International Journal on Applications in Information and Communication Engineering
Volume 4: Issue 3:September 2018, pp 6 - 10. www.aetsjournal.com ISSN (Online) : 2394 - 6237

DYNAMIC RESOURCE ALLOCATION IN HETROENEOUS CLOUD COMPUTING

S. KRISHNANARAYANAN , D.KALEESWARAN

Abstract— Major cloud computing companies have started to integrate frame-works for parallel data processing in their product portfolio, making it easy for customers to access these services and to deploy their programs. However, the processing frameworks which are currently used stem from the field of cluster computing and disregard the particular nature of a cloud. As a result, the allocated compute resources may be inadequate for big parts of the submitted job and unnecessarily increase processing time and cost. We study the multi-resource allocation problem in cloud computing systems where the resource pool is constructed from a large number of heterogeneous servers, representing different points in the configuration space of resources such as processing, memory, and storage. DRFH also ensures some level of service isolation among the users. As a direct application, we design a simple heuristic that implements DRFH in real-world systems. Large-scale simulations driven by Google cluster traces show that DRFH significantly outperforms the traditional slot-based scheduler, leading to much higher resource utilization with substantially shorter job completion times.

I. INTRODUCTION

Cloud computing allows customers to scale up and down their resources based on needs. Cloud computing technology makes the resources as a single point of access to the client and cost is pay per usage. Cloud computing is a computing technology, where a pool of resources are connected in private and public networks and to provide these dynamically scalable infrastructure for application. Cloud computing is not application oriented and this is a service oriented. It offers the virtualized resources to the cloud users. Cloud computing provide dynamic provisioning and thus can allocate machines to store data and add or remove the machines according to the workload demands. Cloud computing platforms such as, those provided by Microsoft, Amazon, Google, IBM. Cloud computing is an environment for sharing resources without the knowledge of the infrastructure and can makes it possible to access the applications and its associated data from anywhere at any time.

Computational world is very broad and complex. In this respect, cloud computing has undertaken almost entire space. Basically, cloud is a collection of resources (hardware and software) distributed at worldwide datacenters. There are many servers available at various datacenters which are provided by service providers throughout the world. We are paying as per our demand for using those resources. There are many popular issues for research in cloud computing like virtualization, data security, license management, scalable storage management, mobile cloud, availability of services, task scheduling. But, scheduling of job is always a prime topic of research in cloud computing. There are heterogeneous resources available at various datacenters. So, traditional scheduling algorithms like FCFS, shortest job first, round-robin and priority etc, are not recommendable. The various challenges of task scheduling in cloud environment are:

1) To allocate resources to task.
2) To decide in which order the cloud should execute the task.
3) To schedule overhead when VMs prepare, terminate or switch task, communication overhead should be minimum.
4) It requires continuous VM status monitoring.
5) Cost of using VMs, dispatching VMs to different tasks etc.

Since scheduling world is very dynamic, heterogeneous and complex in cloud computing, we require some efficient scheduling technique that can optimize and improve the overall performance of scheduling system. Scheduling technique should be such that can do well and provide complete satisfaction at user end, at service provider end and also load balancing at system end.

In cloud computing, scheduling of task is done at two levels:

S. krishnanarayanan , Department of computer Science and Engineering , Arjun College of Technology, Coimbatore. (Email ID: krishna.narayanan08@gmail.com)

D.Kaleeswaran, Department of computer Science and Engineering , Arjun College of Technology, Coimbatore. (Email ID: kaleeswaranme@gmail.com)

1) Resource level scheduling, deployment of VMs at available physical nodes is done
2) User level scheduling, tasks are assigned to VMs

Customer put QoS constraints like minimum cost and task done faster etc. CSP (Cloud Service Provider) requires maximum returns on investment. At system level, maximum resource utilization and load balancing is required. In cloud computing, Broker acts as an intermediator between user and CSP. Broker exists (as shown in fig.1) at system level, the broker decides where to map job or task submitted by user to the resource provided by CSP [2]. So, while designing any new scheduling algorithm all the changes are performed at DCB (Data Center Broker). Likewise, many researchers have provided many scheduling algorithms, which are working well in one or the other way.

