Cloud Partitioning Based Load Balancing Model for Cloud Service Optimization

1PUVVALA SUPRIYA, 2K.VINAY KUMAR
1M.Tech Student, Department of CSE, Gokul Institute of Technology and Sciences, Piridi village Bobbili mandalam, Vizianagaram dt. Jntu kakinada university A.P, India
2M.Tech Assistant Professor, Department of CSE, Gokul Institute of Technology and Sciences, Piridi village Bobbili mandalam, Vizianagaram dt. Jntu kakinada university A.P, India
1 su priya.puvvala@gmail.com, 2 vinyakumar.gokul0@gmail.com

Abstract-- Cloud computing is a new phenomenon or technology that paves way for new model of computing. Cloud offers many services including Infrastructure as a Service (IaaS). With respect to this service optimal utilization of infrastructure services is essential for sustainable server provision. Towards this end, in this paper, a load balancing model is designed and implemented using CloudSim. This is achieved by dividing a cloud into many partitions. Each partition can have multiple nodes for processing data. Each partition is equipped with a load balancer which monitors the load of the nodes in the partition. Each partition can have the load status such as “Idle”, “Normal” and “Overloaded”. A controller will be on top of all partitions. The controller has coordinated communication with load balancers which takes load balancing decisions on the fly based on the status information provided load balancers. We built a prototype application that demonstrates the proof of concept. The simulation results reveal that the cloud partitioning approach is very useful in load balancing for optimized cloud services.

Index Terms – Cloud computing, load balancing model, public cloud, and cloud partition

I. INTRODUCTION

Cloud computing is an emerging technology that enables individuals and businesses to have access huge amount of computing resources. It provides services such as Platform as a Service (PaaS), Infrastructure as a Service (IaaS) and Software as a Service (IaaS). Recent improvements in cloud also added another service namely mining as a Service (MaaS). Storage and computing services and mining services are provided in pay per use fashion without the need for capital investment. From the cloud service provider point of view huge computing resources are made
available. Optimal utilization of these resources is very important problem to be addressed. In other words the load balancing is the open problem that needs to be resolved. Good solutions towards it can improve user satisfaction.

As can be seen in Figure 1, the computing resources of cloud can be accessed across the world from any device across the world. These devices can avail cloud services in pay per use fashion.

The focus of this paper is on the partitioning of cloud and implements a methodology that promotes load balancing in order to optimize the services of cloud. The rest of the work is done by developing a methodology which considers the improvements on the cloud partition approach especially cloud division approach based on the geographical location of the servers. The load balancing Meta Data needs to be refreshed periodically and the refresh period needs to be determined. Load status evaluation and exploring other load balancing strategies are open issues that can be addressed. The CloudSim is used for building a simulator application that demonstrates the proof concept. CloudSim is a framework for modeling and simulation of cloud computing infrastructures and services. This framework is used to simulate the proposed work.

The remainder of the paper is structured as follows. Section II provides review of literature. Section II reviews literature on prior works. Section III presents the proposed methodology. Section IV provides implementation and results. Section V concludes the paper besides providing directions for future work.

II. RELATED WORKS
Some of the recent electronic journal articles [1]-[10] have been studied. The study of literature has provided many insights.

- In distributed computing environment such as cloud computing numerous requests are made to applications. The sudden bursts of requests can cause the servers to crash or to stop working. This is one of the potential problems that deny service to valued customers or cloud users.
- Load balancing is indispensable to overcome such problems in cloud computing environment. Many load balancing algorithms came into existence. They include simple round robin algorithm, equally spread current execution algorithm and ant colony optimization algorithm.
- Live VM migration and capacity extension are other approaches found for load balancing.
Another insight into load balancing approach is to partition the cloud. In this approach cloud is divided into multiple partitions. There is a controller to monitor all partitions with respect to their load. There is load balancer associated with each partition which takes care of balancing load at the given partition level.

A. METHODOLOGY OF PROPOSED SYSTEM

A load balancing model is designed and implemented using CloudSim. This is achieved by dividing a cloud into many partitions. Each partition can have multiple nodes for processing data. Each partition is equipped with a load balancer which monitors the load of the nodes in the partition. Each partition can have the load status such as “Idle”, “Normal” and “Overloaded”. A controller will be on top of all partitions. The controller has coordinated communication with load balancers which takes load balancing decisions on the fly based on the status information provided load balancers. The schematic overview of the proposed system is as follows.

