ENHANCED EFFICIENT LOAD PREDICTION ALGORITHM IN CLOUD COMPUTING

P.Vignesh ,Sreedhar A.Manas Musthafa,S.Saran kishore
UG Students, Bachelor of Engineering
[1]M.Anandapriya, Assistant Professor
Department of Computer Science and Engineering,
Sri Krishna College of Engineering and Technology, Coimbatore, India.
[1]anandapriyam@skcet.ac.in

Abstract

In cloud computing environment, customers are allowed to scale up and down their resource usage according to their needs. Here resources are multiplexed from physical machines to virtual machines through virtualization technology. In this paper, we are trying to avoid overloading for every physical machine of an automated resources management system that uses virtualization technology for allocating resources dynamically. We develop a new algorithm for predicting the future load of each physical machine and then decide which may be overloaded next. Then we can take the necessary action to prevent the overload in the system. The experimental results support the improvements of our algorithm.

I INTRODUCTION

Cloud Computing provides us a means by which we can access the applications as utilities, over the Internet. It allows us to create, configure, and customize applications online. The term Cloud refers to a Network or Internet. In other words, we can say that Cloud is something, which is present at remote location. Cloud can provide services over network, i.e., on public networks or on private networks. Applications such as e-mail, web conferencing, customer relationship management (CRM), all run in cloud.

Cloud Computing refers to manipulating, configuring, and accessing the applications online. It offers online data storage, infrastructure and application. Cloud Computing has numerous advantages. Some of them are listed below:

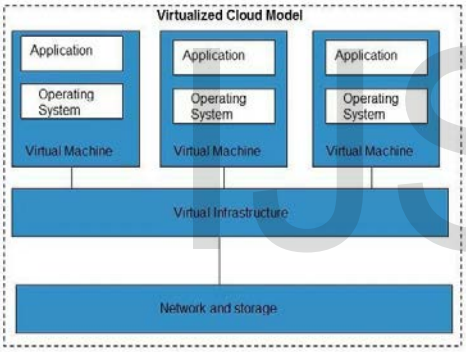
- One can access applications as utilities, over the Internet. Manipulate and configure the application online at any time.
- It does not require to install a specific piece of software to access or manipulate cloud application.
- Cloud Computing offers online development and deployment tools, programming run-time environment through Platform as a Service model.
- Cloud resources are available over the network in a manner that provides platform independent access to any type of clients.
- Cloud Computing offers on-demand self-service. The resources can be used without interaction with cloud service provider.
- Cloud Computing is highly cost effective because it operates at higher efficiencies with greater utilization. It just requires an Internet connection.
- Cloud Computing offers load balancing that makes it more reliable.
- Cloud Computing allows multiple tenants to share a pool of resources. One can share single physical instance of hardware, database and basic infrastructure.

Cloud computing has a service-oriented architecture in which services are broadly divided into three categories: Infrastructure-as-a- Service **(IaaS)**, which includes equipment such as hardware, Storage, servers, and networking components are made accessible over the Internet; Platform-as-a-Service **(PaaS)**, which includes hardware and software

computing platforms such as virtualized servers, operating systems, and the like; and Software-as-a-Service **(SaaS)**, which includes software applications and other hosted services.

Virtualization is a technique, which allows to share single physical instance of an application or resource among multiple organizations or tenants (customers). It does so by assigning a logical name to a physical resource and providing a pointer to that physical resource when demanded.

Prediction is most important for the data allocation in the cloud environment. Now a days a large number of users uses cloud services provided by the cloud providers like Amazon Web Service, Microsoft Azure, Google cloud etc. Their arises a situation in which the load on the cloud server is high and sometimes load will be less than expected.



(a).Virtualized Cloud Model

This leads to over utilization and under utilization of the resources. Therefore there is a need for load prediction algorithms in optimizing the resource utilization and balancing the load at different instances of time where the load on cloud servers varies. This helps in the minimal consumption of resources and allocating it dynamically.