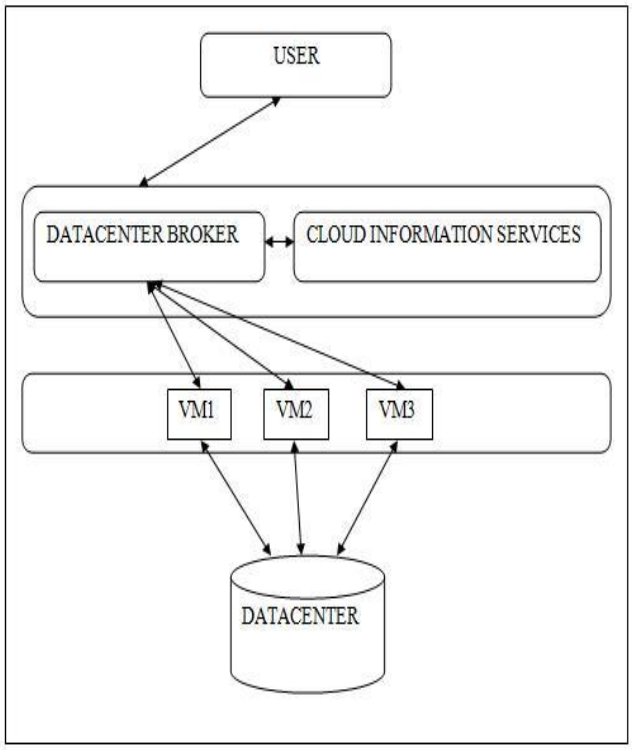

Figure1. Scheduling in Cloud

But our proposed scheduling algorithm for task scheduling is highly improved and efficient which is based on concept of "Weighted Fair Queuing" technique to improve quality of service. It has removed the drawbacks of already existing priority based task scheduling algorithms, i.e. starvation of low priority queues (long waiting priority queue). Now, with our proposed algorithm, there occur no long waiting priority queues. It provides fairness at priority level by implementing combination of priority queue and round-robin fashion scheduling at grouped cloudlet level. This improved algorithm proves to be beneficial to all, (resource manager and load balancing) and satisfying QoS constraints at each level. The remainder of the paper is sectioned as follows: section 2 discusses literature review, section 3 discusses proposed methodology for cloudlet scheduling, section 4 discusses proposed algorithm, section 5 shows experimental data and results and finally section 6 concludes the overall study.

II. RELATED WORKS

Grouping task for scheduling after prioritization effectively reduces processing time in comparison to task scheduling without grouping. Cost of task scheduling is further reduced, if VM is selected dynamically on basis of cost and processing power [1]. Average turnaround time and average cost of overall task scheduling is minimized, when turnaround time and cost of each job is minimize individually. As a result, number of tasks increases, which improves the performance [2]. This paper has focused on grouping task, prioritization and greedy resource allocation. Criteria for calculating cost of every task must not be same, as some tasks are simple, some tasks are complex. Different task have different CPU requirement, memory requirement etc. So, activity based costing is better way of calculating cost of each task, which measures cost of objects and performance of activities and computes cost more accurately [3].Taskexecution cost can be reduced and user required QoS is improved using load balancing at resource level scheduling [4].ERUA algorithm [5] satisfy user and cloud service provider through dynamic resource management where utilization ratio must fall under 1, leading to better resource utilization. This paper focus at resource level scheduling. Processing each job individually increases communication cost and time. Because of this communication overhead overall performance of task scheduling increases. But, job-grouping technique groups the small scaled user jobs in job groups which reduces overhead communication time [6].

There are different levels of elasticity structures offered by different cases of data flow structures, operator characteristics and other parameters etc for data flow schedule optimization on cloud [7]. Multiple QoS constraints based scheduling strategy to address multiple workflows in a decentralized cloud computing environment including two level task scheduling mechanism based on load balancing in cloud computing[8].This task scheduling mechanism not only meet user's requirements, but also get high resource

utilization. Priority based dynamic resources allocation to tasks scheduling algorithm, which considers multiple SLA objective of job, by preempting best-effort job in cloud environment is described in [9].In paper [10], it has been discussed that hierarchical scheduler exploit the multicore architecture for effective scheduling They have used diversity of task priority at local and global level for proper load balancing across heterogeneous processors.

TDP algorithm is there, where 'T' stands for task selection, 'D' for deadline and 'P' means priority in terms of cost, which selects task according to its constraints and requirements, finally scheduling is done using single priority queue [11].In heterogeneous environment of resources, the turnaround time of each job is minimized individually, to minimize the average turnaround time of all submitted jobs in a timeslot [12].

