iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** V **Month of publication:** May 2015

DOI:

Volume 3 Issue V, May 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

SAKTHI: Scheduling Algorithm K to Hybrid in Cloud Computing

Sakthivelmurugan.V[1], Rajkumar.K[2]
Department of Information Technology, PSNA College Of Engineering and Technology

Abstract— Cloud computing is based on the concepts of distributed computing, grid computing, utility computing and virtualization. It is a virtual pool of resources which are provided to users via Internet. These cloud computing environments provide an illusion of infinite computing resources to cloud users so that they can increase or decrease their resource consumption rate according to the demands. At the same time, the cloud environment meets a number of challenges. Cloud providers and cloud users are two important players in cloud environment to pursue different goals. The Cloud service providers want to maximize throughput by achieving high resource utilization, while users want to minimize expenses while meeting their performance requirements. However, it is difficult to allocate resources in a mutually optimal way due to the lack of information sharing between them. It introduces new scheduling algorithm name as SAKTHI which is abbreviated by Scheduling Algorithm K to Hybrid In Cloud.
Keywords— Distributed computing,Cloud providers,Cloud user,Resource Utilization,Hybrid Algorithm,Scheduling

I. INTRODUCTION

Cloud computing comes in focus development of grid computing, virtualization and web technologies. Cloud computing is an internet based computing that delivers infrastructure as a service (IaaS), platform as a service (PaaS), and software as services (SaaS). In SaaS, software application is made available by the cloud provider. In PaaS an application development platform is provided as a service to the developer to create a web based application. In IaaS computing infrastructure is provided as a service to the requester in the form of Virtual Machine (VM)[10][11].
SaaS is a model of software deployment where an application is hosted as a service provided to customers across the Internet. SaaS is generally used to refer to business software rather than consumer software, which falls under Web 2.0. By removing the need to install and run an application on a user's own computer it is seen as a way for businesses to get the same benefits as commercial software with smaller cost outlay. Saas also alleviates the burden of software maintenance and support but users relinquish control over software versions and requirements. The other terms that are used in this sphere include *Platform as a Service* (PaaS) and *Infrastructure as a Service* (IaaS). Several large Web companies (such as Amazon and Google)[1][2] are now exploiting the fact that they have data storage capacity which can be hired out to others. This approach, known as 'cloud storage' allows data stored remotely to be temporarily cached on desktop computers, mobile phones or other Internet-linked devices. Amazon's Elastic Compute Cloud (EC^2) and Simple Storage Solution (S3) are well known examples.
The use of the cloud provides a number of opportunities:
It enables services to be used without any understanding of their infrastructure.
Cloud computing works using economies of scale. It lowers the outlay expense for start up companies, as they would no longer need to buy their own software or servers. Cost would be by on-demand pricing. Vendors and Service providers claim costs by establishing an ongoing revenue stream.
Data and services are stored remotely but accessible from 'anywhere'.In parallel there has been backlash against cloud computing:
Use of cloud computing means dependence on others and that could possibly limit flexibility and innovation. The 'others' are likely become the bigger Internet companies like Google and IBM who may monopolise the market. Some argue that this use of supercomputers is a return to the time of mainframe computing that the PC was a reaction against.
Security could prove to be a big issue. It is still unclear how safe outsourced data is and when using these services ownership of data is not always clear.
There are also issues relating to policy and access. If your data is stored abroad whose FOI policy do you adhere to? What happens if the remote server goes down? How will you then access files? There have been cases of users being locked out of accounts and losing access to data.
Hybrid scheduling is presented which helps in achieving Service Level agreement with quick response from the service provider. In our proposed approach Quality of Service metric such as response time is achieved by executing the high priority jobs (deadline based jobs) first by estimating job completion time and the priority jobs are spawned from the remaining job with the help of Task Scheduler. Scheduling is the one of the most prominent activities that executes in the cloud computing

environment. To increase the efficiency of the work load of cloud computing, scheduling is one of the tasks performed to get maximum profit.

The main objective of the scheduling algorithms in cloud environment is to utilize the resources properly while managing the load between the resources so that to get the minimum execution time. Here we mainly discuss three scheduling algorithm First come first serve, Round robin scheduling and new scheduling approach is generalized priority algorithm. Hybrid scheduling achieves better throughput and good performance.

Cloud Services can also be used to hold structured data. There has been some discussion of this being a potentially useful notion possibly aligned with the Semantic Web [2], though concerns, such as this resulting in data becoming undifferentiated [3], have been raised.Scheduling in cloud computing can be categorized into three stages.

Discovering a resource and filtering them.

Selecting a target resource (Decision stage).

Submission of a particular task to a target resource.

