A Survey on Resource Scheduling and Allocation Policy in a Cloud Environment

V. Manimaran[1], **S. Prabhu**[2]

[1]PG Scholar, Department of Computer Science and Engineering, Nandha Engineering College, Erode, Tamilnadu, India

[2]Assistant Professor, Department of Computer Science and Engineering, Nandha Engineering College, Erode, Tamilnadu, India

Abstract: ***In this paper the computability of distinctive resource scheduling algorithms and allocation policies in different dimensions are expressed. Resource Scheduling is a complicated task in cloud computing environment since there are many alternative computer systems with varying capacities. Resource allocation task is mainly scheduled for the Process which gives the available user preferences and resources. Recently, there has been a dramatic increase in the popularity of cloud computing systems that multiplex many users on the same physical infrastructure, and rent computing resources on-demand, bill on a pay-as-you-go basis. These cloud computing environments provide an illusion of infinite computing resources to cloud users so that they can decrease or increase their resource consumption rate according to the demands. Cloud computing offers more number of cloud users requesting number of cloud services simultaneously, so there should be a provision that all resources are made available to requesting user in efficient manner to satisfy their need without compromising on the performance of the cloud resources. The process of optimizing the resources being allocated over various virtual machines is the main challenge in cloud computing.***

Keywords: Cloud Computing, Resource Management, Resource Scheduling, Resource Allocation.

1. Introduction

Computing based on the internet sharing resources is called as cloud computing. Cloud computing is the fastest growing technology that is offering omnipresent services to users. The need of services to the lowest level is in demand. Nowadays no one is ready to purchase the devices that provide the services. The users rather purchase the services provided by the devices at the big servers. The infrastructure of pay-per-use is highly in demand. The users from different locations just like to have the services and pay for the time being they are availing the services. Cloud computing enables convenient and on-demand network access to shared pool of computing resources that needs to be controlled. It delivers applications which are accessible from web browsers, desktop and mobile applications. Optimization of energy efficiency in cloud computing is unavoidable. It is a large scale computing using virtual resources. Its popularity is emerging as a cost effective alternative and also High Performance Computing for supercomputers. There have been different clouds releases until now Eucalyptus, Hadoop, and Nimbus etc.

Resource scheduling is the basic and key process for clouds in Infrastructure as a Service (IaaS) as the need of the request processing is necessary in the cloud. Each server has limited resources so jobs/requests needs to be scheduled. Each application in the cloud is designed as a business processes including a set of abstract processes. To allocate the resources to the tasks there need to schedule of the resources as well as tasks coming to the resources. There need to be a Service Level Agreements (SLAs) for Quality of Service (QoS). Till now no algorithm is been introduced which considers reliability and availability. According to the paradigm of cloud there has been a lot of task scheduling algorithms, some are being fetched on the basics of scheduling done on the operating systems. The basics of operating system job scheduling is taken and applied to the resources being installed in the cloud environment. Cloud computing has a base of distributed, grid and virtualization. Till now unbalanced strategies are being introduced. The cost for transferring data and information should also be included. It should be secure, optimal and convenient. The main objective is to satisfy providers and consumers in optimized strategies as to gain resource efficiency and maximum profit.

2. Need for Resource Scheduling

In cloud computing, Resource Allocation (RA) is the process of assigning available resources to the required cloud applications over the internet. Resource allocation starves services if the allocation is managed imprecisely. Resource provisioning solves that problem by allowing the service providers to manage the resources for every individual module. Resource Allocation Strategy (RAS) is all about integrating cloud provider activities for utilizing and allocating rare resources within the limit of cloud environment so as to meet the needs of the cloud application. It requires the type and amount of resources needed by every application in order to finish a user job. The order and time of allocation of resources are also an input for an optimal RAS.

