters and a list of VMs for each cluster will be kept on file by the load balancer. This list keeps track of each cluster's minimum and maximum resource-specific capacity. The scanning overhead for the complete list of virtual machines is reduced when they are divided up into a cluster [5][12].

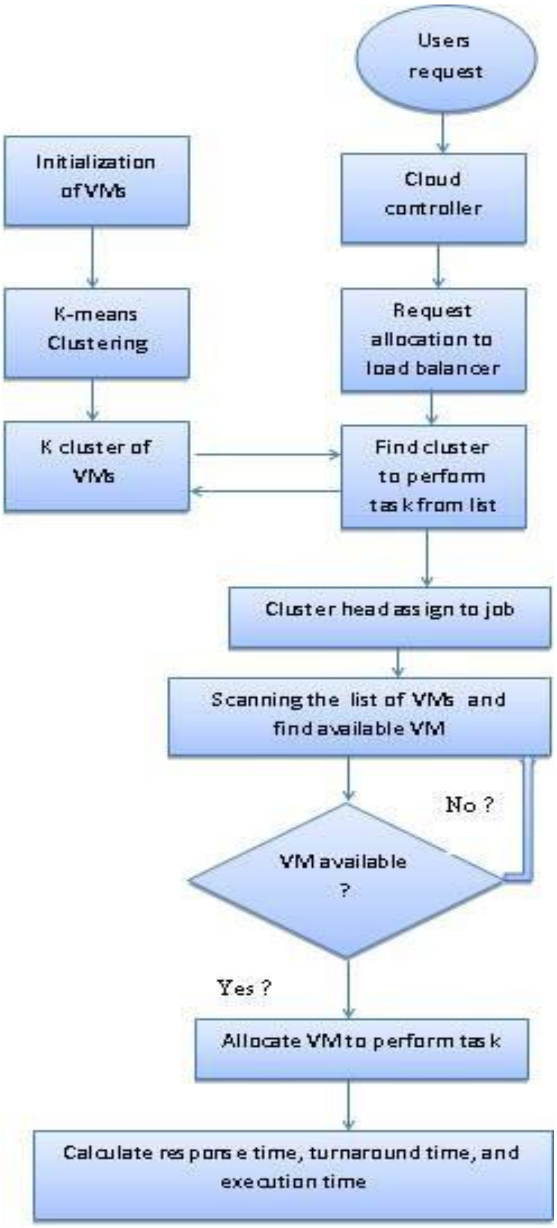

Users request
Initialization of VMs
Cloud controller
K-means Clustering
Request allocation to load balancer
K cluster of VMs
Find cluster to perform task from list
Cluster head assign to job
Scanning the list of VMs and find available VM
No ?
VM available ?
Yes ?
Allocate VM to perform task
Calculate response time, turnaround time, and execution time

Fig. 2: System Execution Flow.

The system execution flow is shown in the following fig. 2. The system must first initialise all of the virtual machines (VMs) with their unique resource types, capabilities, and statuses. VMs are clustered using k-means clustering to create k clusters. The network bandwidth, memory, and CPU processing speed are clustering-related parameters. The proper load balancer is assigned to the new request when it is received by the cloud controller. The cluster that can handle the incoming request is selected from a list of all the clusters that the load balancer has. Requests are assigned to the appropriate VM in the cluster's VM list when the cluster has been chosen. VMs that are accessible and meet the precise requirements of the task. If more than one VM meets the requirements, the work will go to the VM that was located first. Then, the VM's remaining resource quantities are updated. After the request processing is complete, the state of that VM is changed from AVAILABLE to BUSY and then back to AVAILABLE. The list of VMs' capacity is updated by the load balancer.

Waiting time, execution time, turnaround time, and throughput are all improved by this technology. Where, Response time equals VM allocation and creation time Execution time equals task size times CPU speed Time of completion minus time of generation equals turnaround time. Throughput equals the total number of tasks completed in time.

5. CONCLUSION

By eliminating the technique of scanning the complete list of VMs from the beginning, this solution gets around the issues with the present throttled algorithm. Instead, it begins the scanning of the specific cluster's VMs. In this manner, the load balancer will find the best available virtual machine inside the cluster to which the work must be given, hence reducing the time required for list scanning. Additionally, this approach assigns a better and more appropriate VM to the task according to its specifications.

This system achieves superior waiting time, execution time, turnaround time, and throughput performance by clustering the VMS into groups.

REFERENCES

[1] "Cloud Computing - Research Issues, Challenges, Architecture, Platforms and Applications: A Survey," International Journal of Future Computer and Communication, vol. 1, no. 4, December 2012.

[2] Gajender Pal and Kuldeep Kumar 2. Manish Kumar Barala published "A Review Paper on Cloud Computing" in the September 2014 issue of the International Journal for Research in Applied Science and Engineering Technology (IJRASET).

[3] "A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms," Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on, pp. 13 7 - 142, 3-4 Dec. 2012. By Nuaimi K.A., Mohamed N., and Nuaimi M.A.

[4] A load balancing model based on cloud partitioning for the public cloud was described in Tsinghua Science and Technology, vol. 18, no. 1, in February 2013, pp. 34–39.

[5] "Cluster Based Load Balancing in Cloud Computing," Surbhi Kapoor and Dr. Chetna Dabas, 2015.

[6] "Load Balancing in Cloud Computing: A State of the Art Survey," Mohammadreza Mesbahi and Amir Masoud Rahmani, I.J. Modern Education and Computer Science, vol. 3, pp. 64-78, March 2016.

[7] "Stratus: Load Balancing the Cloud for Carbon Emissions Control," Joseph Doyle and Robert Shorten, Cloud Computing, IEEE Transactions on, vol. 1, no. 1, pp. 1-1, Jan.-June 2013.

[8] An Effective Distributed Approach for Load Balancing in Cloud Computing, 2015 International Conference on Computational Intelligence and Communication Networks, Aarti Vig and Rajendra Sing.

[9] Shridhar G. Domanal and G. Ram Mohana Reddy, "Load Balancing in Cloud Environment Using a Novel Hybrid Scheduling Algorithm," 2015 IEEE International Conference on Cloud Computing in Emerging Markets

[10] "Improving Network VO Virtualization for Cloud Computing," Parallel and Distributed Systems, IEEE Transactions on, vol. 1.25, no. 3, March 2014, pp. 673-681. M. Bourguiba, K. Haddadou, and EI Korbi.

[11] Computer and Communication Technology (ICCCT), 2014 International Conference on, vol. 4, pp. 87-95, 26-28 September 2014. S. B. Shaw and A. K. Singh, "A survey on scheduling and load balancing approaches in cloud computing environment,"

[12] Jasmin James and Dr. Bhupendra Verma, "Efficient VM Load Balancing Algorithm for a Cloud Computing Environment," International Journal on Computer Science and Engineering (IJCSE), vol. 4, no. 09, pages 1658–1663, September 2012.