Temperature- Aware Virtual Machine Scheduling in Green Clouds

Vikas Kumar

Abstract— This paper present the temperature- aware virtual machine scheduling in green clouds which is design to maintain the temperature of virtualized cloud system below critical temperature threshold by scheduling VMs according to temperature of node and insures reliable quality of service. High temperature gradients degrade reliability and performance therefore it requires vigorous cooling in order to keep the equipment and the software stable. Moreover, high energy consumption not only increases operational cost, which reduces the profit margin of Cloud providers, but also leads to high carbon emissions which is not friendly for environment. Thus apart from saving energy and money by avoiding huge investment on cooling, it also reduces carbon footprints.

Index Terms—cloud computing, virtual machine

I. INTRODUCTION

This document describe proposed technique to maintain the temperature of virtualized cloud system below critical temperature threshold by scheduling VMs according to temperature of node and insures reliable quality of service. This technique consists of two levels. At top, there is load balancer whose core functions are to entertain new VM requests and control the migration of VMs from one node to other node. This paper comprises of six sections including the present one which describes the goal of this system. Section II shows research based papers which illustrates related work in cloud computing. Section III gives a brief introduction regarding architecture of scheduling. Section IV shows working of proposed scheduling technique which illustrates algorithms for VM migration and allocation of node to VM request. And at last section V describe the conclusion and references.

II. PAPERS BASED ON RELATED WORK

Nattakarn Phaphoom et al. [1] provide a comprehensive review on the building blocks of cloud computing and relevant technological aspects. It focuses on four key areas including architecture, virtualization, data management, and security issues.

Gaurav Dhiman et al. [2] present v Green, a multi-tiered software system for energy efficient computing in virtualized environments. It comprises of novel hierarchical metrics that capture power and performance characteristics of virtual and physical machines, and policies, which use it for energy efficient virtual machine scheduling across the whole deployment.

Manuscript received on June, 2013.

Vikas Kumar, Computer Science and Applications, School of Mathematics and Computer Applications, Thapar University, Patiala, India.

Ramesh et al. [3] explains basic power management scheme in the general computing as well as grid computing. And this paper strongly performed an analysis on various categories of real time grid systems. The power consumption on various grid levels based on multiple volumes in the organization level is analyzed. The conclusion is focused the future requirement of research direction in the energy efficient system design of grid computing

Barroso et al. [4] describes energy-proportional designs which enable large energy savings in servers, potentially doubling their efficiency in real-life use. Achieving energy proportionality will require significant improvements in the energy usage profile of every system component, particularly the memory and disk subsystems.

Aman Kansal et al. [5] describe the challenges developers face in optimizing software for energy efficiency by exploiting application-level knowledge. To address these challenges, we propose the development of automated tools that profile the energy usage of various resource components used by an application and guide the design choices accordingly.

Henri Arjamaa et al. [6] present energy consumption estimates of ICT equipment in Finland and in three important industrial countries, namely the United States, Germany, and the United Kingdom. In addition, a worldwide estimate of the energy consumption of data centers is presented. The results are then analyzed, which give answers to questions, such as how valid are the estimation methods used and are the estimation methods comparable with each other.

Christopher K. Lennard et al. [7] describe resynthesis procedures used for reducing power consumption in CMOS networks have produced poor results as they select nodes for resynthesis based upon local circuit properties. In this, a technique is presented for optimizing the choice of regions used in resynthesis. The cost function which is developed is able to predict the amount of global improvement in power expected through the resynthesis of network nodes under both zero as well as arbitrary delay assumptions.

Pinheiro et al. [8] have proposed a technique for managing a cluster of physical machines with the objective of minimizing the power consumption, while providing the required Quality of Service (QoS). The authors use the throughput and execution time of applications as constraints for ensuring the QoS. Here nodes are assumed to be homogeneous. The algorithm periodically monitors the load and decides which nodes should be turned on or off to minimize the power consumption by the system, while providing expected performance.

