Environment Conscious Public Cloud Scheduling Algorithm with Load Balancing

Ashish Kumar Singh
Dept. of Computer Science
SRIT, Jabalpur, MP, India

Sandeep Sahu
Dept. of Computer Science
SRIT, Jabalpur, MP, India

ABSTRACT
Cloud Computing is very fruitful area for today's business. It provide lot of advantages like reduction of hardware and software cost for organization. Organization not has to invest a lot of money for their infrastructure and they gain full benefit of their investment. Number of cloud provider also increasing some of which have their own data center and some also have virtual data centre to provide services over the internet. Cloud is a model where customer expects more services or services without interruption for their money and cloud provider expect more ROI (Return on investment). These thing depend enough on effective scheduling Cloud provider have many VM at their data centre so they must have a proper load balancing to utilize their all VM. At present time environment is becoming serious issue and in near future problem is increasing more and more. So scheduling algorithm must consider the issue of minimization of carbon emission. In our work we propose an algorithm which will address environment and load balancing issue for public cloud.

Keywords
Carbon emission, Cloud Manager, Profit Maximization, Public cloud, Virtual Data Centre

1. INTRODUCTION
Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing has become a popular buzzword; it has been widely used to refer to different technologies, services, and concepts.

Cloud computing providers offer their services according to three fundamental models

1.1 Infrastructure as a service (IaaS):
In this most basic cloud service model, cloud providers offer computers, as physical or more often as virtual machines, and other resources. IaaS refers not to a machine that does all the work, but simply to a facility given to businesses that offers users the leverage of extra storage space in servers and data centers. Example - Google drive (drive.google.com) 5GB Free Space

1.2 Platform as a service (paas):
In the PaaS model, cloud providers deliver a computing platform typically including operating system, programming language execution environment, database, and web server Example - force.com

1.3 Software as a service (Saas):
Cloud providers install and operate application software in the cloud and cloud users access the software from cloud clients Example Google Apps (www.google.com/apps) 30Day Free Trial for one domain QuickBooks Online for small business, visible campus for college management.

1.4 Performance of cloud Computing
Cloud computing is pay-go-model so the customer want to gain more output on their investment . For this good performance means high throughput is desired. Users want the execution of their task as soon as possible if these criteria are not fulfilled then other have less uses because form user's point of view this is the biggest criteria. So for cloud provider to make cloud computing fruitful they must provide high performance.
For the better performance improvement there is the need of a good scheduling algorithm which schedule the job in such a way that provide high performance means high throughput .

Load balancing is also desired because there are a number of VMs are available and a good load balancing strategy distributes the load properly among the VMs. In the absence of this there are some VM are over –loaded and some are under- utilized.

At present time due to huge expansion of cloud Carbon – emission rate is also increased which affect the environment. So it is also biggest challenge that scheduling algorithm must be environment friendly.

To increase the profit of cloud provider scheduling algorithm also schedule the job in max – profit manner means scheduling algorithm maximize the profit of cloud provider

But Cloud Computing performance have depend a lot more on the scheduling algorithm and proper load balancing algorithm. Scheduling algorithm will make such a sequence of process that throughputs are increased and load balancing algorithm divide the load properly between all available resources. Cloud Computing is a parallel processing model where these issue is of vital importance so they also have importance. As cloud is a pay-go-model, the business performance needs to be accelerated which is a challenging issue in the domain. At present time with the importance of environment this issue also become of equally importance.

2. REVIEW OF PRIOR WORKS
Author of [1] suggests an algorithm which covers environment conscious issue for scheduling of HPC applications on distributed cloud centers. Author study shows that at present the carbon emission of ICT industry is becoming equal to the carbon emission of aviation industry.

So now some of government imposing a carbon emission limit over the ICT industry. So if cloud providers not consider this issue they are not able to extend their infrastructure in future. For environment conscious the author consider the carbon emission rate of data center. Carbon emission rate of different data center is different so scheduling algorithm schedule the job to a data center which have minimum carbon emission rate.

Author also covers the issue of cost which will maximize the profit of cloud provider. For cost he considers execution price, elasticity price and data transfer price. On base of these information authors suggest algorithm which will find out data center which maximize the profit from the data – center pair of minimum carbon rate. Author algorithm is of HPC application where applications have the need of more than one VM. After selection of a data center scheduling algorithm select the VM according to the requirement. Author solution is good for environment conscious but he not cover the issue of load balancing to choose the VM form a data center which is also a major issue at present time .