Though we have so many scheduling algorithms available, still some algorithm are better in one way or some are better in other way, none of them is completely efficient. Many algorithms make use of priority, but they all have disadvantage of long waiting queues. Generally, what happens in earlier proposed algorithms, they classify tasks as cost-based or deadline –based and then apply simple priority queues. In deadline-based grouping, task with higher deadline, having lower priority wait for longer duration for its execution, though it arrived so early. Also, in cost-based scheduling, the task with lesser execution cost (lower in priority) have to wait for longer duration for its execution. But, our proposed algorithm has included all good points of existing algorithms with new and enhanced version of priority that has surely remove the deficiency of all existing priority based task scheduling. We have added weighted fair queue to introduce priority of fairness in our proposed task scheduling algorithm.

The concept of "Weighted Fair Queue" is taken from the book William Stallings "High-Speed Networks and Internet, Performance and Quality of Service, Pearson education", under the topic "Scheduling", which is a Technique to improve QoS.

III. THE PROPOSED METHODOLOGY

Our proposed scheduling method includes following concepts:

Constraint-based grouping of tasks: - Groupingmeans collecting similar kind of tasks altogether in one dimension, all of them having same requirements. Task grouping can have many types like; grouping of tasks can be based on length of task, location of task, deadline of task, cost of task, complexity of task etc. Here, we have included two-dimensional grouping of task. One dimension consist of constraint imposed by user i.e. Deadline based task grouping. Actually, almost every user wants his request to be met as earlier as possible, so applying deadline constraints over tasks will be quiet beneficial for user. Second dimension satisfy constraint of cloud service provider (CSP) which is, maximum resource utilization and earning maximum profit i.e. cost-based task grouping. From business point of view, applying cost-based constraint on tasks submitted by user will allow CSP to mend more money. As, CSP will his use minimum cost machine to execute maximum length task, so he is getting more money by utilizing benefit from minimum cost machine. Grouping of tasks certainly reduces communication overhead. For example, if 1000 tasks are given for execution, suppose 680 tasks are deadline-based and rests are cost-based. Then, to check for its constraint (deadline-based or cost-based) separately for each task at runtime would increase overhead for system. So, it is better to group them previously according to their constraints, before arranging them in different priority list.

Weighted Fair Priority queue :- Among all theavailable scheduling methods, Weighted Fair Queue is best scheduling technique where tasks(cloudlets) are assign to different priority queues for scheduling. Weighted Fair Queue Model is shown in fig 2; in these queues are weighted based on priority of queue. Selected VM process task in each queue based on round-robin fashion where number of tasks selected for execution from each queue depends on its queue weight. For example, let weight assigned to high priority queue is 3, weight of mid queue is 2 and weight granted to low priority queue is 1.then at each cycle(round) 3 tasks are processed from high priority queue, 2 tasks from mid priority queue and 1 task is executed from low priority queue. In this way, tasks would be executed in round-robin fashion and priority fashion scheduling, both method goes together side-by-side. So, there will be no long waiting low priority queues.

Greedy resource (VM) allocation: - This approachis greedy in respect that it selects resource, which appears best at instant. It means scheduler or broker selects VM with minimum turnaround time for each individual task for deadline-based task scheduling. Minimizing turnaround time for each job will definitely reduce overall turnaround time and increase response time for all task taken together. This is a great enhancement regarding system performance, providing benefit at system level. For cost-based tasks, it selects VM with minimum cost and accurate processing power for cost-

based tasks. The task with highest cost (which is decided based on task length) is assign to VM with minimum possible cost. This reduces cost of execution of each task. Here, cost refers to CSP's minimum cost machine is best utilized to execute the maximum length task of the user and providing maximum money benefit to him. So, he is spending less and earning more. During dynamic optimization, Greedy allocation of resource searches local optima and finally reaches global optima. Continuous VM-status monitoring is done by calculating waiting time and then updating turnaround time of respective VM at each VM selection. Here, figure 3 is depicting the model of proposed algorithm, which is making use of weighted fair queue i.e. shown in figure 2.