II EXISITING SYSTEM

In cloud computing, customers are allowed to scale up and down their resource usage according to their needs. Here resources are multiplexed from physical machines to virtual machines through virtualization technology. Therefore the main objective is to avoid overloading for every physical machine of an automated resources management system that uses virtualization technology for allocating resources dynamically. A system that uses virtualization technology to allocate data centre resources dynamically based on application demands and support green computing by optimizing the number of servers in use.

Hence an efficient algorithm for predicting the future load of each physical machine which will decide which may be overloaded next. A cloud centre can have a large number of facility (server) nodes, typically of the order of hundreds or thousands, traditional queuing analysis rarely considers systems of this size. The coefficient of variation of task service time may be high. Due to the dynamic nature of cloud environments, diversity of user's requests and time dependency of load, cloud centres must provide expected quality of service at widely varying loads.

III PROPOSED SYSTEM

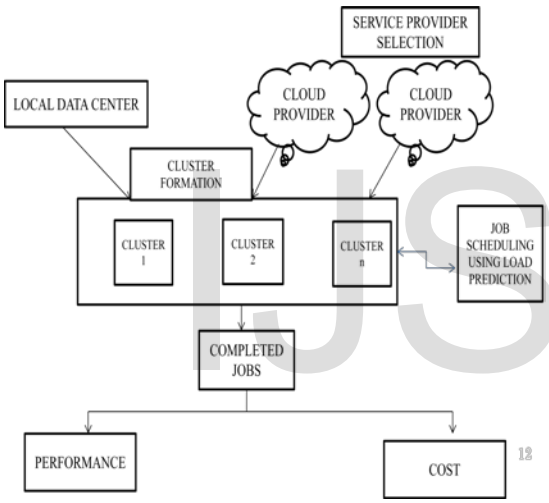
In proposed system the design and implementation of a resource management system for cloud computing services is presented for multiplexing the virtual resources to physical resources whenever the demand for the resource arises. The cloud services can maximize the profit using an optimal pricing scheme when a virtual machine migration takes place. Optimal pricing necessitates an appropriately simplified price-demand model and the model has to be changed dynamically.

Parallelism on cloud infrastructure may benefit the performance of structure creation. The cost of transferring the data from a virtual machine to the other machine may contain the cost of net grating the column in the existing cache table. In case of multiple cloud databases, the cost of data movement is incorporated in the building cost. The concept of skewness is used to measure the

unevenness in the utilization of multiple resources in servers. By minimizing the skewness the overall utilization of server resources is achieved.

Advance reservation of resources is difficult to be achieved due to uncertainty of consumer's future demand. Optimal Cloud Resource Provisioning algorithm is proposed to address the problem of uncertainty. The OCRP algorithm considers cost and price uncertainty since the priority of tasks handled by the server changes dynamically and hence is used for provisioning the multiple resources.

PROPOSED SYSTEM WITH LOAD PREDICTION



(b)Architecture Diagram

LOAD PREDICTION ALGORITHM

We need to predict the future resource needs of VMs. One of the possibility is to look inside a VM for application level statistics, e.g., by parsing logs of pending requests. Doing so requires modification of the VM which may not always be possible. Instead, we make our prediction based on the past external behaviors of VMs.

Step 1:

Calculate an exponentially weighted moving average (EWMA) using a TCP-like scheme.Here the estimated load and observed load at particular time t can be calculated using the following equation

$E(t) = \alpha * E(t - 1) + (1 - \alpha) * O(t);\ 0 < \alpha < 1$ ----------(1)

where α reflects a trade-off between stability and responsiveness.

Step 2:

We use the EWMA formula to predict the CPU load on the server. We measure the load every minute and predict the load in the next minute. Use the value for α as $\alpha = 0.7$

Step 3:

When the observed resource usage is going down, we want to be conservative in reducing our estimation. In most of the time (77%) the predicted values are higher than the observed ones. The median error is increased to 9.4% because we trade accuracy for safety.