Author of [4] suggest hybrid computing solution which will be the outcome of grid and cloud computing and traditional HPC. He try to combine the best of all these three strategy to gain the new model. Author provides an Elastic Cluster solution and show how it can be used to achieve effective and predictable execution of HPC workloads. Author presents a hybrid computing architecture and a set of strategies for achieving effective and predictable execution of HPC workloads on a hierarchy of resources. Author solution covers increasing and decreasing the size of and Elastic Cluster to adopt to change in workload. Author also covers the issue of scale – out means distributing the workload across Elastic Cluster. Author suggest elastic cluster which is good with high overhead. That was for huge investment. Author also not cover the issue of environment conscious and profit maximization which a major concern for today's cloud environment.

Author of [5] propose a threshold – based dynamic resource allocation scheme for cloud computing that dynamically allocate the virtual resources or virtual machines among the cloud computing applications based on their load changes instead of allocating resources to meet the peak demand. Author studied that peak-load based solution suffer form underutilization of resources because process not all time has the need of all resources .He suggest that dynamic resource allocation scheme is better because static allocation may suffer from either insufficient resources at some time or wastes of resources at some time . So to meet the fluctuating demand dynamic resource allocation here is better.

Author solution monitor and predict the resource needs of the cloud applications and adjust the virtual resources based on application's actual demand. Author focuses on dynamic resource allocation scheme for cloud computing which is good for utilization of resources but for the performance of cloud computing a good scheduling and load balancing is also desired.

Author of [6] propose a solution for managing large image collections. Author presents a cloud computing service and its application for the storage and analysis of very –large image. His solution allows that an input image can be divided into different sub-images that can be stored and processed separately by different agents in the system, facilitating processing very-large images in a parallel manner. His purpose is to create a cloud computing service capable of storing and analyzing very- large image datasets. Author develop cloud computing service prototype for storing and analyzing images. Adequate parallelism and workload balancing of our distributed system is crucial feature to ensure an improved performance. Author actually study real life application of very large image datasets and parallelism will really improves the performance such application. His solution divide large image into sub-images and improve the performance by parallelism. Author not covers issue of environment and cost and also of load balancing.

Author of [8] suggest the algebraic scheduling of the processes because he found that some different processes have different need for the execution so he suggest the algorithm which show the process resource demand in the form of utility function . He suggest that desired resource demand should be in the form soft constraint means it is not necessary that process can execute with the desired resource. As cloud resources and applications grow more heterogeneous, allocating the right resources to different tenants' activities increasingly depends upon understanding tradeoffs regarding their individual behaviors. One may require a specific amount of RAM, another may benefit from a GPU, and a third may benefit from executing on the same rack as a fourth.

Author modify the existing approach where resource consumers has to specify zero or more hard constraints with each request, based on some predetermined attribute schema understood by the cluster scheduler . Such constraints could serve as a filter on the set of machines, enabling identification of the subset that is suitable for the corresponding request. But, this approach ignores an important issue: in many cases, the desired machine characteristics provide benefit but are not mandatory.

For example, running a new task on the same machine as another with which it communicates frequently can improve performance, but failing to do so does not prevent the task's execution—it just makes it somewhat less efficient. We refer to such cases as Soft constraints. Treating them as hard constraints can lead to unnecessary resource under-utilization, and ignoring them can lead to lower efficiency, making them an important consideration for cloud schedulers.

This paper proposes a specific approach for accommodating soft constraints, as well as hard constraints and general machine heterogeneity. In this model, each job submitted for processing is accompanied by a resource request, which is expressed as utility functions in the form of algebraic expressions indicating what benefit would be realized if particular resources were assigned to it.

Author suggests algebraic scheduling because of heterogeneity in cloud environment. He suggests that process has to express their resource requirement in soft constraint and algebraic scheduling decide that process has to be executed on which VM. In his scheduling utility function express all option in sequence which will have a lot of overhead.

