Optimization of Cloud Task Scheduling through Multi-Objective Task Grouping Techniques

Priyanka. S.*, Dr. Venkatesan. R, Deepa. K, Sivaranjani. M

Assistant Professor (SRG), Department of Information Technology, Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu
Professor, Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, Tamil Nadu
Assistant Professor (SLG), Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, Tamil Nadu
Assistant Professor, Department of Information Technology, Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu

KEYWORDS: Task grouping, Cloud, Scheduling, QoS.

ABSTRACT
In recent years, the information communication technology (ICT) appeared new paradigm of utility computing called cloud computing. The consumer cloud is always important of high performance for cloud computing service and satisfy service agree level (SLA). In cloud computing, there is a need of further improvement in task scheduling algorithm to group of tasks, which will reduce the response time and enhance computing resource utilization. This grouping strategy considers the processing capacity, memory size and service type requirement of each task to realize the optimization for cloud computing environment. It also improves computation/communication ratio and utilization of available resources by grouping the user tasks before resource allocation. The experimental results were conducted in a simulation cloud computing environment. The results show that the task grouping improves the CPU utilization by reducing the makespan when compared to Multi objective task scheduling.

INTRODUCTION
Within few years cloud computing grab the IT market very fast and most of the IT industry start using the cloud computing. In cloud computing the word cloud refers as internet, so the meaning of cloud computing is Internet Based Computing. In other words it‟s a kind of server based computing. Cloud computing provide on demand services to the client. The services includes SaaS (Software as a service) where application software and database access provided to the user pay per use basis, IaaS (infrastructure as a Service) where virtual machine provided to the user using virtualization of physical machine which includes processing power, storage and other resources, PaaS (Platform as a Service) where cloud provider provides a computing platform which includes OS, programming language execution platform and web server.

Growth of cloud computing slower down the efficiency, throughput and utilization of resources for which cloud computing need to be evolved. The scheduler retrieves information of the processing capability and memory size for computing resource. Then, the scheduler selects the appropriate computing resource target to groups based on processing capability and memory size for resource in order to attain the reduce for response time of task execution. Thus, improves computation/communication ratio and utilization of available resources. The CloudSim toolkit has been used to test the task grouping and scheduling. The toolkit, a java-based discrete-event cloud computing simulation package, supported both system and behavior modeling of cloud computing system components such as data center and virtual machines (VMs). Mapping of task to resource and resource management are also supported.

MATERIALS AND METHODS
Cloud computing service providers have several datacentres in order to optimally serve customer needs around the world. However, existing system does not provide the proper scheduling of customer requested application among the VMs in datacenters to achieve reasonable QoS levels. Every datacenter in cloud computing consist of numerous servers and each server runs numerous VMs. Each VMs have different capability to execute different QoS's tasks requested by the customer.

Cloud broker is responsible for mediating negotiations between SaaS and cloud provider and such negotiation are driven by QoS requirements. Broker acts on behalf of SaaS for allocation of resources that can meet application's QoS requirements.

Assigning QoS for Tasks and VMs

Cloud broker sends request to the cloud service provider for the QoS of requested task. In proposed task scheduling algorithm task"s priority is assigned according to the QoS. High QoS task assigned with low QoS value and the low QoS task assigned with high QoS value. Hence, the task with lower QoS value is a high priority and the task with high QoS value is a low priority. The QoS for tasks are documented in SLA. Task's QoS value is associated throughout its life cycle. Also cloud broker sends request to the cloud service provider for list of VMs created in the datacenters. After receiving the list of VMs cloud broker assigns QoS to the VMs. Millions of instructions per second (MIPS) of a VM is considered for assigning VMs QoS. VM with high MIPS is a high QoS VM and VM with low MIPS is low QoS.