Step 4:

When α is between 0 and 1, the predicted value is always between the historic value and the observed value. To reflect acceleration set α to a negative value. When α is between -1 and 0, the equation (1) can be transformed as (2) which is given below.

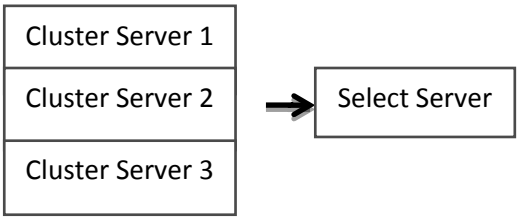
$E(t) = - |\alpha| * E(t - 1) + (1 + |\alpha|) * O(t)$;-1 < α < 0

$E(t) = O(t) + |\alpha| * (O(t) - E(t - 1))$ ---------- (2)

This prediction is done based on the past external behaviours of VMs.

CLUSTERED CLASSIFICATION

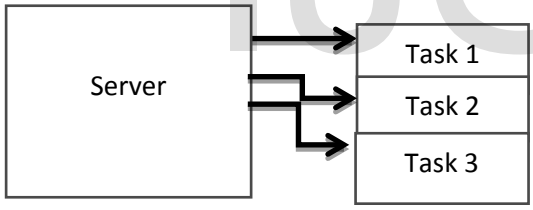
Clustered Classification is used to estimate which Server doing which job. Which is done by Monitoring Server access, cost calculation and equal sharing of jobs in Server. Group applications into service Performance of classes which are then mapped onto server clusters which parses application level information in Web requests and forwards them to the servers with the corresponding applications running. Each application can run on multiple server machines and the set of their running instances are often managed by some clustering software.



(c)Cluster Classification

DIFFERENTIATED SERVICES

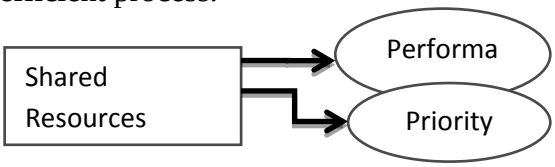
After the servers may be clustered then allocate the task which can be assigned to each server and calculate the performance and priority. Each server machine can host multiple applications. The applications store their state information in the backend storage servers. It is important that the applications themselves are stateless so that they can be replicated safely. In this process the cluster server estimates the server Capabilities and assigns which job is to be assigned to which server. The seperation of task is done on the basis of Minimum Execution Time First.



(d)Differentiated Services

DISTRIBUTIONAL CLASSES

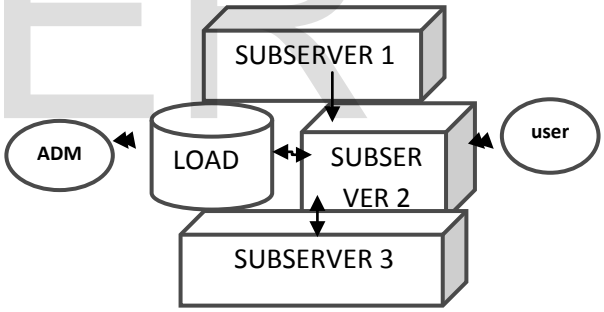
The system under consideration contains m servers which render service in order of task request arrivals (FCFS).The capacity of system is m þ r which means the buffer size for incoming request is equal to r. As the population size of a typical Server centre is relatively high while the probability that a given user will request service is relatively small, the arrival process can be modelled as an efficient process.



