ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

As An Effective and Efficient Profit Maximization Scheme of Quality of Service in Cloud Computing

Dr. Shaik Abdul Nabi
Professor,
Dept of CSE,
AVN Institute of Engineering and Technology.

G.Dayakar
Assistant Professor,
Dept of CSE,
AVN Institute of Engineering and Technology.

Aithu Harish
PG Scholar,
Dept of CSE,
AVN Institute of Engineering and Technology.

ABSTRACT:

The cloud is a next generation platform that provides dynamic resource pools, virtualization, and high availability. Today, it have the ability to utilize scalable, distributed computing environments within the confines of the Internet, a practice known as cloud computing. Cloud computing is the concept implemented to decipher the daily computing problems, likes of hardware software and resource availability unhurried by computer users. The cloud computing provides an undemanding and non-ineffectual solution for daily computing. Prevailing cloud systems mainly focus on finding an effective solution for the resource management.

In cloud computing, the study of economics of the cloud is critically important. The maximization of profit is done in this. For maximizing the profit first should understand the cost and revenue. Profit maximization must consider the user satisfaction also the cost of the cloud includes the renting cost and power consumption cost. For maximizing, must reduce the cost. For this it will configure the server perfectly. For configuring the server, calculate the expected waiting time and service charge is calculated. Using the optimizing method, will optimize the speed and the size so get maximum profit.

Keywords:

Cloud computing, Pricing model, load balancing.

I. Introduction:

The cloud is a next generation platform that provides dynamic resource pools, virtualization, and high availability. Today, have the ability to utilize scalable, distributed computing environments within the confines of the Internet, a practice known as cloud computing. Cloud computing is the concept implemented to decipher the daily computing problems, likes of hardware software and resource availability unhurried by computer users. The cloud computing provides an undemanding and non-ineffectual solution for daily computing. Prevailing cloud systems mainly focus on finding an effective solution for the resource management.

Cloud Computing [1] is Internet based computing where virtual shared servers provide software, infrastructure, platform, devices and other resources and hosting to customers on a pay-as-you-use basis. The cloud makes it possible for user to access your information from anywhere at any time. Cloud computing enables a User what you Need and Pay for what you Use cost model. This will enable businesses to invest on innovative solutions that will help them address key customer challenges instead of worrying about operational details. “Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

with minimal management effort or service provider interaction." More specifically, cloud describes the use of a collection of services, applications, information, and infrastructure comprised of pools of computer, network, information, and storage resources. These components can be rapidly orchestrated, provisioned, implemented and decommissioned, and scaled up or down; providing for an on-demand utility-like model of allocation and consumption. Cloud enhances collaboration, agility, scaling, and availability, and provides the potential for cost reduction through optimized and efficient computing. In business concepts the profit is the main factor to be exist in the field of the particular environment. Obviously, the need of profit maximization in cloud computing environment is required. 60 billion servers are currently working in this world. So the server required a huge amount of power. In order to preserve the consumption of energy, need of maximum utilization of resources is important.

The important things to consider while developing such algorithm are : estimation of load, comparison of load, stability of different system, performance of system, interaction between the nodes, nature of work to be transferred, selecting of nodes and many other ones. This load considered can be in terms of CPU load, amount of memory used, delay or Network load. A dynamic load balancing scheme need to be proposed for avoiding over-loaded and under-loaded scenarios in cloud system. Thus the problem of dynamic Application placement should be addressed for allocating jobs to machines based on their changing demands in heterogeneous cloud environments. The profit maximization [2] is done here. In this the service charges for each service that is requested by the user is calculated. Normally between the user and server has some agreement i.e., service level agreement. In this service level agreement, defined the QoS need to provide for the user and the maximum needed execution time also. If the service provider violates this SLA no charge is provided for the particular service.

So their will be the lost of the profit. Here calculating the optimal speed and size of the input the service level agreement is provided and here a pricing model is, developed according to optimal size and speed and service charge is calculated. So the service provider can maximize the profit.