Srikantaiah et al. [9] have investigated the problem of dynamic consolidation of applications in virtualized heterogeneous systems in order to minimize energy consumption, while meeting performance requirements. The authors have explored the impact of the workload consolidation on the energy-per-application metric depending on both CPU and disk utilizations.

Elnozahy et al. [10] have investigated the problem of power-efficient resource management in a single web-application environment with fixed response time and load-balancing handled by the application. The two main power-saving techniques are switching power of computing nodes on or off and Dynamic Voltage and Frequency Scaling (DVFS).

Nathuji and Schwan et al. [11] have studied power management techniques in the context of virtualized data centers, which has not been done before. Besides hardware scaling and VMs consolidation, the authors have introduced and applied a new power management technique called “soft resource scaling.

Dodonov and De Mello et al. [12] have proposed an approach to scheduling distributed applications in Grids based on predictions of communication events. They have proposed the migration of communicating processes if the migration cost is lower than the cost of the predicted communication with the objective of minimizing the total execution time.

Guo et al. [13] have proposed and implemented a virtual cluster management system that allocates the resources in a way satisfying bandwidth guarantees. The allocation is determined by a heuristic that minimizes the total bandwidth utilization. The VM allocation is adapted i.e. migration is performed when some of the VMs are reallocated or power off but protocols for the migration are defined statically.

Berral et al. [14] presented a theoretical approach for handling energy-aware scheduling in data centers. Here, the authors propose a framework which provides an allocation methodology using techniques that include turning on or off machines, power-aware allocation algorithms and machine learning to deal with uncertain information while the expected QoS is maintained through the avoidance of SLA violations.

Song et al. [15] have proposed resource allocation to applications according to their priorities in multi-application virtualized cluster. The approach requires machine learning to obtain utility functions for the applications and define application priorities.

III. ARCHITECTURE

Our proposed technique consists of two levels. At top, there is Load Balancer whose core functions are to entertain new VM requests and control the migration of VMs from one node to other node. To do it functions efficiently, it will keep track of each node in Virtualized Cloud System and maintain following information in tabular form say Global List:-

- Current Temperature.
- Optimal Temperature.
- Critical Temperature.
- CPU Utilization.
- Free Memory (RAM)

Load Balancer will sort above information in increasing order of current temperature of nodes.

Bottom level .i.e. second level consists of physical nodes on which VMs run. Every node will be connected to Load Balancer. Apart from Load Balancer’s Global List every node will maintain a Local List which contain following information regarding VM in it:-

- VM Name.
- Type of Operating System
- CPU Utilization
- Number of Processors required

Every node sorts its Local List in increasing order of CPU utilization of VMs.

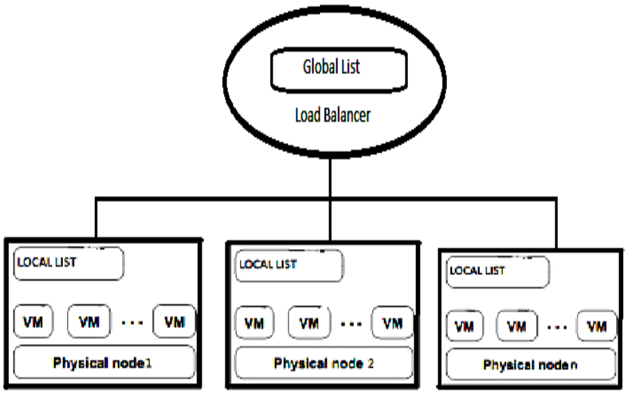


Fig.1. System Architecture

Updating Global and Local Lists:

Global List maintained by Load Balancer is to be updated after a fixed interval of time. This fixed interval would be set by the cloud administrator. Duration of this interval would depend upon the complexity of system and services provided by system. Beside this fixed interval Load Balancer will update Global List on occurrence of any of following events:-

- Request for new VM arrives.
- Migrations of VMs.
- Shutdown of VMs
- Powers on of VMs.