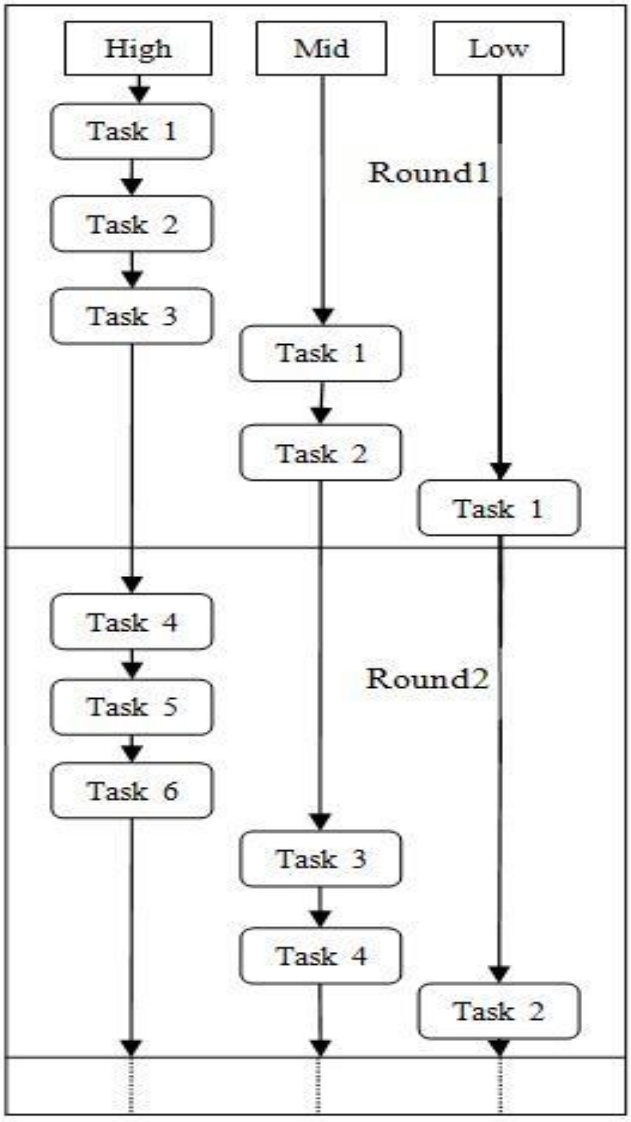

Figure 2. Weighted Fair Queue Model

Making use of this best scheduling technique surely enhances the degree of multiprogramming by making each VM busy executing at least one task at a time, definitely CPU utilization increases and response time decreases for each task in particular. Hence, it is three way optimization technique, as user is getting his task executed faster and Cloud Service Provider is getting maximum benefit at cost of machine level. Also resource utilization i.e. VM utilization is at best promising level.

IV. EXPERIMENTAL DATA ANALYSIS

The CloudSim toolkit is used demonstrate the simulation. The simulation results are verified using CloudSim (2.1.1) to check the correctness of proposed algorithm [2]. The simulation results of proposed algorithm are compared with the Sequential assignment, which is in-build in CloudSim and Dynamic Optimization Algorithm for Task scheduling [2].

Table 1. Results of Proposed Algorithm and Existing Algorithm

No.of cloudlets	Sequential algorithm	Dynamic Optimization Without fair priority	Proposed Algorithm With fair priority
20	98.6154	67.6984	45.3135
40	417.5499	283.441	235.2001
60	1041.3430	539.131	485.3890
80	1953.1897	885.0639	716.0989
100	3103.9777	1486.6538	1166.1269

Comparing proposed algorithm with sequential i.e. FCFS scheduling algorithm and Dynamic optimization scheduling algorithm [2] shows the tremendous improvement in results. As the number of cloudlets are increasing definitely, the total execution time has decreased together for both deadline and cost based tasks. The result shown above are the average of total execution time obtained after several number of execution for each number of cloudlet (e.g. we have run the implementation 20 times for 100 cloudlets and calculated its average).

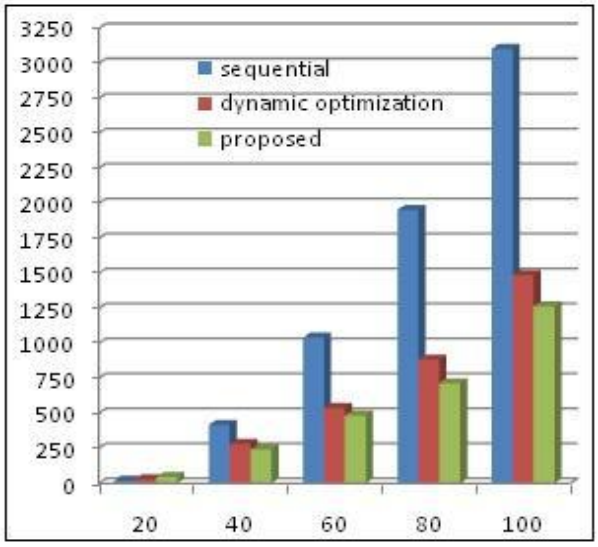

Figure 3 : Traditional V/S Proposed Algorithm With Respect To Time

The above bar graph showing task completion time and comparison of traditional algorithm and proposed algorithm.