II. RELATED WORK

In this section, we describe the related work ok task scheduling in cloud computing environment. Resource allocation and scheduling of resources have been an important aspect that affects the performance of networking, parallel, distributed computing and cloud computing. Many researchers have proposed various algorithms for allocating, scheduling and scaling the resources efficiently in the cloud.

A. First Come First Serve

FCFS for parallel processing and is aiming at the resource with the smallest waiting queue time and is selected for the incoming task. The Cloud Sim toolkit supports First Come First Serve (FCFS) scheduling strategy for internal scheduling of jobs. Allocation of application-specific VMs to Hosts in a Cloud-based data center is the responsibility of the virtual machine provisioned component. The default policy implemented by the VM provisioned is a straightforward policy that allocates a VM to the Host in First-Come-First-Serve (FCFS) basis. The disadvantages of FCFS is that it is non preemptive. The shortest tasks which are at the back of the queue have to wait for the long task at the front to finish .Its turnaround and response is quite low.

B. Round Robin Scheduling

Round Robin (RR) algorithm focuses on the fairness. RR uses the ring as its queue to store jobs. Each job in a queue has the same execution time and it will be executed in turn. If a job can't be completed during its turn, it will be stored back to the queue waiting for the next turn. The advantage of RR algorithm is that each job will be executed in turn and they don't have to be waited for the previous one to get completed. But if the load is found to be heavy, RR will take a long time to complete all the jobs. The Cloud Sim toolkit supports RR scheduling strategy for internal scheduling of jobs. The drawback of RR is that the largest job takes enough time for completion.

C. Min-Min Algorithm

Min-Min algorithm selects the smaller tasks to be executed first.

D. Max-Min Algorithm

Max-Min algorithm selects the bigger tasks to be executed first. In this paper we will discuss other scheduling algorithms i.e, Heuristic algorithms.

E. K Scheduling Algorithm

Customer defines the priority according to the user demand you have to define the parameter of cloudlet like size, memory, bandwidth scheduling policy etc. In the proposed strategy, the tasks are initially prioritized according to their size such that one having highest size has highest rank. K is the priority based approach.k will be increased as Client's request. It achieves only 80% throughput in cloud environment.

III.PROPOSED ALGORITHM

Hybrid Scheduling is a class of scheduling mechanisms that mix different scheduling criteria or disciplines in one algorithm. The algorithm is given below. It stores all suitable Virtual Machines in a VM List.Scheduler and provisioner play important role in cloud computing.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

A. Scheduler

Jobs that are accepted by the Admission Control are received by the Scheduler module, which makes decisions based on a number of factors such as the pool to which the idle resources belongs to and job priority and ownership. In order to prevent starvation of regular jobs, a minimum amount of resources to be made available for regular tasks can be defined. These resources compose the regular pool and its access is coordinated via a regular queue. The rest of the local machines belong to the deadline pool, whose accesses are coordinated via deadline queues. Finally, dynamically provisioned machines belong to external pools and are coordinated by external queues. Figure 3 depicts the organization of the resource pools and queues in the Scheduler. Tasks that compose submitted jobs are forwarded either to the regular queue or to one of the deadline queues (there is one of such queues for each resource that belongs to the deadline pool). They respectively store tasks without deadline constraints and tasks with such constraints. Tasks on each queue are rearranged every time a new job is received by the Scheduler and every time a task completes. A third set of queues, external is also present in the Scheduler. There is one of such queues for each user and it contains tasks that belong to jobs that require dynamic provisioning to complete before the deadline. Tasks on this queue execute preferentially in dynamically provisioned resources, as detailed later in this section. Algorithm details the procedure for building the regular queue. This procedure runs every time a new job is received and every time a new resource is added to this pool. In the case of the deadline pool, whenever a new job is received, tasks are scheduled to different resource queues following a policy such as Round Robin, Worst Fit, Best Fit, and HEFT [8]. We do not apply backfilling techniques to prioritize tasks with closer deadlines because it may motivate users to make late submission of jobs or to overestimate execution time of tasks (both situations that would increase priority of their jobs over others). Dispatching of tasks for execution depends on the pool that the idle resource belongs to. When a resource from the regular pool becomes idle, the task on top of the regular queue is dispatched for execution in such resource. If the regular queue is empty, the waiting task from deadline queues with the smallest lag time (which we define as the difference between the time to the deadline and the estimated execution time) is removed from its queue and dispatched for execution. Finally, if the deadline queue is also empty, the first task on the external queue is dispatched for execution. When a resource from the deadline pool becomes idle, the next task on its queue is dispatched. If the queue is empty, the task from other queues with the smallest lag time is removed from its original queue and dispatched[3]. If the deadline queue is empty, the first task in the external queue is dispatched or, if the queue is empty, the first task in the regular queue is dispatched. Whenever a resource from the external queue becomes available, the first task on the external queue that belongs to the user that required the resource is dispatched to the resource. If there is no such a task, a task from the user is sought in the deadline queue. The first task from the user whose estimated execution time is smaller than the time left before the end of the resource's billing period is dispatched. If no task from the user meets this condition, the first task from the user in the regular queue is dispatched.

