OPTIMIZED VIRTUAL MACHINE TREE BASED SCHEDULING TECHNIQUE IN CLOUD USING K-WAY TREES

Prasanna Kumari.M
M.Tech Computer Science Dept.
LBS collage of engineering
Kasaragod, Kerala,India

Abstract: Task scheduling plays a key role in cloud computing systems. Scheduling of tasks cannot be done on the basis of single criteria but under a lot of rules and regulations that we can term as an agreement between users and providers of cloud. This agreement is nothing but the quality of service that the user wants from the providers. Providing good quality of services to the users according to the agreement is a decisive task for the providers as at the same time there are a large number of tasks running at the provider's side. We use internet daily for Different purposes but never think about the vast number of services provided by different service providers. Where do they store and how they control? The answer is Cloud Computing. Cloud Computing is the technology specially builds to manage such kind of data and provide it whenever needed. Cloud computing refers to services that run in a distributed network and are accessible through common internet protocols. It merges a lot of physical resources and offers them to users as services according to service request. We focuses on issues of task distribution over the cloud resources by the use of k-way virtual machine tree based task scheduling. The results show that the proposed algorithm gives better results in terms of response time of the cloud leading to a better QoS.

Index Terms: Virtualization, Virtual Machine Tree, Cloud Computing

I. INTRODUCTION

Cloud computing is an internet based computing, which provides dynamically scalable and virtualized resources as a service to the requester using pay per use model. Cloud computing is nothing but the way in which an infrastructure deploys and delivers application. Cloud computing offering any kind of software or hardware services through internet. The main purpose behind using this technology is to decrease costs and increase efficiency and performances. Cloud computing elements are, clients, data centers, distributed server. Clients are the devices that the end users interact with to manage their information on the cloud .The data center is the server where the application to which you subscribe is housed. The distributed server is the servers they are housed in the geographically disparate locations. Cloud computing basically includes data centers and application delivery networks. A request from client reaches to the infrastructure and through application delivery network it is send to the cloud data center locations. Now at this point there are hundreds and thousands of client requests are approaching. To tackle with such situation these data centers are always interconnected and uses the technique called virtualization. They create the number of replicas of original data in order to provide the service to maximum number of clients which will improve the overall efficiency of system. They make user to work on these virtual platforms. Because the resources are limited and the need is growing day by day. If something were to happen at one site, causing a failure, the service would be still accessed through another site. Cloud computing technology is used to make cloud resource efficient for better profits as service. Virtualization is used in cloud computing to efficiently use the available resources, for that task scheduling has a better role and k way virtual machine tree is used (i.e. unary, binary, ternary….. k way trees). Task scheduling plays a key role to improve flexibility and reliability of systems in cloud. The main reason behind scheduling tasks to the resources in accordance with the given time bound, which involves finding out a complete and best sequence in which various tasks can be executed to give the best and satisfactory result to the user. The demand for computing resources is increasing concurrently. Cloud can be viewed as a new IT delivery mechanism which is made so by the Service Oriented Architecture. In cloud, resources are provided to the requester as a service. Services are in the form of Software, Platform as well as Infrastructure.

II. RELATED WORKS

A number of Algorithms have been proposed previously with the motive of task scheduling in cloud computing. Genetic simulated annealing algorithm scheduled the tasks on the basis of Quality of service requirements like cost, time, bandwidth of network, reliability, distance of different type of tasks. Another algorithm was proposed on the basis of Activity based Costing (ABC) which assigns priority levels for each task and uses cost drivers. Both performance of activities and cost of the object are measures by the use of this approach. An improved cost based scheduling algorithm is presented in paper [6] for making efficient mapping of tasks to available resources in cloud. The algorithm measures both resource cost and computation performance. Algorithm improves computation/communication ratio by grouping the user tasks according to particular cloud resources processing capability. In paper [2] authors introduce a Multiple QoS Constrained Scheduling Strategy to schedule multiple workflows. The proposed system consists of three core components: Pre-processor, Scheduler and Executor. An approach based on Virtual machine tree was proposed in previously which distributed the task in a binary tree structure leading to an improvement in the efficiency of the system. Proposed algorithm is based on Virtual machine tree algorithm, with an improvement using k-way tree in place of binary trees.