There is excessive need for the cloud services to the schedule of the resources as this scheduling will further followed by the task/job scheduling inside of the resources. There may be more instances of the single resource that they can be run at the same time. There is need of checking of availability and reliability and also the load must be equalized among the resources of the same type. For the above parameters there need to be a function or procedure that could check them and allocation should be done in the best and optimal way. There are network strategies that could provide services like compute, storage and bandwidth management at less cost. The best way is combine together the computability of network strategies with scheduling algorithms. Usually when tasks are scheduled they are done according to user's requirements and requests but while looking into all the aspects the computation needs to be done. Application scalability is the main aim for the cloud services to achieve. In cloud scalability of resources allows real time provisioning of the resources for services. Cloud has complex execution environment but it has to provide the QoS to its users. Virtual resources are used effectively for the fully customizable configuration environment for application.

3. Related Work

There have been a lot of work done on resource scheduling and allocation in cloud computing. New algorithms and management techniques and different methods for resource scheduling in cloud computing are being preferred to make cloud computing a best experience for providers as well as customers. The surveys on scheduling strategies, techniques, methods have been done and a lot of task/job scheduling algorithms are introduced. The resource scheduling is been a tough job in cloud especially as it is the one which decides which process will be allocated to which resource and for how much time [2]. There are also resource allocation strategies that take into consideration the input parameters and on the basis if whether they are related to either of provider and customer. These parameters are execution time, policy, virtual machine, application, auction, utility function, gossip, hardware resource dependency, SLAs. While making a strategy the allocation methods should keep into consideration resource contention, fragmentation, under provisioning and over provisioning [6]. The different task scheduling methods in cloud computing are Cloud Service, User Level, Dynamic and Static, Heuristic, Workflow and Real Time scheduling. Some of the scheduling algorithms in cloud whether or job or task or resources or workflow are Compromised-Time-Cost, Particle Swarm Optimization based Heuristic, Improved cost based for tasks, RASA workflow, SHEFT workflow, Innovative transaction intensive cost constraint, Multiple QoS Constrained for Multi- Workflows. There are also the workflow scheduling algorithms that are described some of which are deadline constrained, ant colony, market oriented hierarchical etc. These surveys concluded that there is still a need for reliable and available resource scheduling algorithms as none of them focuses on both parameters [5].

4. Algorithms Introduced with Domains

4.1 Polynomial Time Algorithm

In the general setting, each server Sj is characterized by its capacity bj (i.e., the quantity of data that it can send, or the number of flops that it can process during one time unit, depending on the context) and its degree dj (i.e., the maximal number of open TCP connections, or the number of virtual machines that it can handle simultaneously). On the other hand, each client Ci is characterized by its demand wi (i.e., the number of tasks that it can process during one time unit, or its computational demand per time unit). Our goal is to build a bipartite graph between servers and clients, so that capacity, demand and degree constraints are satisfied [1].

4.2 Algorithm Based on Energy Consumption Methods

This algorithm is being implemented in Hadoop distributed file system with Energy Management and Regulation also called as GreenHDFS. This algorithm focuses on usage of the resources that are not fully utilized while execution of the environment. Due to fast development in technology the old methods of saving energy has been challenging. The works introduced till now are taken into account only with hardware but not with software. While this algorithm checks the energy consumption of the various computing resources that are involved in cloud like node, storage, switch and network. The resources CPU, main memory and storage has been worked till now and future work includes temperature and fan speed. Node in cloud computing is similar to servers composed of more than one multi core CPU which provides parallel services. The energy consumption depends on the type of the job whether compute intensive or I/O or storage. The clustering is done in a way to save energy. The user first chooses type of the job and then the job is in execution mode again the type is analyzed by counting the number of instruction execution speed. The basics of Round Robin algorithm are used. There are three phases in this algorithm: Infrastructure Preparation, Job Preprocessing and Job Execution. These estimates are approximate as the monitoring method used is not direct i.e. by sensors. This algorithm till now is implemented on Eucalyptus and data processing program is Hadoop. The readings of this algorithm are compared with the basic round robin algorithm in original environment [19].