3. PROPOSED PUBLIC CLOUD SCHEDULING ALGORITHM

Important point of scheduling in private cloud

1. Cloud computing is pay- go model so number of job rejected must be less so scheduling must consider this issue.

2. Cloud provider has to gain more profit so scheduling algorithm must increase the profit of cloud provider.
3. Algorithm must be environment conscious because Carbon emission rate of ICT (Information and Communication Technology) industry is now approximately equal to the aviation industry so government impose carbon emission limit. If scheduling algorithm not cover this issue then cloud provider are not able expand there infrastructure.
4. Environment conscious also provide the benefit of point (1) and (2) because if cloud provide has more infrastructure then number of job rejection is also low and to meet the more demand of cloud user cloud provider has to increase infrastructure.
5. Every data centre has own ready queue which is common for all the VM of that data centre. The data centre may virtual also. Data centre services is provided by some cloud provider like amazon to other cloud provider. It means some cloud provider not have their own data centre pay and use this service from other cloud provider. On that case it is called VDC (Virtual Data Centre).
6. Load balancing must be there so process must be migrated form over loaded VM to less loaded VM within data centre. This thing will not done continuously mean it will done on periodic base so it will not increase more overhead. Cloud manager periodically monitors the status of the VMs for the distribution of the load, if an overloaded VM is found then the cloud manage migrates the load of the overloaded VM to the underutilized VM.
7. On summarizing we can say that
    a. For every arrived process a data centers are chosen which have minimum carbon emission.
    b. If more than one data centre is found then selection of one data centre is based on profit maximization.
    c. After selection of data centre job is put on the queue of data centre. A data centre has many VM and all VM of DC or VDC have common queue. For higher throughput processes are selected on the base of SJF with bound waiting.
    d. A least loaded VM is chosen for every scheduled process in a DC.

4. PROPOSED ALGORITHM FOR PUBLIC CLOUD

A user submit his requirement for an application j in the form of a tuple (dj, eji , (DT)j) where dj is the deadline to complete application j, eji is the application execution time on the data center i. , (DT)j is the size of data to be transferred.

Here (CO2E)ij = riCO2 x Eij Where riCO2 is the carbon emission rate of data center i.

And (Profit)ij = (ProfitExec)ij + (ProfData)ij means (ProfitExec)ij = ejipc – pie x Eij

Here pc CPU execution price fot processing time pie is the electricity price

(ProfData)ij = (DT)j x (pDTU – piDT)

Here pDTU is the data transfer price for the upload/download piDT is the data transfer price for upload/download Cloud provider has to pay data center I the energy cost and data transfer cost depending on its electricity price pie and data transfer price for piDT upload/download. Cloud provider then charges foxed price to the user for executing his application based on the CPU execution price pc and data transfer price piDT for the processing time and upload/download respectively.

Step – I → For each application in the list of application to be mapped find the data Center of which the carbon emission is the the minimum means minimum (CO2E)ij among all the data centers which can complete the application by its deadline

Step – II → Among all the application – data center pairs found in Step – I find the pair that results in the maximum profit means maximum (Prof)ij

Step –III → once a data center a selected then it will put in the queue of of request. Cloud manager in the data center maintain a data structure comprising of the Job ID, VM ID and VM Status.

Step – IV → Add new process to the tail of the queue and

p_num= p_num + 1;

[p_num indicate the number of process in the queue of the DC]

Step – V → While i<p_num

tag[process]=tag[process] -1;

(tab[process] indicate the bound waiting tag for processes of DC) If tag[process]=0 then

move this process at the head of the queue

Repeat

Step – VI → Select min load_per[VM] (least loaded and if more than one then least hop Time and VM capacity indicate by the num field also more than the request size) VM from the VM pool and

p_cur[VM] = p_cur[VM] +1;

load_per[VM] = p_cur[VM]*100/num[VM]

Step – VII → if tag[process] ==0 then

step – IX Else step – VIII

Step – VIII → Select min burst[process] form Ready queue of DC

Step – IX → Remove the selected process form queue and dispatch it to selected VM

p_num= p_num -1;

Step – X→ goto step – I

Step – XII → Stop

Time Complexity

Let n processes arrives in unit time then there are d data center and on average there are v number of VM in data centers.

Time required to choose minimum carbon emission and max profit data center on worst case = nd

Time required to choose a VM is = nd * v

Time required for load balancing on an average = d*v*n

Time required for process selection (Let q number of process is already in the queue of the DC) in worst case = (q+1) n

Time required for bound waiting updating = (q+1) n

Total time = (nvd + nvd + 2n (q+1))

Normally n (Number of processes arrived) and v (average number of VM in data center) and is much more than of number of data center menas, n&v >>d

So, Total time = nv + nv +2nq

= 2nv + 2nq

Total time = O (n2).