III. . PROBLEM STATEMENT

In the world of Cloud Computing, scheduling of the re-questers task is an interesting issue which is open for research. The success of this rising model is dependent on the effective-ness of techniques used to execute the requester's task in the most optimal way. Thus to achieve this, the following scenario has been taken up as the problem statement. Submitting the tasks provided by a user to the virtual Machines is a problem. Taking following scenario, let C is the cloud we are working with then, C = {dc1, dc2, dc3... dcn} i.e. the cloud consists of group of datacentre dc1 to dcn each datacentre consists of number of host machines in it, which are expressed as, dci = {h1, h2, h3... hm}. Each host machine is itself consists of a number of virtual machines running on that host, which can further represented as hj = {vmj1, vmj2, vmj3, ..., vmjn}. Each virtual machine can be distinguished by the value of its id and MIPS (million instruction per second it can execute), so vmjk = {vmid, vmMIPS} where vmid represents the identification number of the virtual machine and vm MIPS represents the MIPS value of that particular virtual machine. We have a list of cloudlets to be executed in the system, represented by CL (cloudlet list), so CL can be represented as CL = {c1,c2,c3,.....,cp} The list of cloudlets is required to be executed in the cloud, which can be done in several different ways leading to different execution time, The Problem of Load Distribution is how to allocate these cloudlets in the cloud such that the average execution time and average wait time for this group gets minimized.

IV. PROPOSED METHEDOLOGY

The cloud consists of numerous number of virtual machines with variable capacity of processing, The way in which the tasks are submitted to the virtual machines affects the execution time as well as the waiting time of the all the cloudlets. The submission order of the tasks and the Virtual Machines in which these tasks are executed greatly influences the execution time of the entire workload. For an optimal scheduling strategy, the tasks and Virtual Machines binding must be wisely chosen. In the proposed novel scheduling mechanism, first we priorities the tasks and Virtual Machines. We create a tree based data structure called Virtual Machine Tree (VMT) in which each nodes of a tree represents a Virtual Machine. Then grouping of task is done based on number of leaves in the VMT. The modified DFS algorithm will identify the suitable Virtual Machines, for which the submitted tasks will be executed. The details about the prioritizing, grouping and construction of VMT is given below.

A. Prioritizing

In the proposed strategy, the tasks are initially prioritized according to their size such that one having highest size has highest rank. The Virtual Machines are also ranked (prioritized) according to their MIPS value such that the one having highest MIPS has the highest rank. Thus, the key factor for prioritizing tasks is their size and for VM is their MIPS.

B. Virtual Machine Tree (VMT)

Second step is to create the virtual machine tree (VMT). It is a binary tree or ternary tree contains N number of nodes. Where N indicates the number of nodes in the VMT and have the property of binary tree such as each node has at least zero child or at most two child and property of ternary tree such as each node has at least zero child or at most three child's. There is one special property in VMT. Node value at level L+1 is lesser than or equal to node value of level L where L should be greater than or equal to zero. Each node of virtual machine tree contains child nodes; it may be one, two or three. The term leaf node referred nodes which don't have any child node. The term internal node represents the nodes which have child nodes. Representation of VMT is given below. Consider a 5 computational specific Virtual Machines represented by their Id and MIPS as V = {{0, 250}, {1, 1000}, {2, 250}, {3,500}, {4, 250}}.Figure below shows the VMT. The VMT is constructed based on the prioritized order of Virtual Machines from left to right, such that Virtual Machine with highest MIPS becomes the root.

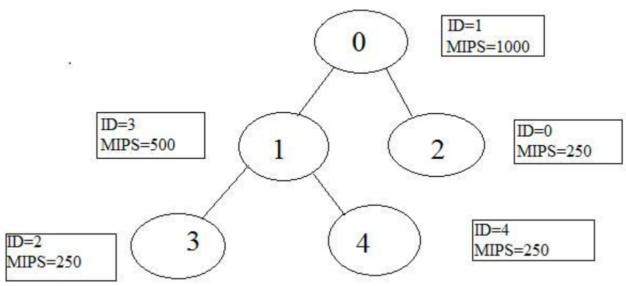


Fig.1. Virtual Machines as a nodes of a VMT

Here VMT with the root node representing the Virtual Machine with Id 1 and MIPS 1000. The root node has two children. The left child node represents the Virtual Machine with Id 3 and MIPS 500. The right child node represents the Virtual Machine with Id 0 and MIPS 250. Similarly node which represents the Virtual Machine with Id 3 and MIPS 500 has 2 children. The left child of this node represents the Virtual Machines with Id 2 and MIPS 250, right child represents the Virtual Machine with Id 4 and MIPS 250 respectively.