Local List which is maintained by every node is updated under the occurrence any of following events:-

- Migrations of VMs.
- Power on of VMs.
- Shutdown of VMs.

Waiting Queue:

It is the queue, through which request for new virtual machines is entertained. Every new request for virtual machine is added at end of Waiting Queue. Load Balancer will service VM requests in same order as appeared in Waiting Queue. If VM request gets node, then Load Balancer will remove that VM request from Waiting Queue otherwise move further in Waiting Queue and service the queue till end.

Critical Queue:

It is just like simple queue which contain the information of those nodes which have Current Temperature equals to its Critical Temperature or more than its Critical Temperature. Critical Queue is used in migration of VMs from one physical node to other in order to optimizing the temperature of Virtualized Cloud System. Information in Critical Queue is sorted in decreasing order of Current Temperature of nodes.

Critical Temperature:

It is core temperature which is set by original equipment manufacturer (OEM), running the core on Critical Temperature or beyond requires significant amount of cooling to perform computation optimally.

Moreover, running machine for significant amount of time beyond Critical Temperature may produce irreversible changes in hardware. And degrades hardware’s computation efficiency and also decreases the life of core. This Critical Temperature is different for different core e.g. corei3/i5/i7 has 65°C, 72°C for Dual core 2080 .

Optimal Temperature:

It is core temperature set by OEM, core's performance does not degrade or no instability occurs while running core on Optimal Temperature or below Optimal Temperature. Moreover, running core on Optimal Temperature or below Optimal Temperature doesn't require any extensive cooling. So running core on this temperature or below this temperature, we can save cost of cooling and increase life of hardware.

IV. PROPOSED ALGORITHM

1. Algorithm for VM Migration:

Step1.FOREACH node in Global List

Step2. IF (Current Temperature >= Critical Temperature)

Step3. Put node in Critical Queue

Step4. END IF

Step5. END FOREACH

Step6. Sort Critical Queue in decreasing order of Current Temperature of nodes

Step7. FOREACH node in Critical Queue

Step8. WHILE (Current Temperature > =Optimal Temperature)

Step9. Find node in Global List which suit current VM in local list in Critical Queue

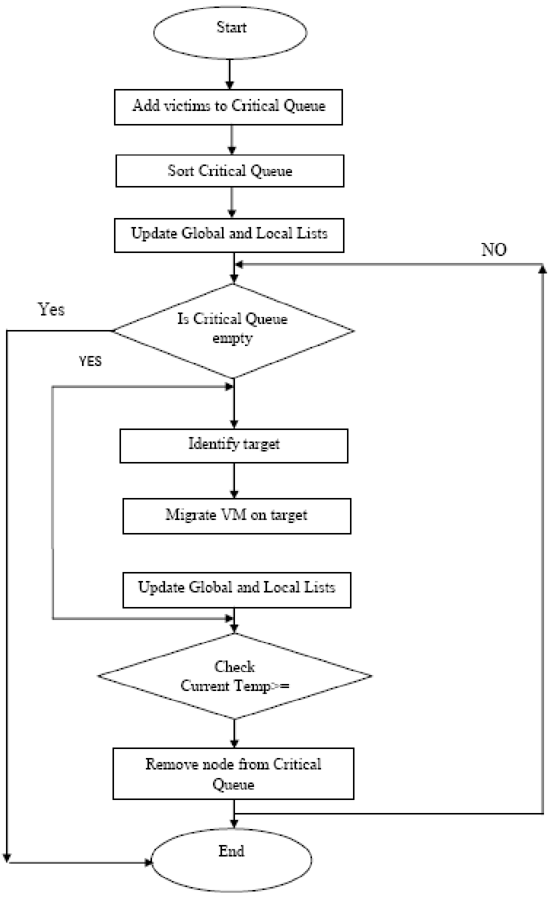
Step10. Migrate VM

Step11. Update Global List and Local Lists

Step12. END WHILE

Step13. END FOREACH

1. Flow chart for VM Migration



2. Algorithm for Allocation of Node to VM Request:

Step1. IF (new VM request arrives == true OR waiting time ==T)