II. Related Work

It includes the relative mechanisms and the methods which are implemented earlier and also the advantages and disadvantages of each method is described briefly. According to the survey of the earlier mechanism, it finds that the current system implemented has more advantages. Saurabh Kumar Garg et al. [1] Here focus on Meta scheduling of different applications from a community of users considering a commodity market. In commodity markets, service providers primarily charge the end user for services that consumes based on the value, derive from it. Pricing policies are based on the demand from the users and the supply of resources is the main driver in the competitive, commodity market models. Therefore, a user competes with other users and a resource owner with other resource owners.

The financial institution Morgan Stanley is an example of a user community that has various branches across the world. Each branch has computational needs and QoS constraints that can be satisfied by Grid resources. In this scenario, it is more appealing for the company to schedule various applications in a coordinated manner. Here propose two meta-scheduling online heuristics Min-Min Cost Time Tradeoff and Max-Min Cost Time Tradeoff to manage the tradeoff between overall execution time and cost and minimize them simultaneously on the basis of a tradeoff factor. The tradeoff factor indicates the priority of optimizing cost over time. These heuristics can be easily integrated in existing meta-brokers of Grid Market Infrastructures. Second, in order to study the effectiveness and efficiency of the proposed heuristics, evaluated our heuristics by an extensive simulation study.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The heuristics can run in either batch mode or immediate mode. In the batch mode, the meta-broker waits for a certain time interval .Then at the end of the schedule interval, the meta-broker allocates all user applications to available resources. In contrast, immediate mode heuristics immediately map a task to some machine in the system for execution upon the arrival of the application. In simulation, studied the heuristics in batch mode.

For scheduling approaches outside Grid computing, Min-Min, Min-Max and Suffrage are three major task-level heuristics employed for resource allocation. As developed based on specific domain knowledge, cannot be applied directly to Grid scheduling problems, and hence have to be enhanced accordingly. The main contribution are thus to design two heuristics to manage and optimize the tradeoff between cost and execution time of user application in a concurrent user's environment for Utility Grids. Adopt some ideas from Min-Min and Min-Max heuristics to design my algorithm.

The meta-broker presented in this work envisions future market models where various service providers with large computing installations and consumers from educational, industrial and research institutions will meet. Service providers sell the CPU time slots on their resource and the consumer will buy these time slots to run their applications. The meta-broker may have control over allocations to some or all processors in a resource for some time intervals. This scenario can be formulated as an economic system with three main participants:

Service Providers-Each of the resources can be considered: Meta-Broker System a provider of services such as CPU time slots. Each free CPU slot includes two parameters: number of processors and time for which are free. Providers have to satisfy requests of the local users at each site and Grid user requests that arrive through the meta-broker. Providers assign CPUs for the exclusive use of the meta-broker through advanced reservation, and supply information about the availability of CPUs and usage cost per second at regular intervals. The economic system considered here is co-operative in nature, that is, the participants trust and benefit each other by co-operating with each other. Therefore, the possibility of providers supplying wrong or malicious information is discounted. It is assumed that service price does not change during the scheduling of applications.

Users: Users submit their applications to the meta-scheduler for execution at the resources inthe computing installation/Grid. The users require that the applications be executed in the most economic and efficient manner. The users also can provide a trade-off factor to indicate the importance of cost over execution time, otherwise it will be set by the meta-broker. The tradeoff factor can be calculated by user on the basis of urgency and budget for executing the application. In the current system, assume user applications are based on the parallel application model, that is, the application requires a certain number of CPUs simultaneously on the same Grid resource for certain time interval.

Meta-Broker: The meta-broker uses the information supplied by the providers and the users to match jobs to the appropriate services. The scheduling of user applications is done in batch mode at the end of a Schedule Interval. At the end of a SI, the meta-broker calculates the best schedule for all user applications after negotiating the time slots with the service providers. The objective of the meta-broker is to schedule all user application such that both total time and cost for applications execution are minimized.

In general, users have two QoS requirements, i.e., the processing time and execution cost for executing their applications on pay-per-use services. The users normally would like to get the execution done at the lowest possible cost in minimum time. Thus, introduce trade-off factor which indicates the importance level of cost for users over time. Two meta-scheduling heuristics that aim to manage the tradeoff between execution cost and time.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

RizosSakellariou et al. [2] How a scheduling algorithm can allocate jobs to machines in a way that satisfies constraints of Deadline and Budget at the same time. Each job is considered to be a set of independent Gridlets, objects that contain all the information related to a job and its execution management details such as job length in million instructions, disk I/O operations, input and output file sizes and the job originator.

Consider that a budget constraint needs to besatisfied. Each job, when running on a machine, costs some money. Thus, the overall aim is to find the schedule that gives the shortest makespan for a given DAG and a given set of resources without exceeding the budget available. In order to solve the problem of scheduling optimally under a budget constraint, The idea in both approaches is to start from an assignment which has good performance under one of the two optimization criteria considered and swap tasks between machines trying to optimize as much as possible for the other criterion. The idea is to keep swapping tasks between machines by choosing first those tasks where the largest savings in terms of money will result in the smallest loss in terms of schedule length.

Call this approach as LOSS. The second approach starts with the cheapest assignment of tasks onto resources. As long as there is budget available, the idea is to keep swapping tasks between machines by choosing first those tasks where the largest benefits in terms of minimizing the makespan will be obtained for the smallest expense. Call this approach GAIN. If the available budget is bigger or equal to the money cost required for this assignment then this assignment can be used straightaway. Find affordable assignments with better makespan when the loss approach is applied, instead with the gain approach. The loss approach applies re-assignment to an assignment that is given by a good DAG scheduling heuristic, whereas in the gain approach the cheapest assignment is used to build the schedule; this may have the worst makespan.

However, in cases where the available budget is close to the cheapest budget, gain1 gives better makespan than loss1 or loss2. The running time, it appears that the loss approach takes more time as we move towards a budget close to the cost of the cheapest assignment; the opposite happens with the gain approach. This is correlated with the starting basis of each of the two approaches. Qian Zhu et al. [3] While current cloud systems are beginning to offer the utility-like provisioning of services, provisioning of resources has to be controlled by the end users. It is desirable that resource allocation in a cloud environment can be performed automatically and dynamically, based on users' high-level needs. The allocation of resource to each VM can be dynamically controlled, and the resource costs incurred depend upon the resources allocated. Furthermore, aresource model is proposed to map any given combination of values of adaptive parameters to resource requirements in order to guarantee that the resource cost stays under the budget.

The CPU cycle/memory allocation generated through the use of our resource model is within 5 percent of the actual CPU/memory utilization. Furthermore, the model can be trained on one system and then applied on a different system effectively. Second, dynamic resource provisioning algorithm achieves a benefit of up to 200 percent of what is possible through a static provisioning scheme. At the same time, the scheme could perform parameter adaptation to meet a number of different time and budget constraints for the two applications. Our cloud environment allows on demand access to resources. Applications are charged for their resource usage according to a pricing model. fine-grained allocation and pricing of resources is possible for the virtual environment.

CPU usage: Xen provides a Simple Earliest Deadline First scheduler that implements weighted fair sharing of the CPU capacity among all the VMs. The share of CPU cycles for a particular VM can be changed at runtime. The SEDF scheduler can operate in two modes: capped and non-capped. In the capped mode, a VM cannot use more than its share of the total CPU

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

time in any time interval,even if there are idle CPU cycles available.

Memory usage: Each VM is configured with a maximum entitled memory. The VM starts with an initial memory allocation, which can be later increased up to the specified maximum value. Pricing model: this work assumes a fine-grained pricing model where a higher allocation of CPU cycle percentage or memory is associated with a higher cost for each time unit. Beyond this basic assumption, our resource allocation framework is independent of the details of the pricing model Evaluated frame work using two different pricing models. For simplicity, only focus on costs associated with computing cycle allocation and memory allocation. Depending upon the application, additional costs may be associated with storage and data transfers. A linear pricing model and an exponential pricing model. In the linear pricing model, the resource cost charged to the users is linearly scaled with the amount of resources that have been assigned to the application.

How resource models are generated with the goal of converting changes in values of an adaptive parameter into CPU cycles and memory allocation requests Feedback control model has been applied for dynamic virtual resource provisioning. Unlike the previous work which optimizes a single performance metric by directly controlling the resources allocated to the application,here consider a more unique and complex problem where the application benefit depends on the values of the adaptive parameters thus making it hard to maximize the benefit by controlling the resource allocations.

Use the feedback control model to guide the parameter adaptation in order to maximize the application benefit while satisfying the time constraint and resource budget. Then virtual resources are dynamically provisioned according to the change in the adaptive parameters. In control theory, an object to be controlled is typically represented as an input-output system, where the inputs are the control knobs and the outputs are the metrics being controlled. Typically, a controller manipulates the inputs to the system under the guidance of a performance objective

Processing progress: It is defined as the ratio between the currently obtained application benefit and the elapsed execution time. This metric measures the rate at which the application processing is gaining the benefit.

Performance/cost ratio: It is defined as the ratio between the currently obtained application benefit and the cost of the resources that have been assigned to the application. This metric measures the rate of gaining the benefit for every unit of resource budget consumption. Demonstrating model is effective in CPU cycle and memory allocation with high resource utilization. Also, models trained on one type of hardware can still be effective on another type of hardware. Demonstrate that the maximum benefit achieved byour dynamic resource provisioning method is larger than that achieved by Static Scheduling, within the time constraint. At the same time, the resource cost always stays under the pre specified budget.

Gemma Reig et al. [4] Here a prediction system to determine the minimum job resource requirements to be executed before its deadline. One key innovation of the prediction system is the usage of Machine Learning to enable the translation from service-level metrics to resource requirements. Enabling the cloud to non-expert IT users by means of using service-level metrics and help providers to do a smart utilization of their resources by using the resources left by web applications to execute jobs in an efficient way e.g. discard jobs in advance, avoiding the risk of wasting resources in executing jobs that will not meet their deadlines. The Scheduler accepts incoming jobs and web applications to be planned. It queries the Prediction System and it decides, depending on the policy being used and the resources status, how to allocate resources to the incoming jobs and how to elastically size up and down the resource allocation for

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

web applications, in order to fulfill their respective QoS. The Prediction System is in charge of predicting the minimum resource requirements needed to meet SLAs.It consists of an Analytical Predictor module anda Self-Adjusting Predictor module that predicts by learning from previous job executions. Architecture includes a dual-purpose predictor that allows users to negotiate with providers in service-level terms and provides a mean for the Scheduler to perform smart resource allocation using these predictions. Here introduced ML techniques in a Self-Adjusting Predictor that predicts the required resources to fulfill a givenservice-level metric using the results from previous executions. Regarding the CPU prediction, achieve high prediction accuracy using the Bagging with M5P algorithm. An AnalyticalPredictor that is used to predict the resource requirements whilst the Self-Adjusting Predictor is not enough trained.

Ana Maria Oprescu et al. [5] BaTS,budget, constrained scheduler. BaTS can schedule large bags of tasks onto multiple clouds with different CPU performance and cost. BaTS schedules such that a bag of tasks will be executed within a given budget, while minimizing the completion time. BaTS requires no a-priori information about task completion times, instead BaTS learns application throughput at run time, using an initial sampling phase and a moving average throughout the computation. BaTS are scheduling large bags of tasks onto multiple cloud platforms. The core functionality is to allocate a number of machines from different clouds, and to adapt the allocation regularly by acquiring or releasing machines in order to minimize the overall makespan while respecting the given budget limitation machines.