(e)Distributional Classes

LOAD SHIFTING

The load of data centre applications can change continuously. We only need to invoke our algorithm periodically or when the load changes cross certain thresholds. Hence, if a flash crowd requires an application to add a large number of servers, all the servers are started in parallel. Our algorithm is highly efficient and can scale to tens of thousands of servers and applications. The amount of load change during a decision interval may correspond to the arrivals or departures of several items in a row. A large load unit reduces the overhead of our algorithm because the same amount of load change can be represented by fewer items. It also increases the stability of our algorithm against small oscillation in load. On the other hand, it can lead to inefficient use of server resources and decrease the satisfaction ratio of application demands.

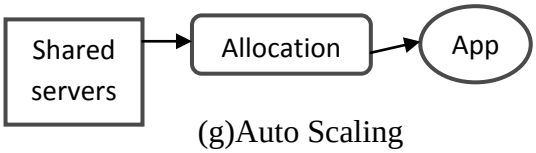


(f)Load Shifting

AUTO SCALING

The space timing calculates by the reference of Server usage. That is, the cost also calculates based on Server space utilization and Server usage. The server calculates which Server doing which job. That is monitoring Server access, cost calculation and equal sharing of jobs in Server. We analyze and compare the performance offered by different configurations of the computing cluster, focused in the execution of loosely coupled applications. Different cluster configurations with different number of

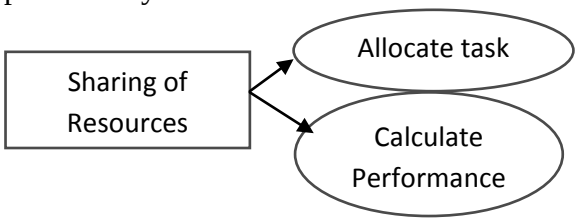
worker nodes from the three Servers Providers and different number of Jobs (depending on the cluster size), as shown in the definition of the different cluster configurations, we use the following acronyms. We want to enable the use of large-scale distributed systems for task-parallel applications, which are linked into useful work flows through the looser task coupling model of passing data via files between dependent tasks and potentially larger class of task-parallel Feature Extraction.



(g)Auto Scaling

RESOURCE UTILIZATION

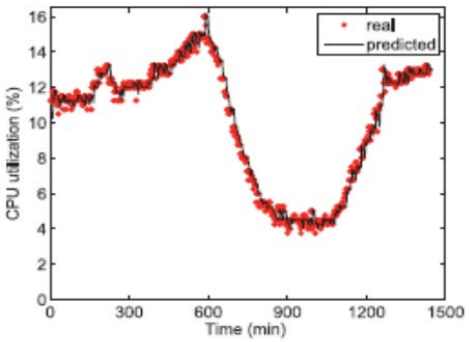
There are also some Server vendors providing auto-scaling solutions for Server users. Users are allowed to define a set of rules to control the scaling actions. However, the rules and the load balancing strategies they used are very simple. They perform the scaling actions simply when some conditions are met and balance the load evenly across all instances. Since they do not take the state of the whole system into consideration, they cannot reach a globally optimal decision. They allocate resources on shared cluster serves periodically.



(h)Resource Utilization

IV. SIMULATION RESULTS

We use our algorithm for prediction the percentage of CPU utilization. We predict the CPU load in every minute by measuring the actual loads of previous two minutes. The simulation results are shown in the figure;



V CONCLUSION

Cloud computing is progressively used in business markets and enterprise. Dynamic resource allocation is growing need of cloud providers for more number of users and with the less number of systems states a review. The proposed system multiplexes virtual to physical resources based on the changing demand. The skewness metric is used by the system to mix with different resources of virtual machines. The proposed algorithm has been achieved overload avoidance by predicting the future needs and by using green computing technology we can turn off the idle serves to conserve energy. In order to improve scheduling effectiveness we have adopted load prediction.