B. Provisioner

The Provisioner makes decisions about utilization of public Cloud resources. It calculates the number of extra resources required to execute a job within its deadline and also decides if machines whose billing periods are finishing will be kept for another period or not. The required number of resources is defined at task level: tasks that belong to an accepted job that can run in the deadline pool before the deadline are scheduled locally. Tasks that cannot be completed on time are put in the external queue by the scheduler, and provisioning decision is made based only on such tasks. Currently, the provisioner assumes a single type of VM[5] to be provisioned. This increases the chance of successful allocation of Cloud resources because it enables acquisition of "reserved" or "pre-paid" resources. Most IaaS offer such type of resource, which guarantees that, whenever resources are required, they will be available, as users paid for them upfront or via a premium plus discounted rates for utilization[7]. Alternatively, the provisioner can register multiple providers, and use resources from another provider when the preferable one cannot supply the required resources. When a virtual machine is reaching the end of its billing period, the Provisioner decides whether the resource should be kept for the next billing period or if it should be decommissioned. This decision is based on the states of external deadline queues. The simplest case is when the external resource is idle or it is running a regular task. It happens when the other queues are empty. In this case, the resource is decommissioned by the Provisioner. A regular task running on the resource is rescheduled in the regular queue. If the provisioned resource is executing a deadline or external task, the resource is kept for the next billing period to avoid risk of missing the job's deadline. In the case that the resource is no more necessary for the user that originally requested it, and there are still external tasks in the queue, the resource is reassigned for the user that needs the resource (providing it has authorization and credit to use them). In this case, accounting responsibilities for the reassigned resource is also changed, as detailed next.

C. Hybrid Scheduling Algorithm

The job can be given through enqueue operation and the job can be released through dequeue operation.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

While(1) do
Begin
Dequeue:
Broadcast an item selected according to the RR Scheduling
Handle the requests occurring during push;
If the enqueue is not empty then
Dequeue:
Pop out from enqueue the item whose stretch is Maximum;
If equal
Pop the item whose sum of the client's priority is high;
Else
Pop the item with the smallest index;
Clear the number of pending request for this item, and delete it.
Handle the requests occurring during the pull queue.
End;
End While;

IV. EXPERIMENTAL EVALUATION

A. Experimental Setup And Comparison

Simulation experiments have been carried out using Cloudsim.The scheduler or job manger manages the job sent from the client and sends to the worker.