Assume that the tasks of a bag are independent of each other, so they are ready to be scheduled immediately. Also assume,The individual tasks are scheduling a round-robin manner onto the allocated machines. Assume that the tasks can be preempted and rescheduled later, if needed by a reconfiguration of the cloud environment. Task model incurs no prior knowledge about the task execution times

For this purpose, BaTS uses a cumulative moving average mechanism. Based on these estimates, BaTS decides which combination of machines would satisfy the budget constraint and optimize the makespan has changed the way compute resources can be accessed. The elasticity of clouds allows users to allocate computers on the fly, according to the application's needs. While each commercial offering has a defined quality of service, users still lack guidance for deciding how many machines of which type and for how long would be necessary for their application. Bags of tasks are an important class of applications that lend themselves well for execution in elastic environments. In this work, introduced BaTS, our budget constrained scheduler for bag-of-tasks applications.

BaTS requires no a-priori information about task execution times. It uses statistical methods to execute samples of tasks on all cloud platforms that are available to a user.BaTS monitor the progress of the tasks and dynamically reconfigures the set of machines, based on the expected budget consumption and completion time.

Comparisons of above described papers are shown in the table 1 below.

EXISTING SYSTEM	METHODS	ADVANTAGES	DISADVANTAGES
Scheduling Parallel Application on Utility Grids: Time and Cost Trade – off.	simplest Earliest Deadline First Scheduler Operates two modes a)Capped b)NonCapped	Resource allocation in cloud environment can be performed automatically and dynamically Cost stays under the budget. The model can be trained on one system and then applied on different system effectively	Problem with dynamic approach is high runtime overhead. Only on-demand pricing model is user Consumers only get profit.
Scheduling Wodcflows with Budget Constrains	Meta scheduling Min-Min cost time tradeoff Max-max cost time tradeoff	Minimize the cost Tradeoff factor indicating level of cost for users	User get benefit not for service provider . power consumption is not considered Based only on pay on demand pricing model. Other pricing model is not considered
Resource provisioning with Budget Constrains for Adaptive Application in Cloud Computing	Loss and Gain Approach	Budget constraints are satisfied. Simple to execute Better makespan is build	Only considering the time and cost OS and other parameters are not considering. Loss approach takes more time.
Prediction of Job Resource Requirements for Deadline Schedulers to Manage High- level SLAs on Cloud	a)Self adjusting predicto r b)Analytical predictor	It contains two predictors. if SAP is not trained, Analytical predictor schedules. It will be executed before the deadline. Predict the CPU for jobs	Cost is not considering. Considering only the execution time. Other parameters are not considering.
BAG- of –Tasks Scheduling under Budget Constraints.	Budget constrained scheduler	does not exceed the budget user can determine the budget	Quality of service is not considered. user only get benefit.

Table 1: Comparison table of existing systems

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

III. GENERAL SYSTEM MODEL

The main aim of our resource allocation is to allocate the online service request for applications which are CPU and memory intensive. To achieve the objective of adapting resource allocation for satisfying these of customers.In ant-colony architecture. The components are users or brokers, cloud controller, virtual machines, physical machines, cloud controller and Queen ant, Worker ant, SLA monitor agent. Users or brokers acting on their behalf submit service request to the cloud via cloud controller for processing. Cloud controller acts as the interface between the cloud service provider and external users or brokers. It acts similar to the queen in the ant colony. In virtual machines where the applications of customers will be deployed. We can dynamically create, start, store and migrate these VMs depending on our requirement, from one physical to another [6]. Physical machines are the physical computing servers that will provide hardware infrastructure for creating virtual machines. Cloud controller and Queen ant receives the request from users or customers and given to the controller. Cloud controller maintains a queue for storing the service request for hosting the applications.

It enqueues each of the service request received in this queue. It generates tester, scout and clean worker ants periodically. The movement of this ant agents is modeled in the following way. Each ant except queen and worker maintains a visited node list which is initially empty. Each node in the cloud maintains a list of neighboring nodes information. Whenever an ant reaches a node it updates the controller about the current utilization and randomly chooses an unvisited neighbouring node. When all the nodes are covered it makes the visited node list empty and continuous the number of ants in the same way. We can change the number of ants that will be produce so that it will yield better result depending on our requirement. Whenever a service request received in the queue one of the worker ants creates a VM with specific CPU processing power and memory etc., if accepted. So worker ants are always looking in the queue it check if there are some pending request to be processed.