4.3 Dynamic Priority Scheduling Algorithm (Service Request Scheduling)

This algorithm is applied on three tier containing service providers, consumers and resource providers. This algorithm gives more optimal then First Come First Serve (FCFS) and Static Priority Scheduling Algorithm (SPSA). The consumer response time for services has been tried to reduce in this algorithm as running instance is charged as it runs per unit time. The delays in provider side happens but are not counted under the cost charged to the customer so they need to be reduced. In three tiers there needs to be two scheduling: resource scheduling and service request scheduling. The FCFS concentrates on fairness to task units but it may result in low priority task units perform before than high priority tasks and SPSA makes task units prioritized before the process of scheduling. The DPSA evaluates task unit scheduled and recalculates and set task unit's priority thus optimizing the scheduling process. Though tasks has their initial priorities but the new priorities being set include SLA between user and cloud, task's features, task's source and operations in cloud. This algorithm considers three queues having highest priority, middle priority and lowest priority. Every queue has a threshold i.e. time a task unit will wait in particular queue. When the some task unit crossed that threshold value then the task unit automatically is moved to higher queue. When task reaches the highest queue it is send to the required component. Finally by comparing the average values and variance of priorities by processing time the DPSA comes out to be more efficient than FCFS and SPSA [13].

4.4 Scheduling by Employing Genetic Algorithm

This algorithm is proposed as a solution for Multi-objective optimization for virtual resources. When one request is made for any resource then the virtual resources scheduling is mapped onto physical resources with proper load balancing which is very complex to achieve. This algorithm is in comparison with rank, random and static algorithm. The layer of virtualization occurs between users and physical layer and it has three characteristics usability, safety and moving. They come from independency of virtualization. The virtual resources are abstracted by making number of instances of actual physical resource nodes with attributes. This algorithm is considered to be heuristic so it contains object functions, code and searching method. Object functions are there for load balancing calculation. NDSA II includes firstly non dominated sorting (set finding the lowest value of object function) and then crowd degree (lower). The GA has selection, crossover and mutation. In this algorithm the selection is tournament selection, crossover is two point crossovers and in mutation if the random number being chosen is the original gene is replaced by randomly generated one. By taking into consideration the CPU usage, memory and bandwidth the NDSA II comes out to be a better algorithm then rank, random and static algorithm as it provides many choices by running just once efficiently [18].

4.5 Optimizing Virtual Machine for High Performance Computing

Open stack is a scheduler which selects physical resources where VM is provisioned. Open stack receives VM request as part of RPC message. Host capability is an valuable input to scheduler which contains list of physical servers and their capabilities. The scheduling algorithm is of two steps: Filtering (exclude hosts which do not have required capability) and Weighing (computes fitness of filtered list using cost functions (e.g. free memory in a host)). Then by sorted list of hosts VM provisioning takes place. While scheduling Open Stack do not consider application type, priorities, processor heterogeneity and network topology. HPC-Aware Scheduler: There are two techniques involved: Topology awareness (as user is unknown of the cluster the VMs are packed to nodes in same rack compared to any placement policy which distribute them over the cluster) and hardware awareness/homogeneity (cloud users unaware of underlying hardware where VMs are placed by ensuring that all VMs are allocated some task). The first modification is to switch the use of group scheduling for considering k VMs problem as a single scheduling problem. Firstly topology aware algorithm runs as described next filtering phase (making a list then maximum number of servers) then using this build plan. For homogeneity the scheduler groups the hosts then applies algorithm to those groups taking into consideration the configuration (currently CPU frequency). The suitability of platform for an HPC application depends on application characteristics, performance requirements and user preferences. The main focus is HPC applications which are comprised of k parallel instances requiring synchronization and allocating VMs in topology aware manner to provide good list of VMs to application user. Its future work includes mixture of HPC and non-HPC applications [10] [14].