TABLE : EXECUTION TIME COMPARISON

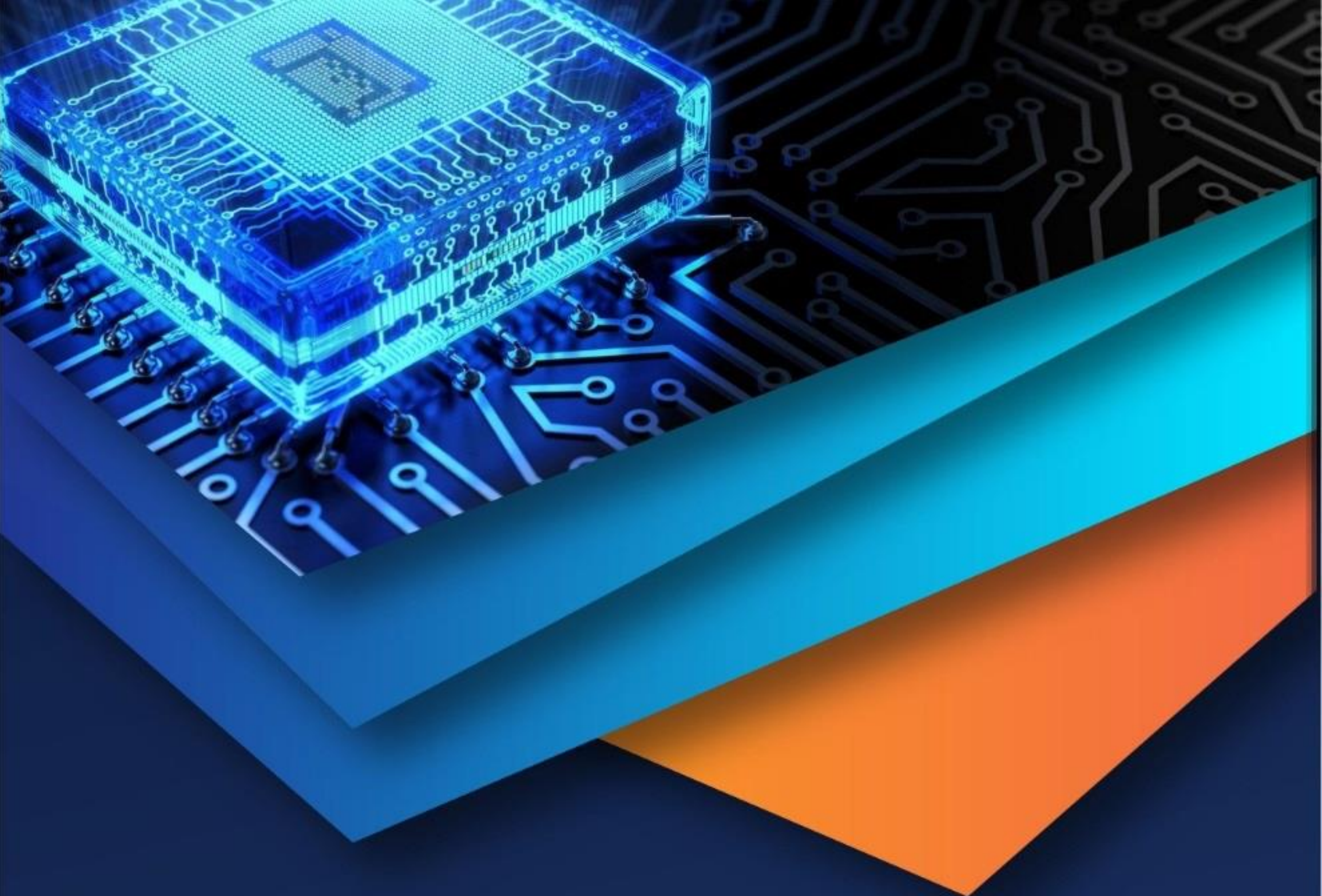
NO. OF JOBS	PROCESSING TIME		
	Hybrid algorithm	FIFO algorithm	RR algorithm
Job1	1.845	2.552	2.205
Job2	1.727	1.772	1.766
Job3	2.929	3.352	3.312
Job4	2.800	3.375	3.300
Job5	2.702	2.296	2.245

V. CONCLUSIONS

The Cloud providers and cloud users are two important players in cloud environment to pursue different goals. The Cloud service providers want to maximize throughput by achieving high resource utilization, while users want to minimize expenses while meeting their performance requirements. However, it is difficult to allocate resources in a mutually optimal way due to the lack of information sharing between them.Performace and efficiency has increased through hybrid scheduling algorithm.It is very better than other scheduling Strategy mechanism.

REFERENCES

[1] Dan C. Marinescu : Cloud Computing Theory and Practice. Pp. 1-50 .
[2] School of Software, Sun Yat-sen University : Introduction to Cloud Computing .
[3] Introduction to Cloud Computing .Fact Sheet , Fiche d'information. Page 1- 6..
[4] Jeffrey Dean and Sanjay Ghemawat : MapReduce: Simplified Data Processing on Large Clusters
[5] Lizheng Guo,Shuguang Zhao, Shigen Shen, Changyuan Jiang : Task Scheduling Optimization in Cloud Computing Based on Heuristic Algorithm. In IEE Journal of Networks, Vol. 7, No. 3, March 2012
[6] Xiaocheng Liu, Albert Y. Zomaya, Fellow IEEE, Chen Wang, Bing Bing Zhou, Junliang Chen, Ting Yang, : Priority-Based Consolidation of Parallel Workloads in the Cloud. IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 9, September 2013
[7] Sharad Mehrotra ,Sai Wu, Feng Li , Beng Chin Ooi : Query Optimization for Massively Parallel DataProcessing.
[8] Ludmila Cherkasova , Roy H. Campbell ,Abhishek Verma : Two Sides of a Coin: Optimizing the Schedule of MapReduce Jobs to Minimize Their Makespan and Improve Cluster Performance.
[9] Anish Das Sarma, Christopher Olston,Xiaodan Wang, Randal Burns : CoScan: Cooperative Scan Sharing in the Cloud.
[10] Softwareasaservice,Wikipedia, <http://en.wikipedia.org/wiki/Software_as_a_service>
[11] WelcometotheDataCloud,TheSemanticWebblog,6Oct2008, <http://blogs.zdnet.com/semantic-web/?p=205>



doi crossref

10.22214/IJRASET

INDEX COPERNICUS

45.98

ISRA JIF

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY