RESEARCH ARTICLE

Efficient Allocation of Resources in Cloud Server Using Lopsidedness

B. Selvi[#1], C. Vinola[*2], Dr. R. Ravi[#3]

[#1] Student of CSE Department, Francis Xavier Engineering College, Anna University, Tirunelveli, India
[*2] Assistant Professor of CSE Department, Francis Xavier Engineering College, Anna University, Tirunelveli, India.
[#3] Professor and Head of CSE Department, Francis Xavier Engineering College, Anna University, Tirunelveli, India

[1] prettyselvibalakrishnan@gmail.com

Abstract– Cloud computing plays a vital role in the organizations resource management. Cloud server allows dynamic resource usage based on the customer needs. Cloud server achieves efficient allocation of resources through virtualization technology. It addresses the system that uses the virtualization technology to allocate the resources dynamically based on the demands and saves energy by optimizing the number of server in use. It introduces the concept to measure the inequality in multi-dimensional resource utilization of a server. The aim is to enlarge the efficient resource utilization system that avoids overload and save energy in cloud by allocating the resources to the multiple clients in an efficient manner using virtual machine mapping on physical system and Idle PMs can be turned off to save energy.

Index Terms-cloud computing, resource allocation, virtual machine, green computing

I. INTRODUCTION

In cloud computing provides the service in an efficient manner. Dynamically allocate the resources to multiple cloud clients at the same time over the network. Now-a-Days many of the business organizations using the concept of cloud computing due to the advantage with resource management and security management.

A cloud computing network is a composite system with a large number of shared multiple resources. These are focus to unpredictable needs and can be affected by outside events beyond the control. Cloud resource allocation management requires composite policies and decisions for multi-objective optimization. It is extremely difficult because of the convolution of the system, which makes it impracticable to have accurate universal state information. It is also subject to continual and unpredictable communications with the surroundings.

The strategies for cloud resource allocation management associated with the three cloud delivery models, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS), differ from one another. In all cases, the cloud providers are faced with huge, sporadic loads that contest the claim of cloud flexibility.

Virtualization is the single most efficient way to decrease IT expenses while boosting effectiveness and liveliness not only for large enterprise, but for small and mid budget organizations also.

Virtualization technology has advantages over the following aspects.

- Run multi-operating systems and applications on a single computer.
- Combine hardware to get hugely higher productivity from smaller number of servers.
- Save 50 percent or more on general IT costs.
- Speed up and make things easier IT management, maintenance, and the consumption of new applications.

The system aims to achieve two goals:

- The capability of a physical machine (PM) should be enough to satisfy the resource requirements of all virtual machines (VMs) running on it. Otherwise, the Physical machine is overload and degrades performance of its VMs.
- The number of PMs used should be minimized as long as they can still satisfy the demands of all VMs. Idle physical machine can be turned off to save energy. There is an intrinsic exchange between the two goals in the face of altering resource needs of VMs.

For overload avoidance, the system should keep the utilization of PMs Low to reduce the possibility of overload in case the resource needs of VMs increase later.

For green computing, the system should keep the utilization of PMs reasonably high to make efficient use of their energy.

It presents the design and implementation of an efficient resource allocation system that balance between the two goals. The following aids are,

- The development of an efficient resource allocation system that avoids overload in the system effectively while minimizing the number of servers use.
- To introduce the concept of "lopsidedness" to measure the uneven utilization of a server.
- By minimizing lopsidedness, the system improves the overall utilization of servers in the face of multidimensional resource constraints.
- To implement a load prediction algorithm that can capture the future resource usages of applications accurately without looking inside the VMs.
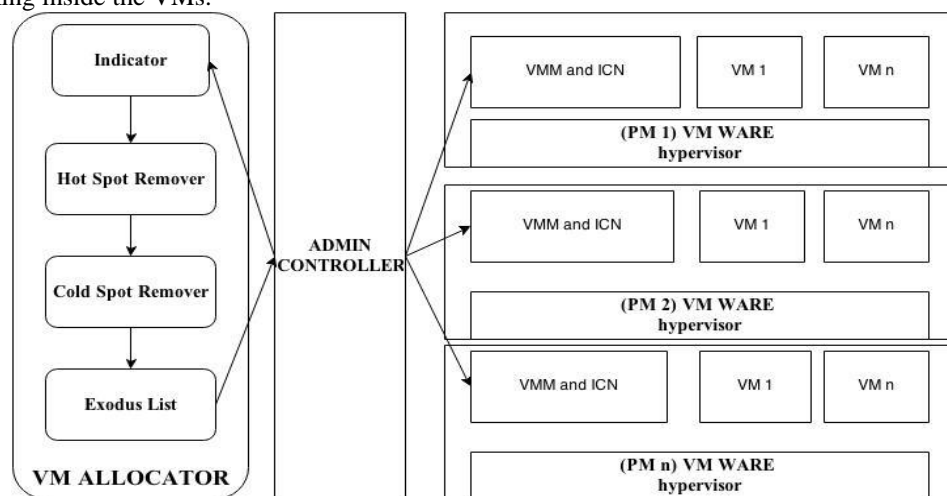

Fig.1 System Architecture

II. System Overview

The architecture of the system is presents in Fig.1. The physical machine runs with VMware hypervisor (VMM) that supports VM0 and one or more VMs in cloud server. Each VM can contain one or more applications residing it. All physical machines can share the same storage space.

The mapping of VMs to PMs is maintains by VMM. Information collector node (ICN) collects the information about VMs resource status that runs on VM0. The virtual machine monitor creates and monitors the virtual machine. The CPU scheduling and network usage monitoring is manage by VMM.

Assume with available sampling technique can measure the working set size on each virtual machine. The information collects at each physical machine and passes the information to the admin controller (AC). AC connects with VM Allocator that activated periodically and gets information from the ICN resource needs history of VMs, and status of VMs.

The allocator has several components. The Indicator Indicates the future demands of virtual machine and total load value for physical machine. The ICN at each node attempts to satisfy the input demands locally by adjusting the resource allocation of VMs sharing the same VMM.

The hotspot remover in VM allocator spots if the resource exploitation of any PM is above the Hot Point. If so, then some VMs runs on the particular PM will be move away to another PM to reduce the selected PM load.

The cold spot remover identifies the system that is below the average utilization (Cold point) of actively used PMs. If so, then it some PMs turned off to save energy. Finally, the exodus list passes to the admin controller.

III. The Lopsidedness Algorithm

The resource allocation system introduces the concept of lopsidedness to measure the unevenness in the utilization of multiple resources on a server. Let consider n be the number of resources and let consider ri be the exploitation of the ith resource. To define the resource lopsidedness of a server p by considering r is the average utilization of resources in server p. In practice, not all types of resources are performance critical and then consider bottleneck resources in the above calculation. By minimizing the lopsidedness, the system can combine different types of workloads nicely and improve the overall utilization of server resources.

A. Hot and Cold Points

The system executes periodically to evaluate the resource allocation status based on the predicted future resource demands of VMs. The system defines a server as a hot spot if the utilization of any of its resources is above a hot threshold. This indicates that the server is overloaded and hence some VMs running on it should be migrated away.

The system defines the temperature of a hot spot p as the square sum of its resource utilization beyond the hot threshold. Consider R is the set of overloaded resources in server p and rt is the hot threshold for resource r. (Note that only overloaded resources are considered in the calculation.) The temperature of a hot spot reflects its degree of overload.

If a server is not a hot spot, its temperature is zero. The system defines a server as a cold spot if the utilizations of all its resources are below a cold threshold. This indicates that the server is mostly idle and a potential candidate to turn off to save energy. However, the system does so only when the average resource utilization of all actively used servers (i.e., APMs) in the system is below a green computing threshold. A server is actively used if it has at least one VM running. Otherwise, it is inactive. Finally, The system define the warm threshold to be a level of resource utilization that is sufficiently high to justify having the server running but not so high as to risk becoming a hot spot in the face of temporary fluctuation of application resource demands.

Different types of resources can have different thresholds. For example, the system can define the hot thresholds for CPU and memory resources to be 90 and 80 percent, respectively. Thus a server is a hot spot if either its CPU usage is above 90 percent or its memory usage is above 80 percent.