5. CONCLUSION & FUTURE WORK

Public cloud services become common on today's date so their expansion also which will affect environment. Our suggested algorithm will minimize carbon emission which will beneficial for cloud provider if they want to be work as cloud provider on long term basis. Our algorithm cover the issue of profit maximization means increase the ROI of cloud provider which is also essential for cloud provider to work in this field as fruitful way. For individual DC higher throughput and load balancing is maintained by efficient scheduling and load balancing.

6. REFERENCES

[1] Saurabh K Garg,Chee Shin Yeo, Arun Anandasivam, Rajkumar Buyya " Environment – Conscious Scheduling of HPC application on distributed Cloud – oriented centers " ScienceDirect My 2010

[2] Rashmi K S and Suma V and Vaidehi M " Factors Influencing Job Rejection in Cloud Environment " IJC May 2012

[3] Rashmi K S and Suma V and Vaidehi M " Enhanced Load Balancing Approach to avoid Deadlocks in Cloud "IJCA-ACCTHPCA, June 2012

[4] Gabriel Mateescu , Wolfgan Gentzsch , Calvin J Ribbens " Hybrid Computing – Where HPC meets grid and Cloud Computing " ScienceDirect Nov 2010

[5] Weiwei Lin , james Z. Wang, Chen Liang, Deyu Qi " A Thrseshold – Based Dynamic Resource Allocation Scheme for Cloud Computing " ScienceDirect 2011

[6] Raul Alonso-Calvo , Jose Crespo , Miguel Garcia-Remesal, Alberto Aungita and Victor Maojo " On Distributing load in cloud computing : A real application for very –large image datasets " ScienceDirect 2010

[7] Ashraf Zia & Muhammad Naeem Ahmad Khan " Idetifying Key Challenges in Performance Issues in Cloud Computing " IJMECS 2012

[8] Alexey Tumanov , James Cipar and Michael A Kozuch " Algebraic Scheduling of Mixed Workloads in Hetrogeneous Clouds " ACM 2012

[9] Monir Abdullah , Mohamed Othman " Cost – Based Multi – QoS Job Scheduling using Divisible Load Theory in Cloud Computing " ScienceDirect 2013

[10] R. Santosh and T. Ravichandran " Non-Preemptive on-Line Scheduling of Real-Time Services with Task Migration for Cloud Computing " EJSR 2012

[11] Soumya Ray and Ajanta De Sarkar " Execution Analysis of Load Balancing Algorithm in Cloud Computing Environment "in International Journal on Cloud Computing : Service and Architecture (IJCCSA) , Vol 2 Oct 2012

[12] K. L. Giridas , A Shajin nargunam "CHPS in Cloud Computing Environment " in International Journal of Engineering and Technology (IJET) Oct – Nov 2012

[13] Zhang. Y and Zhou Y " TransOS : A Transparent computing-based operating system for the cloud"

[14] Donald Mclaughin and Partha Dasgupta " Preemptive Scheduling for Distributed System " [13] Harsora and Dr. Apurva Shah " A Modified Genetic Algorithm for Process Scheduling in Distributed System " IJCA , 2011

[15] Indraveer Chana and Anju Bala " A Survey of Varoous Workflow Scheduling Algorithm in Cloud Environment " NCICT 2011

[16] Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, Cloud Computing A Practical Ap- proach, TATA McGRAW-HILLEdition2010.

[17] Ashish Kumar Singh etal "Scheduling Algorithm with Load Balancing in CloudComputing" IJSER 2014

7. AUTHORS PROFILE

Ashish Kumar Singh completed the B.E. in Computer Science from RGPV, Bhopal and M.B.A. from BU, Bhopal degrees in 2003 and 2007, respectively. Currently he is doing dissertation of M.E. (CS) final semester on same topic as of paper, From SRIT, Jabalpur, MP, India.

Sandeep Sahu completed the M.Tech. in Computer Science from IIT Guwahati, India in 2009. Currently he is working with SRIT, Jabalpur as HOD, Department of Computer Science and Applications. He is the guide of dissertation work of Ashish Singh