Step2. Put new request in Waiting Queue

Step3. FOREACH VM request in Waiting Queue

Step4. FOREACH node in Global List

Step5. Find most appropriate node

Step6. IF (node found == true)

Step7. Allocate node

Step8. Update Global and Local Lists

Step9. Else wait in Waiting Queue

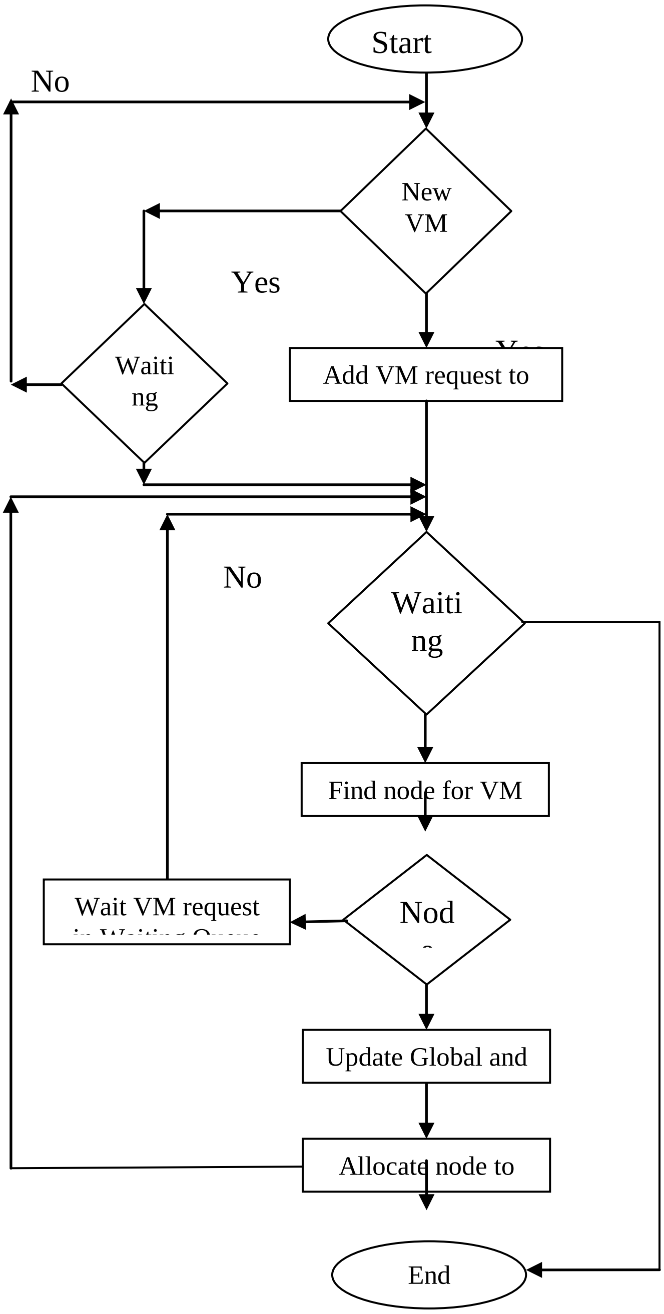
Step10. END IF

Step11. END FOREACH

Step12. END FOREACH

Step13. END IF

2. Flow chart for allocation of node to VM request



V. CONCLUSION AND REFRENCES

As the prevalence of Cloud Computing continues to rise, the need for power saving mechanisms and reducing CO_2 footprint within the Cloud is increasing. In this, a new technique "Temperature-Aware Virtual Machine Scheduling in Green Clouds" is presented and is implemented in order to improve energy efficiency of a datacenter.In this thesis, a new approach for scheduling of virtual machines (VMs) in Cloud environment is presented that provides efficient green enhancements within a scalable Cloud Computing architecture. Proposed thesis "Temperature-Aware Virtual Machine Scheduling in Green Clouds" aims to maintain the temperature of Virtualized Cloud system below critical temperature threshold by scheduling VMs according to temperature of node and insures reliable quality of service (QoS).

REFERENCES

[1] Phaphoom.N, Wang. X, Abrahamson.P." Foundations and Technological Landscape of Cloud Computing" (ISRN Software Engineering Volume 2013 (2013), Article ID 782174, 31 pages)

[2] Dhiman.G, Marchetti.G, Rosing.T "v: Green: a system for energy efficient computing in virtualized environments" (Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design, page 243-248, published in ACM, 2009)

[3] D. Ramesh, A. Krishnan "An Analysis on Energy Efficient System Design in Grid Computing"(Second International Conference, CCSIT 2012, Bangalore, India, January 2-4, 2012. Proceedings, pp 421-428)

[4] Barroso, L.A. Holzle, U. "The Case for Energy-Proportional Computing" (IEEE computer society, Volume: 40, Issue: 12, Dec. 2007 Page:33 – 37, ISSN :0018-9162)

[5] Kansa.A, Zhao.F. "Fine-Grained Energy Profiling for Power-Aware Application Design" (http://research.microsoft.com/en-us/um/people/zhao/pubs/hotmetrics08joulemeter.pdf)

[6] Arjamaa.H, "Energy Consumption Estimates of Information and Communication Technology: synthesis and analysis" (http://www.cse.tkk.fi/en/publications/B/5/papers/Arjamaa_final.pdf)

[7] Christopher K. Lennard A. Richard Newton," An Estimation Technique to Guide Low Power Resynthesis" (http://pdf.aminer.org/000/436/871/an_estimation_technique_to_guide_low_power_resynthesis_algorithms.pdf)

[8] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, "Load Balancing and Unbalancing for Power and Performance in Cluster-Based Systems" (Workshop on Compilers and Operating Systems for Low Power, pp: 182–195, 2009.)

[9] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy Aware Consolidation for Cloud Computing", (Cluster Computing, Vol. 12, pp: 1–15, 2009.)

[10] E. Elnozahy, M. Kistler, R. Rajamony, "Energy-Efficient Server Clusters" (Power-Aware Computer Systems, pp: 179-197, 2003)

[11] R. Nathuji and K. Schwan, "Virtualpower: Coordinated Power Management in Virtualized Enterprise Systems" (ACM SIGOPS Operating Systems Review, Vol. 41, pp: 256-278, 2007.)

[12] E. Dodonov, R. de Mell, "A Novel Approach for Distributed Application Scheduling Based on Prediction of Communication Events" (Future Generation Computer Systems, Vol. 5, pp: 740-752, 2010.)

[13] C. Guo, G. Lu, H. Wang, S. Yang, C. Kong, P. Sun, W. Wu,Y. Zhang, "Secondnet: A Data Center Network Virtualization Architecture with Bandwidth Guarantees", (6th International Conference on emerging Networking Experiments and Technologies, USA, 2010.)

[14] J. L. Berral, R. Nou, F. Julia, "Towards Energy-Aware Scheduling in Data Centers using Machine Learning" (1st International Conference on Energy-Efficient Computing and Networking, Passau, Germany, 2010.)

[15] Y. Song, H. Wang, Y. Li, B. Feng, Y. Sun, "Multi-Tiered On-Demand Resource Scheduling for VM-Based Data Center"(9th IEEE/ACM International Symposium on Cluster Computing and the Grid, China, pp: 148–155, 2009.)

Vikas Kmar has done his M.tech from Thapar University, Patiala, Punjab. He is an active researcher in field of cloud computing.