B. Hot Spot Reduction

The system sort the list of hot spots in the system in descending temperature (i.e., the system handle the hottest one first). Our goal is to eliminate all hot spots if possible. Otherwise, keep their temperature as low as possible. For each server p, the system first decides which of its VMs should be migrated away. The system sort its list of VMs based on the resulting temperature of the server if that VM is migrated away. The system aims to migrate away the VM that can reduce the server's temperature the most. In case of ties, the system selects the VM whose removal can reduce the lopsidedness of the server the most. For each VM in the list, the system sees if the system can find a destination server to accommodate it. The server must not become a hot spot after accepting this VM. Among all such servers, the system select one whose lopsidedness can be reduced the most by accepting this VM. Note that this reduction can be negative which means the system selects the server whose lopsidedness increases the least. If a destination server is found, the system records the migration of the VM to that server and updates the predicted load of related servers. Otherwise, the system moves onto the next VM in the list and try to find a destination server for it. As long as the system can find a destination server for any of its VMs the system consider this run of the algorithm a success and then move onto the next hot spot. Note that each run of the algorithm migrates away at most one VM from the overloaded server. This does not necessarily eliminate the hot spot, but at least reduces its temperature. If it remains a hot spot in the next decision run, the algorithm will repeat this process. It is possible to design the algorithm so that it can migrate away multiple VMs during each run. But this can add more load on the related servers during a period when they are already overloaded. The system decides to use this more conservative approach and leave the system some time to react before initiating additional migrations.

IV. SYSTEM ANALYSIS

In Cloud Environment, the user has to give request to download the file. This request will be store and process by the server to respond the user. It checks the appropriate sub server to assign the task. A job scheduler is a computer application for controlling unattended background program execution; job scheduler is create and connects with all servers to perform the user requested tasks using this module.

In User Request Analysis, the requests are analyze by the scheduler before the task is give to the servers. This module helps to avoid the task overloading by analyzing the nature of the users request. Fist it checks the type of the file going to be download. The users request can be the downloading request of text, image or video file.

In Server Load value, the server load value is identifies for job allocation. To reduce the over load, the different load values are assign to the server according to the type of the processing file. If the requested file is text, then the minimum load value will be assign by the server. If it is video file, the server will assign high load value. If it is image file, then it will take medium load value.

In Server Allocation, the server allocation task will take place. To manage the mixed workloads, the job-scheduling algorithm is follow.

In this the scheduling, depends upon the nature of the request the load values are assign dynamically. Minimum load value server will take high load value job for the next time. High load value server will take minimum load value job for next time. The aim is to enlarge the efficient resource utilization system that avoids overload and save energy in cloud by allocating the resources to the multiple clients in an efficient manner using virtual machine mapping on physical system and Idle PMs can be turned off to save energy.
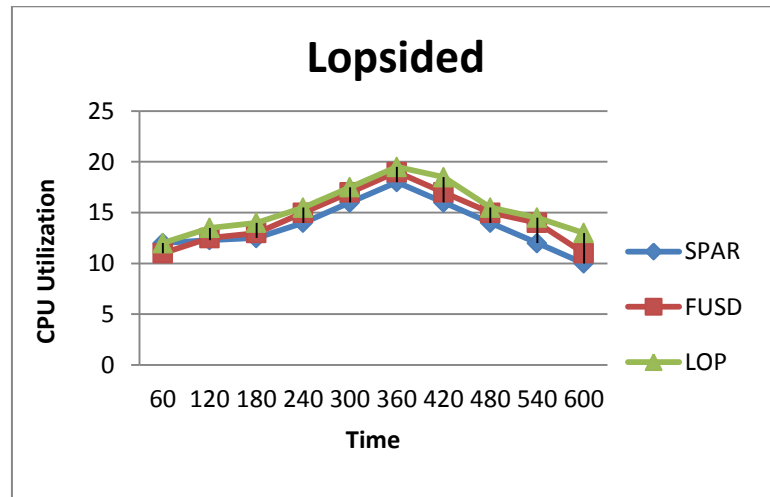
Fig. 2 Comparison graph

IV. CONCLUSION

It presented by the design, implementation and evaluation of efficient resource allocation system for cloud computing services. Allocation system multiplexes by mapping virtual to physical resources based on the demand of users. The contest here is to reduce the number of dynamic servers during low load without sacrificing performance. Then it achieves overload avoidance and saves energy for systems with multi resource constraints to satisfy the new demands locally by adjusting the resource allocation of VMs sharing the same VMM and some of not used PMs could potentially be turn off to save energy. Future work can on prediction algorithm to improve the stability of resource allocation decisions and plan to explore using AI or control theoretic approach to find near optimal values automatically.