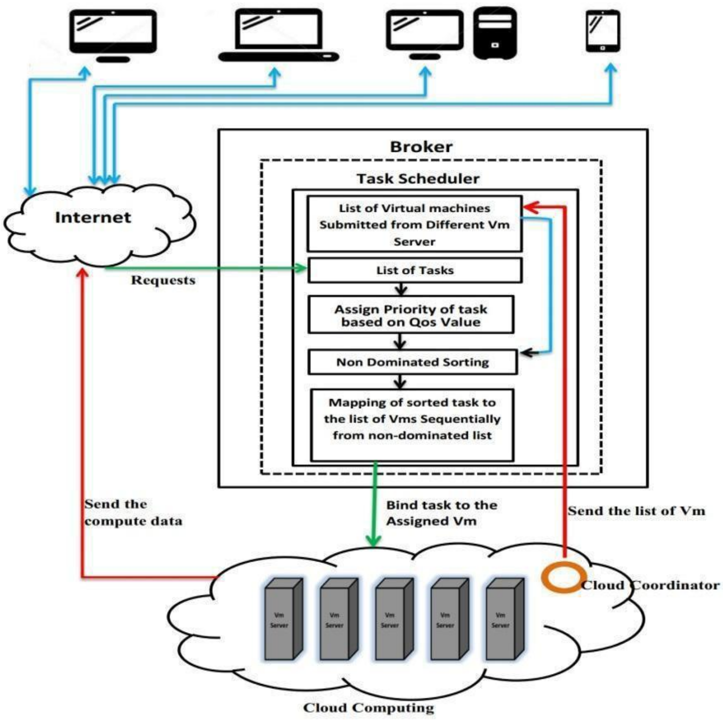

Figure 1. Multi-objective Task scheduling

Non-dominated sorting

A non-dominated sorting [8, 5] is used to solve the multi -objective problems. In multi - objective problems multiple objective functions are considered. In the proposed work, the main goal is to minimize the execution time of a task. Main goal is achieved by selecting a task with minimum task size and minimum (low)QoS value. The two objective functions are as follows.

Minimize $f(S_k) = S_k|$ €j i, $f(S_i) \leq f(S_j)$ Minimize $f(Q_k) = Q_k|$ j i, $f(Q_i) \leq f(_{Qj})$ S, Q T (ID, Q, S)

$i = \{1, 2, 3....n\}$, $j = \{1, 2, 3....n\}$, $k = \{1, 2, 3....n\}$

i, $f(S_i) \leq f(S_j)$ (1)
i, $f(Q_i) \leq f(Q_j)$ (2)

Where, S is size of the task and Q is the task QoS value, T is the set of task and n is the Number of task.A non-dominated sorting used to implement the multi-objective task scheduling algorithm with the above objective In non-dominated sorting, multiple objectiv are applied at a time.

Virtual Machine Selection

Cloud broker maintains a list of VM for VMs. This list is updated in fixed time interval and dynamically at the pick time when number of requests increase suddenly.According to the MIPS of VM, list of VMs is sorted in descending order. VM in the first position of the list have high QoS and at the end of the list low QoS VM. Afternon-dominated sorting finally generated non-dominated task"s set is bound with theVMs. In the process of binding or allocation of VM to a task, it is done sequentiallyaccording to both the list of tasks and VMs. The first VM from theVM"s list to thefirst task in the task"s list and second VM in the VM"s list with second task in the task"s list. Once the allocation reached the last VM, the next task will be submitted once again to the first VM of the VM‟s list and the process of allocation will be repeated for all tasks.

Algorithm 1 : Multi-objective task scheduling algorithm

```
1. Submit both VMs list of successfully created VMs in datacenter and task list to
Broker.
2. Create a received list of tasks.
3. Create a received list of VMs.
4. Non-dominated sorting (list of task)
  i ← 0
  Create empty non-dominated list
  dominated list ←list of task
  Initially put task_i in the non-dominated list
  for all i ← 1 to size of task's list do
    for all j ← 0 to size of non-dominated list do
      if task_j dominates task_i then
        put task_j into non dominated set
      else
        if task_i dominates task_j then
          put task_i into non dominated set
        end if
      else
        put task_i and task_j into non dominated set
      end if
    end for
  end for
   ---- ---
   5. Sort the list of task according to the non-dominated task set.
   6. Sort the VM received list in descending order .
   7. j ← 0 7.
   for all i ← 0 to the size of task's list do
     if j ≥ 0 then
       Bind task_i to the VM_j j++
       if j== number of VMs then
         j=0
       end if
     end if
   end for
```

Architecture of task – grouping schedule

In this section, some reprehensive research works on task grouping and scheduling in distributed computing system and cloud computing environment have been surveyed. Grouping- Based job scheduling model in grid computing, group jobs according to MIPS, memory size and bandwidth of the resource. This model reduces the processing time of jobs utilize grid resources sufficiently, network delay to schedule and execute jobs on the grid due to this study presented and evaluation an extension from computational-communication to computational communication-memory based grouping job scheduling strategy, but the algorithm doesn‟t parallel schedule resource. Scheduling framework for bandwidth-aware job grouping-based scheduling in grid computing, group jobs according to MIPS and bandwidth of resource. This model reduces processing time of jobs compare to a non bandwidth-aware job grouping scheduling framework. Dynamic job grouping-based scheduling for deploying application group jobs according to MIPS of resource only. This model reduces processing time, communication time of jobs and cost. In the same way, improve activity based cost algorithm for cloud computing. Before group tasks scheduling, tasks are sorted according to their priority and they are place in three different list based on three levels of priority. They are high, medium and low priority. This model according MIPS of resource only when the scheduler is grouped task. This algorithm improved the computation-communication ratio. It also is minimization of makespan and cost compare to activity based cost algorithm.