C. Grouping

Steps of execution of cloudlets in the virtual machine group are according the following method.The virtual machines and the cloudlets are first sorted in descending order. The cloudlets are going to sorted on the basis of their number of instructions and the virtual machines are going to be sorted on the basis of their MIPS value which states how many instructions that particular virtual machine can execute in one second. Assuming that we have a set of five virtual machines at any point of time in our cloud {{0,1000},{1,700},{2,600},{3,300},{4,300}} with id's 0 to 4 and MIPS values are considered sorted after preprocessing. In the next step virtual machine tree is created by picking the virtual machines from the list and adding them to the tree one by one from left to right. Considering the example above the tree is constructed by first adding vmid = 0 to the root then adding vmid = 1 and vmid = 2 respectively as its children so on. Tree can be constructed for any number of virtual machine and any number of children, binary tree was previously used and we consider ternary tree representation to see the effect of increasing the number of children.We have to make grouping of cloudlets. we represent total number of cloudlets as TOTAL_CLOUDLETS and Number of leaf nodes in the system be LEAFS and the cloudlets are then divided into groups where group size can be expressed as,

GROUP_SIZE = ⌊TOTAL_CLOUDLETS / LEAFS⌋

And NUM_GROUPS = LEAFS + 1

The additional group contains the left out elements from the list of cloudlets. We assume that the set of cloudlets are, CL = {{0,200},{1,1700},{2,1600},{3,1200},{4,1600},{5,1800},{6,400},{7,100},{8,30},{9,1700},{10,500},{11,1700},{12,300},{13,1600},{14,1800},{15,400},{16,100},{17,300},{18,1700}

After the preprocessing of cloudlets we get

CL={{5,1800},{14,1800},{1,1700},{9,1700},{11,1700},{18,1700},{2,1600},{4,1600},{13,1600},{3,1200},{10,500},{6,400},{15,400},{8,300},{12,300},{1,300},{0,200},{7,100},{16,100}} in the decreasing order. After that cloudlet set is to be divided into group. For the above set of virtual machine the number of leafs equal to 3.

TOTAL_CLOUDLETS = 19

LEAFS = 3

GROUP_SIZE = ⌊19 / 3⌋ = 6

NUM_GROUPS = LEAFS + 1 = 3 + 1 = 4

So, the groups G1, G2, G3, and G4 can be represented as

G1={{5,1800},{14,1800},{1,1700},{9,1700},{11,1700},{18,1700}}

G2={{2,1600},{4,1600},{13,1600},{3,1200},{10,500},{6,400}}

G3={{15,400},{8,300},{12,300},{17,300},{0,200},{7,100}}

G4={{16,100}.

D. Virtual Machine Selection

After the creation of path set all the cloudlet groups are required to be allocated to the virtual machine paths. The groups of cloudlets will be allocated sequentially like first group to first leaf path (from root to leaf node), second group to second leaf path and so on. Now the elements of each group will be submitted to each virtual machine of each path in round robin fashion. Final result can be represented as Fig. 2.

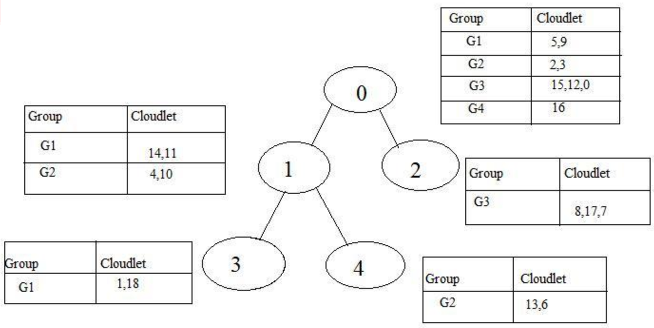
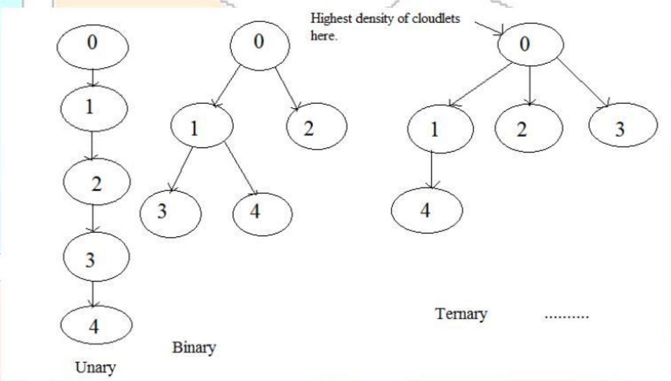


Fig. 2. Binary VM tree after allocation of cloudlets

After allocating all the cloudlets to the virtual machines the cloudlets are executed in the cloud. By the use of this methodology this can be understood that the main purpose for using this technique is to give more number of tasks to the virtual machines having higher capabilities of processing with a kind of distribution in which the larger and tasks from multiple ways comes to this.