As can be seen in Figure 2, the main controller is responsible to make decisions on load balancing by consulting load balancers associated with cloud partitions. The load balancers update the status information such as “Idle”, “Normal” and “Overloaded” based on the number of nodes which idle/normal/overloaded based on corresponding threshold value. The simulation study using CloudSim can provide can help solve the load balancing problem in cloud.

III. PROTOTYPE AND RESULTS

The prototype is implemented in Cloudsim which is a framework which enables modeling and simulation and experimenting on designing Cloud computing infrastructure. Cloudsim toolkit is developed in the GRIDS laboratory at the University of Melbourne. Cloudsim is a self-contained platform which can be used to model data centres, hosts, service brokers, scheduling and allocation policies of a large scaled cloud platform. CloudSim framework is built on the top of layer in GridSim framework. Hence CloudSim is used to model data centers, hosts, VM’s for experimenting in simulated cloud environment.
This page taking number of cloud partitions form user. Base on the input cloud partitions are created.

Figure 3 -- No. of partitions page

Figure 4 -- Controller’s page
If the user enables the load balancer then load balancer will be created in the cloud partitions.

Figure 5 – No. of virtual machines page

The user should give no of data centers and no of virtual machine. In each partition the no of virtual machines are created by load main controller.

Figure 6 -- Load balancing application home page

The above gui is main page of the load balancing. In this page virtual machines created in each partition. And partitions created in entire cloud portions.
We need to give the no of total jobs to main controller. The main controller will send the job by job to cloud partition.

We need to give the each partition capacity. When the main controller sends jobs to partitions. Once the partition capacity is full. The job is forwarded to other partition.

We need to give each virtual machine capacity. When job enters into partition. The partition sends the job into virtual machine. Once the virtual machine is overloaded then the next job is sent to other virtual machine in the same partition.
Figure 10-- Messages for user
Once the configuration completed then those messages will be displayed

Figure 11-- Job is arrived to 1st partition and 1st VM
Once the configuration is completed then main controller will create the partitions and virtual machines. The each is forwarded by main controller the jobs is arrived to 1st virtual machine in 1st partition

![Figure 12](image12.png)

**Figure 12**-- Job is arrived to 1st partition and 2nd VM

The jobs is arrived to 2nd virtual machine in 1st partition

![Figure 13](image13.png)

**Figure 13**--Job is arrived to 1st partition and 1st VM

The job is arrived to 1st virtual machine in 1st partition. In this page this virtual machine capacity is full and it going to overloaded state.
Figure 14 -- Job is arrived to 4\textsuperscript{th} partition and 4\textsuperscript{th} VM
Capacity full in all virtual machines in all partitions. Now the main controller will send the job using load balancing.

Figure 15 -- Job is arrived to 1\textsuperscript{st} partition and 2\textsuperscript{nd} VM
The load balancer will get the idle state of vm in one partition.
If all jobs arrived into partitions then the result will be displayed into the above page

IV. CONCLUSIONS AND FUTURE WORK

In this paper we studied the problem of load balancing in cloud computing environment. We proposed a simulation model based on cloud partitioning. The existing algorithm proposed in [1] solves load balancing problem by using status information such as NORMAL, IDLE, and OVERLOADED. In case of OVERLOADED state, simply no requests will be forwarded to OVERLOADED partition. In case of IDLE state, the request will be passed into the queue based on Round Robin algorithm. In the proposed cloud partitioning approach each partition is equipped with a load balancer which monitors the load of the nodes in the partition. Each partition can have the load status such as “Idle”, “Normal” and “Overloaded”. A controller will be on top of all partitions. The controller has coordinated communication with load balancers which takes load balancing decisions on the fly based on the status information provided load balancers. In future we build an alternative methodology and compare the results for finding a more suitable approach.

REFERENCES


**AUTHORS**

Puvvala Supriya is currently working towards her M.Tech degree in Gokul Institute of Technology and Sciences, Piridi village, Bobbili mandalam, Vizianagaram dt, A.P, India. Her research interests include cloud computing and Networking.

K. Vinay Kumar is working as an Assistant professor Gokul Institute of Technology and Sciences, Piridi village, Bobbili mandalam, Vizianagaram dt, A.P, India. His main research interests are data mining and big data mining.