There are task schedule, computational server, storage server, task group and selection resource, information collection, cloud information services, dispatcher, resource. In architecture of task-grouping schedule model are using basic of grid model and task scheduling. The scheduler accept task for request user with SLA parameter such as service type, size of file and scheduler is classification task users request based on service type. The services types are two major in cloud computing, there are storage service and computational service. The storage server accepts tasks for storage service where storage server provided data storage and it don't require mapping of services where storage of data doesn't need map resource with task request. The computational server accepts tasks for computational services where provides mapping between tasks and computing resource based SLA parameters. Begin the schedule task grouping by grid model which contain of five basic blocks are task grouping and selection resource, information collection, cloud information service (CIS), dispatcher and resource.

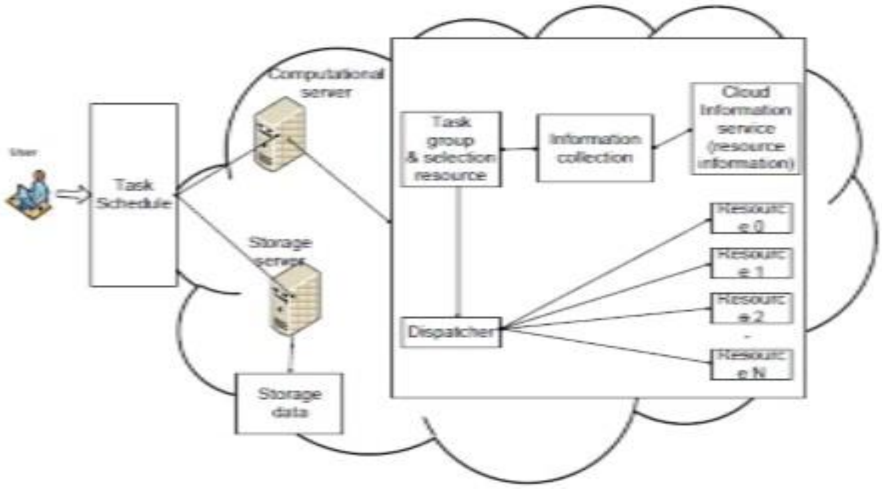

Figure 2. Architecture of Task Group Scheduling Model

It collects available of the computing resource and get characteristic resource such as processing capability (MIPS) and memory size to each available of resource through cloud information service. The task group and selection resource used for information collector to gather necessary information resource to perform task selection via required for information such as MIPS, memory size. the dispatcher functions as sender where sends grouped task t their appropriate resource based on the schedule model during the mapping of tasks with computing resource and it gathers the results of the processed tasks from the resource. The resource functions execute of group tasks and send to result for user.

TASK – GROUPING ALGORITHM

Step 1: The n number of tasks to be scheduled and the m number of resources to be provided are received by the scheduler.
Step 2: All the available tasks are submitted to the scheduler.
Step 3: Set tot_len=0, Set resource_id j=1 and index i=1.
Step 4: Get the MIPS of the resource j.
Step 5: Get the length (MI) of the task from the list.
Step 6: If the resource MIPS is less than task length then,

6.1 The task cannot allocated to the resource.
6.2 Get the MIPS of the next resource.
6.3 Goto step 4.

Step 7: If the resource MIPS is greater than task length then repeat
while (tot_len<=resource MIPS) and there exist ungrouped tasks in the list.

7.1 tot_len=previous total length + current task length.
7.2 Get the length of the next task.

If the total length is greater than resource MIPS, then subtract length of the last task from the tot_clen.
End while.
Step 8: If ungrouped tasks are available, then perform multi objective task scheduling algorithm.

RESULTS AND DISCUSSION

In order to obtain results of the proposed algorithm the simulation was done using CloudSim 3.0.2 Simulator..In Our simulation scenario, the proposed algorithm is compared to the existing task scheduling algorithm, for this purpose following illustrative example is taken. We have created many VMs and tasks with different task size. Task size ranges from 50 to 250 and the QoS value ranges from 0-800.