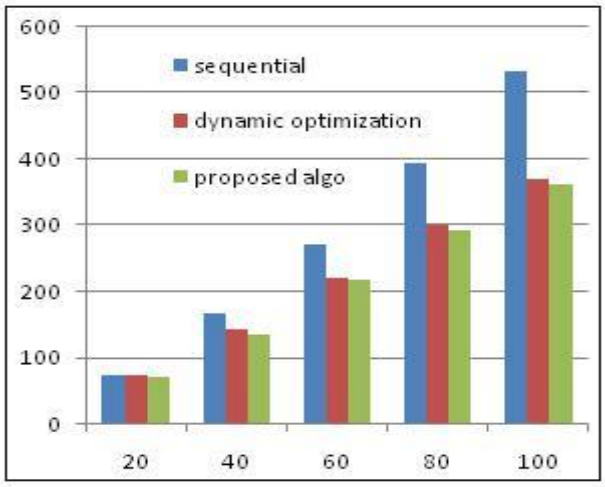

Figure 4 : Cost Comparison Traditional V/S Proposed Algorithm With Respect To Cost

V. CONCLUSIONS AND FUTURE WORK

Scheduling of task in cloud environment is a highly researched and challenging issue in cloud computing. To meet thousand of service requests while making best possible use of available resources and simultaneously satisfying both user as well as service provider , is the challenge for task scheduler. Traditional methods of scheduling lead to overpricing and slow processing for bulk of tasks. Some task scheduling algorithm is cost based, some are deadline based and many algorithms make use of priority based scheduling. But they suffer from long waiting priority queues .Our proposed algorithm definitely meet all the challenges, along with constraint based optimization scheduling. We have also, introduced fair-priority scheduling concept i.e. combination of priority with round-robin scheduling scheme. This brings fairness at priority level and increases utilization of resources at system level and thereby providing much more efficient results than that can be provided by any other existing task scheduling algorithm.

References

[1] Yogita Chawla and Mansi Bhonsle, "Dynamically optimized cost based task scheduling in CloudComputing", International Journal of Emerging Trends & Technology in Computer Science, Volume 2, Issue 3 May-June 2013.

[2] Monika Chaudhary and Sateesh Kumar Peddoju, "A Dynamic Optimization Algorithm for Task Scheduling in Cloud Environment" International Journal of Engeenering Research and Application, Vol 2, Issue 3, May-June 2012.

[3] Qi Cao, Zhi-bo Wei and Wen-Mao Gong, "An optimization algorithm for task scheduling based on activity based costing in cloud computing", Wuhan University, China 978-1-4244-2902-8/09/$25.00 2009 IEEE.

[4] Hong Sun, Shi-ping Chen, Chen Jin and Kai Guo, "Research and simulation of task scheduling algorithm in cloud computing" University of Shanghai, China, July 25, 2013.

[5] Ram Kumar Sharma and Nagesh, Sharma, "A Dynamic optimization algorithm for task scheduling in cloud computing with resource utilization,"IJSET@2013 volume no.2, Issue no 10, pp: 62-68, year 2013.

[6] Nithiapidary Muthuvelu, Junyang Liu, Nay Lin Soe,Srikumar Venugopal, Anthony Sulistio and Rajkumar Buyya, "A Dynamic job grouping-based scheduling for deploying applications with fine-grained tasks on global grids" The University of Melbourne, Australia ICT building, 111 barry street, carlton VIC 3053,2005.

[7] Herald Kllapi, Eva Sitaridi and Manolis M. Tsangaris, "Schedule optimization for data processing flows on the Cloud", University of Athens, copyright 2011.

[8] Vandana Choudhary, Saurabh Kacker, Tanupriya Choudhray and Vasudha Vashisht, "An Approach to improve task scheduling in a decentralized cloud computing environment", IJCTA| jan-feb 2012.

[9] Chandrashekhar S.Pawar and Rajnikant B. Wagh, "Priority based dynamic resource allocation in Cloud computing", Shirpur, India, 2011.