Number of cloudlets assigned to the upper level increases…..

Fig. 3. Unary, Binary, Ternary trees for five VMs

V. CONCLUSION

We proposed a tree like structure for the execution of cloudlets for fairly distributing the cloudlet on the virtual machines. Experiments were conducted on unary, binary and ternary trees to see the effect of variation in the virtual machine capabilities. The future plan is to implement it in a real world cloud infrastructure. Here only execution time is considered to enhance the performance. In the future work other resource parameters will also be considered for better performance and efficient utilization of the resources.

.

REFERENCES

[1] Cloud computing basics.

[2] M. Xu, L. Cui and H. Wang, "Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing," In Proceedings of IEEE International Symposium on Parallel and Distributed Processing with Applications, 2009, pp. 629-634

[3] Ali Babar, M. and Chauhan, A., Toward a Framework for Migrating Software Systems to Cloud Computing, Tech Report TR-2011-139, IT University of Copenhagen, Denmark, 2011.

[4] S. Long, Y. Zhao,"A toolkit for modeling and simulating cloud data storage: an extension to CloudSim", In International Conference on Control Engineering and Communication Technology, 2012

[5] R. Achar, P.S. Thilagam, Shwetha, Pooja, Roshni and Andrea, "Optimal Scheduling of Computational Task in Cloud using Virtual Machine Tree" In Third International Conference on Emerging Applications of Information Technology (EAIT), 2012

[6] S. Selvarani and G. S. Sadhasivam, "Improved Cost-Based Algorithmfor Task Scheduling in Cloud Computing," In Proceedings of IEEEInternational Conference on Computational Intelligence and ComputingResearch, 2010, pp. 1-5

[7] G. Guo-Ning and H. Ting-Lei, "Genetic Simulated Annealing Algorithm for Task Scheduling based on Cloud Computing Environment," In Proceedings of International Conference on Intelligent Computing and Integrated Systems, 2010, pp. 60-63

[8] Y. Yang, Kelvin, J. chen, X. Lin, D.Yuan and H. Jin, "An Algorithm in SwinDeW-C for Scheduling Transaction Intensive Cost Constrained Cloud Workflow," In Proceedings of Fourth IEEE International Conference on eScience, 2008, pp. 374-375

[9] Akshi Bhutani, Isha Jauhari, Vinay Kumar Kaushik Department of Computer Science and Engineering Academy of Business and Sciences Engineering College Ghaziabad, India "Optimized Virtual Machine Tree based Scheduling Technique in Cloud Using K-way Trees"

[10] S. Liu, G. Quan and S. Ren, "On-line Scheduling of Real-time Services for Cloud Computing," In Proceedings of 6^{th} World Congress on Services (SERVICES-1), 2010, pp. 459 464

[11] S. Selvarani and G. S. Sadhasivam, "Improved Cost-Based Algorithm for Task Scheduling in Cloud Computing," In Proceedings of IEEE International Conference on Computational Intelligence and Computing Research, 2010, pp. 1-5.

[12] Ashwani Kumar Yadav, Hardwari Lal Mandoria" Study of Task Scheduling Algorithms in the Cloud Computing Environment: A Review" Ashwani Kumar Yadav et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 8 (4) , 2017, 462-468

[13] Somayeh Taherian Dehkordi, Vahid Khatibi Bardsiri" TASA: A New Task Scheduling Algorithm in Cloud Computing" Journal of Advances in Computer Engineering and Technology, 1(4) 2015

[14] Pirtpal Singh" A Review: Cloud Computing using Various Task Scheduling Algorithms" International Journal of Computer Applications (0975 – 8887)Volume 142 – No.7, May 2016

[15] Saleh Atiewi, Salman Yussof and Mohd Ezanee" A Comparative Analysis of Task Scheduling Algorithms of Virtual Machines in Cloud Environment "2015 Saleh Atiewi, Salman Yussof and Mohd Ezanee. This open access article is distributed under a Creative Commons Attribution (CC-BY) 3.0 license

[16] Ali Babar, M. and Chauhan, A., Toward a Framework for Migrating Software Systems to Cloud Computing, Tech Report TR-2011-139, IT University of Copenhagen, Denmark, 2011.

[17] S. Long, Y. Zhao,"A toolkit for modeling and simulating cloud data storage: an extension to CloudSim", In International Conference on Control Engineering and Communication Technology, 2012

[18] Raghavendra Achar, P. Santhi Thilagam, Shwetha D, Pooja H, Roshni and Andrea"Optimal Scheduling of Computational Task in Cloud using Virtual Machine Tree" 2012 Third International Conference on Emerging Applications of Information Technology (EAIT)