In Figure 3, X-axis represents number of tasks and the Y-axis represents the makespan of the task. When the number of cloudlets is 50 and the Virtual machine is 5, Multi-Objective completes the task in 411.104 sec whereas proposed system completeS the task in 175.29 sec. As a result, the makespan is reduced by using task grouping.

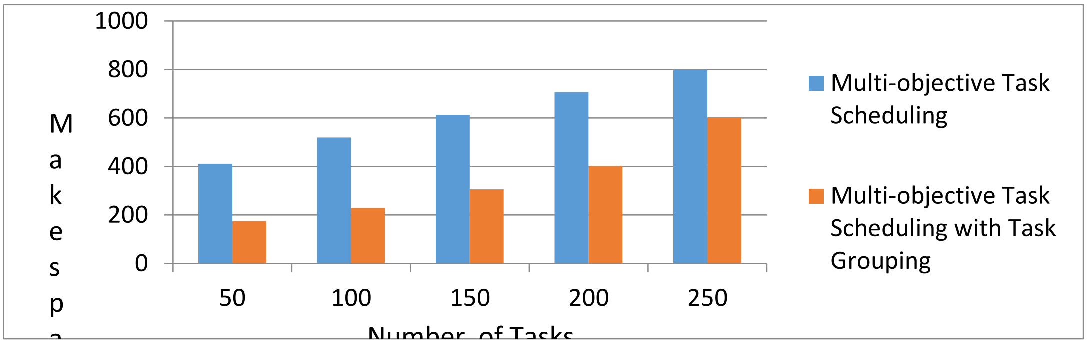

Figure 3. Comparison Chart of Makespan

CONCLUSION

Cloud computing is a distributed computing which mainly focuses on providing services to the customers and it provides computational as well as storage resources to users. To utilize the resources efficiently, task scheduling provides the solution. Scheduling is the process in which the tasks are assigned to the VM's. In cloud computing environment, many algorithms are available to solve scheduling of tasks and resource allocation problems. Since scheduling of tasks in cloud computing is an NP-hard optimization problem, an efficient task scheduling strategy is required. The proposed task-grouping scheduling strategy aims at minimum total tasks completion time and maximum utilization of the cloud resources. It considers the processing capacity, memory size and service type requirement of each task to realize the optimization for cloud computing environment. The proposed scheduling strategy provides the better result compared to multi objective task scheduling.

In future, the work shall include some other QOS parameters for efficient scheduling of tasks and resources where the number of data center and user job changes dynamically.

REFERENCES

1. El-Sayed T. El-kenawy, Ali Ibraheem El-Desoky, Mohamed F. Al-rahamawy"Extended Max-Min Scheduling Using Petri Net and Load Balancing,"International Journal of Soft Computing and Engineering (IJSCE) ISSN:2231-2307, Volume-2, Issue-4, September 2012.
2. Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger and Dawn Leaf, " NIST Cloud Computing Reference Architecture" Special Publication 500-292, September 2011.
3. Jianfeng Zhao, Wenhua Zeng, Min Liu, Guangming Li; Min Liu, "Multiobjective optimization model of virtual resources scheduling under cloud computing and it‟s solution, "Cloud and Service Computing (CSC), International Conference on , vol., no., pp.185,190, 12-14, 2011
4. Kaiqi Xiong, Harry Perros, "Service Performance and Analysis in Cloud Computing", 2009 IEEE.
5. R. Buyya, C. S. Yeo, S. Venugopal, J.Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems", Elsevier Science, Amsterdam, The Netherlands, 2009 (pp. 599-616).
6. R.N. Calheiros, R. Ranjan, A. Beloglazov, C.A.F. De Rose, R. Buyya, Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, Software: Practice and Experience 41 (1) (2011) 23–50.
7. Sandeep Tayal , "Task Scheduling optimization for the Cloud Computing Systems",(IJAEST) International journal of advanced engineering sciences and technologies , Vol No. 5, Issue No. 2, 111 115.
8. ShalmaliAmbike, DiptiBhansali, JaeeKshirsagar, JuhiBansiwal ,"An Optimistic Differentiated Job Scheduling System for Cloud Computing , "International Journal of Engineering Research and Applications (IJERA)ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 2,Mar-Apr 2012, pp.1212-1214.
9. Shamsollah Ghanbari, Mohamed Othman "A Priority based Job SchedulingAlgorithm in Cloud Computing", International Conference on Advances Scienceand Contemporary Engineering 2012 (ICASCE 2012).
10. Wei-Tek Tsai ,Xin Sun, Janaka Balasooriya, "Service-Oriented Cloud Computing Architecture". Seventh International Conference on Information Technology, IEEE.2010.