The worker and is only responsible for deploying the request on a VM. Load balancing decisions are taken by tester ant. After deploying it creates a service level agreement monitor agent that monitors the hosted applications. It passes this information to the hypervisor on that host in the form of a variable (SLAM), which is calculated depending on the performance of applications. Calculation Provider benefit of allocation of Job 'J' to Node N.

IV.CONCLUSION:

A pricing model is developed for cloud computing which takes many factors into considerations, such as the requirement r of a service, the workload of an application environment, the configuration (m and s) of a multi-server system, the service level agreement c, the satisfaction (r ands0) of a consumer, the quality (W and T) of a service, the penalty d of a low-quality service, the cost of renting, the cost of energy consumption, and a service provider's margin and profit. And this will schedules the job according to optimization of speed and size of the input hereby maximizing the profit.

REFERENCES:

[1]M. Armbrustet al., "A View of Cloud Computing," Common. ACM, vol. 53, no. 4, 2010, pp. 50–58.

[2]JunweiCao,Kai Hwang, Keqin Lin, Albert Y.Zamaya"Optimal Multiserver Configuration for profit Maximization in cloud computing ," ieee transactions on parallel and distributed systems, vol. 24, no. 6, June 2013.

[3]Qian Zhu, Student Member, IEEE, and GaganAgrawal,Senior Member, IEEE, "Resource Provisioning with Budget Constraints for Adaptive Applications in Cloud Environments," ieee transactions on services computing, vol. 5, no. 4, october-december 2012.

[4]Pankesh Patel, AjithRanabahuAmitSheth, "ServiceLevel Agreement in Cloud Computing," Knoesis Center, Wright State University, USA.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[5] Saurabh Kumar Garg, Rajkumar Buyya1 and H. J. Siegel "Scheduling Parallel Application On Utility Grids: Time And Cost Trade-off manag-ement" Management" The University of Mel-bouneVictoria 3010, Australia.

[6]Gemma Reig, Javier Alonso, and JordiGuitart "Prediction Of Job Resource Requirements For Deadline Schedulers to Manage High-Level SLAs On The Cloud", 2010 Ninth IEEE International Symposium on Network Computing and Applications**.**

[7]Mayank Mishra, Anwesha Das, PurushottamKulkarni, and AnirudhaSahoo,IIT Bombay"Dynamic Resource Management Using Virtual Machine Migrations," International journal in cloud computing.

Author's Details:

Dr. Shaik Abdul Nabi is working as professor & Head of the Dept. of CSE, AVN Inst.Of Engg.& Tech, Hyderabad, T.S, India. He completed his B.E (Computer Science) from Osmania University, Hyderabad. He has completed his M.Tech. from JNTU Hyderabad campus and he received Doctor of Philosophy (Ph.D) in the area of Web Mining from AcharyaNagarjuna University, Guntur, AP, India. He is a certified professional by Microsoft.He is having 17 years of Teaching Experience in various Engineering Colleges. He has published 15 publications in International / National Journals and presented 08 papers in National / International conferences. His expertise areas are Data warehousing and Data Mining, Data Structures & UNIX Networking Programming, Cloud Computing and Mobile Computing.

G.Dayakar is working as Asst. Professor in Dept. of CSE, AVN Institute Of Engineering & Technology, Hyderabad, T.S, India. He completed his B.Tech (Computer Science) from JNTU, Hyderabad. He has completed his M.Tech. from JNTU Hyderabad campus, India. He is a certified professional in Teaching by National Institute Of Technical Teachers Training & Research (Govt Of India) He is having 10 years of Teaching Experience in various Engineering Colleges. His expertise areas are Design and Analysis Of Algortithms, Data Structures & UNIX Networking Programming.

Aithu Harish PG Scholar in Dept of CSE .AVN Institute of Engineering And Technology.