REFERENCES

[1] Aameek Singh, Dushmanta Mohapatra, Madhukar Korupolu. Server-Storage Virtualization: Integration and Load Balancing in Data Centers. Proc. ACM/IEEE Conf. Supercomputing, 2008

[2] Abu Sharkh.M, Jammal.M, Shami.A, Ouda.A. Resource Allocation in a Network-Based Cloud Computing Environment: Design Challenges. Nov 2013

[3] Akilandeshwari.K. Succinct Observation on Resource Management in Cloud Environment. 2013

[4] Alex Ho, Andrew, Boris Dragovic, Ian Pratt, Keir Fraser, Paul Barham, Steven Hand, Tim Harris. Xen and the

Art of Virtualization. Proc. ACM Symp. Operating Systems Principles (SOSP ’03), Oct. 2003

[5] Andrew Goldberg, Jon Currey, Kunal Talwar, Michael Isard, Vijayan Prabhakaran, Udi Wieder. Quincy: Fair Scheduling for Distributed Computing Clusters. Proc. ACM Symp. Operating System Principles (SOSP ’09), Oct. 2009

[6] Andrzej Kochut, Kirk Beaty, Norman Bobroff. Dynamic Placement of Virtual Machines for Managing SLA Violations. Proc. IFIP/IEEE Int’l Symp. Integrated Network Management (IM ’07), 2007

[7] Andy Konwinski, Anthony D. Joseph, Ion Stoica, Matei Zaharia, Randy Katz. Improving MapReduce Performance in Heterogeneous Environments. Proc. Symp. Operating Systems Design and Implementation (OSDI ’08), 2008

[8] Ariel Rabkin, Armando Fox, David A. Patterson, Gunho Lee, Ion Stoica, Matei Zaharia, Michael Armbrust, Randy H. Katz, Rean Griffith. Above the Clouds: A Berkeley View of Cloud Computing technical report, Univ. of California, Berkeley, Feb 2009

[9] Arun Venkataramani, Mazin Yousif, Prashant Shenoy, Timothy Wood. Black-box and Gray-box Strategies for Virtual Machine Migration. Proc. Symp. Networked Systems Design and Implementation (NSDI ’07), Apr. 2007 Carl A. Waldspurger. Memory Resource Management in VMware ESX Server. Proc. Symp. Operating Systems Design and Implementation (OSDI ’02), Aug. 2002

[11] Chunqiang Tang, Giovanni Pacifici, Malgorzata Steinder, Michael Spreitzer. A Scalable Application Placement Controller for Enterprise Data Centers. Proc. Int’l World Wide Web Conf. (WWW ’07), May 2007

[12] Dhruba Borthakur, Ion Stoica, Joydeep Sen Sarma, Khaled Elmeleegy,Matei Zaharia,Scott Shenker. Delay Scheduling: A Simple Technique for Achieving Locality and Fairness in Cluster Scheduling. Proc. European Conf. Computer Systems (EuroSys ’10), 2010

[13] Feng Zhao, Gong Chen, Jie Liu, Leonidas Rigas, Lin Xiao, Suman Nath,Wenbo He. Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services. Proc. USENIX Symp. Networked Systems Design and Implementation (NSDI ’08), Apr. 2008

[14] Harish Narware, GVR Kiran,Nitin Bindal, Prasad Saripalli, Ravi Shankar. Load Prediction and Hot Spot Detection Models for Autonomic Cloud Computing. 2011

[15] James Broberg, Rajkumar Buyya, Srikumar Venugopal, William Voorsluys. Cost of Virtual Machine Live Migration in Clouds: A Performance Evaluation. 2010

[16] James Scott, Paramvir Bahl, Rajesh Gupta, Ranveer Chandra, Steve Hodges, Yuvraj Agarwal. Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage. Proc. USENIX Symp. Networked Systems Design and Implementation (NSDI ’09), 2009

[17] Kang G. Shin, Pradeep Padala, Ramachandran Ramjee, Tathagata Das, Venkata N. Padmanabhan. LiteGreen: Saving Energy in Networked Desktops Using Virtualization. Proc. USENIX Ann. Technical Conf., 2010