REFERENCES

[1] Anton Beloglazov and Rajkumar Buyya (2013), 'Managing Overloaded Hosts For Dynamic Consolidation of Virtual Machines In Cloud Data Centers Under Quality of Service Constraints', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 7, pp. 1366-1379.
[2] Ayad Barsoum and Anwar Hasan (2013), 'Enabling Dynamic Data And Indirect Mutual Trust For Cloud Computing Storage Systems', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 12, pp. 2375-2385.
[3] Daniel Warneke, and Odej Kao (2011), 'Exploiting Dynamic Resource Allocation For Efficient Parallel Data Processing In The Cloud', IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No. 6, pp. 985-997.
[4] Fung Po Tso and Dimitrios P. Pezaros (2013), 'Improving Data Center Network Utilization Using Near-Optimal Traffic Engineering', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 6, pp. 1139-1147.
[5] Hong Xu, and Baochun Li (2013), 'Anchor: A Versatile and Efficient Framework for Resource Management in The Cloud', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 6, pp. 1066-1076.
[6] Jia Rao, Yudi Wei, Jiayu Gong, and Cheng-Zhong Xu (2013), 'Qos Guarantees And Service Differentiation For Dynamic Cloud Applications', IEEE Transactions on Network and Service Management, Vol. 10, No. 1, pp. 43-54.
[7] Junwei Cao, Keqin Li,and Ivan Stojmenovic (2013), 'Optimal Power Allocation and Load Distribution For Multiple Heterogeneous Multicore Server Processors Across Clouds and Data Centers', IEEE Transactions on Computers, Vol. 32, No. 99, pp.145-159.
[8] Kuo-Yi Chen, Morris Chang. J, and Ting-Wei Hou (2011), 'Multithreading In Java: Performance and Scalability on Multicore Systems', IEEE Transactions On Computers, Vol. 60, No. 11, pp. 1521-1534.
[9] Olivier Beaumont, Lionel Eyraud-Dubois, Christopher Thraves Caro, and Hejer Rejeb (2013), 'Heterogeneous Resource Allocation Under Degree Constraints', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 5, pp. 926-937.
[10] Rafael Moreno-Vozmediano, Ruben S. Montero, and Ignacio M. Llorente (2011), 'Multicloud Deployment Of Computing Clusters For Loosely Coupled MTC Applications', IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No. 6, pp. 924-930.
[11] Sangho Yi, Artur Andrzejak, and Derrick Kondo (2012), 'Monetary Cost-Aware Checkpointing And Migration on Amazon Cloud Spot Instances', IEEE Transactions on Services Computing, Vol. 5, No. 4, pp. 512-524.
[12] Sheng Di and Cho-Li Wang (2013), 'Dynamic Optimization of Multiattribute Resource Allocation In Self-Organizing Clouds', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 3, pp. 464-478.
[13] Xiangping Bu, Jia Rao, and Cheng-Zhong Xu (2013), 'Coordinated Self-Configuration of Virtual Machines And Appliances Using A Model-Free Learning Approach', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 4, pp.681-690.

[14] Xiaocheng Liu, Chen Wang, Bing Bing Zhou, Junliang Chen, Ting Yang, and Albert Y. Zomaya (2013), 'Priority-Based Consolidation Of Parallel Workloads In The Cloud', IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 9, pp. 1874-1883.

[15] Ying Song, Yuzhong Sun, and Weisong Shi(2013), 'A Two-Tiered On-Demand Resource Allocation Mechanism For VM-Based Data Centers', IEEE Transactions on Services Computing, Vol. 6, No. 1, pp. 116-129.

AUTHOR(S) PROFILE

B. Selvi is presently studying M.E second year Computer Science and Engineering in Francis Xavier Engineering college, Tirunelveli. She has completed her B.E Computer Science and Engineering from Francis Xavier Engineering College, Tirunelveli in 2012. Her fields of interest are Virtual Machine and Cloud Computing.

C. Vinola is presently working as an Associate Professor in the Department of Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli. She has completed her B.E. Electrical and Electronics Engineering in Government College of Engineering, Tirunelveli. She has completed her M.E. Computer Science and Engineering in Mepco Schlenk Engineering College and have a total of 8 years of teaching experience in various reputed institutions.

Dr. R. Ravi is an Editor in International Journal of Security and its Applications (South Korea). He is presently working as a Professor & Head and Research Centre Head, Department of Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli. He completed his B.E in Computer Science and Engineering from Thiagarajar College of engineering, Madurai in the year 1994 and M.E in Computer Science and Engineering from Jadavpur Government research University, Kolkata. He has completed his Ph.D in Networks from Anna University Chennai. He has 18 years of experience in teaching as Professor and Head of department in various colleges. He published 12 International Journals, 1 National Journal. His areas of interest are Virtual Private networks, Networks, Natural Language Processing and Cyber security