4.6 Dynamic Resource Scheduling and Workflow Management

An economic algorithm with business parameters determines the trade off between performance and effectiveness. An market oriented workflow architecture is introduced to meet customer demands and enhances the efficiency of the algorithms. This enhancement is done to improve dynamic algorithm with already predictive resource mechanisms. This solution helps in sustaining the consumers operations with different priorities [11].

4.7 Dynamic and Integrated Load Balancing Algorithm

It treats CPU, memory and bandwidth for physical as well as virtual machines. Total measurement of cloud data center imbalance level and average for server imbalance level are presented. This algorithm shows good performance with regard to imbalance level and overall running time as it has extra good features [12] [7].

4.8 Adaptive Optimal Global Resource Scheduling

It employs linear programming algorithm to reduce extra cost for power consumption and other expenditures with solid restrictions on networking environment. It promotes resource utility through finely grained resources and in depth restrains expenditure analysis for remote access by taking into account resource configuration, real time load and service deployment trade offing between performance, computation cost and response time. A greedy algorithm for small scale pool with many networking resources is provided [4].

4.9 Load Balance Based Algorithm

The data processing power of the nodes and data transferring power of the nodes and transfer delay between nodes is considered. Algorithm selects best node to complete the task to improve efficiency, minimize average response time of the tasks. These calculations are made on the basis of the dynamic load of the nodes in particular cloud. The prediction of time needed to complete the task is done resulting in increasing efficiency, reducing average response time and increasing throughput. The supposition that time to finish the task can be predicted is considered for this algorithm [3] [9].

4.10 Green Power Management for Virtual Machines

It includes 3 phases: Dynamic resource allocation mechanism, Virtualization management and green power management. The green power management is presented to reduce the load balancing for the virtual machine management. It supports green power mechanism applied on virtual machine resource monitor. Expected improvement contains violent CPU highly loading solution. It shows energy saving feature with setting of sensitivity parameters and also considers perfect smooth virtual changes [8].

4.11 Component Based Resource Allocation

This allocation model provides future resource allocation and managing need in cloud computing. The future perspective refers to whenever a new node is added to the cloud it combines them with the existing without much complication. The information generated by the component resource when new nodes are being added to the cloud will be of utmost importance. Functionality of node can be added to any component at any time to provide enhancements [20].

4.12 Pareto Based Optimal Scheduling

The cloud banking model is introduced with features like multi dimensional Pareto optimal theory and optimization analysis aiming at improving resource utilization as well as consumer satisfaction. This algorithm characterizes the user's requirements based on above mentioned features. It takes into consideration resource prices and execution time [17].

5. Results and Analysis

The following table summarizes scheduling strategies on scheduling method, parameters, and other factors. The different algorithms are working on same parameters at some cases. Each algorithm focuses on improving different parts of cloud environment. The differences are shown in Table 1.

Table 1: Different Scheduling Strategies/Algorithms

S. No	*Resource Allocation Strategy/Algorithm*	*Scheduling Parameters*	*Impacts*
1	Polynomial Time Algorithm	Throughput, Capacity, Degree Constraint	Number of clients assigned to a server is smaller than the server's degree and their overall demand is smaller than the server's capacity, while maximizing the overall throughput.
2	Algorithm based on Energy Consumption Methods	Energy consumption	It manages the performance challenges for resource allocation model.
3	Dynamic priority scheduling algorithm	Servicing delay	Priority of allocating resources is focused.
4	Scheduling by employing Genetic Algorithm	Time and Cost	Fault-tolerance is handled in allocating resources.
5	Optimizing Virtual Machine for High Performance Computing	Utilization of servers	It allows users to get the service at maximum level.
6	Dynamic resource scheduling and workflow management	Servicing Delay	The resource allocation is dynamically handled by managing on-demand requests.
7	Dynamic and integrated load balancing algorithm	Time and memory capacity.	It manages on-demand resource allocation among VM.
8	Adaptive optimal global resource scheduling	Resource cost, Memory usage.	Feedback control-based approach is developed for maximizing adaptive application QoS.
9	Load balance based algorithm	Memory capacity and usage.	It provides a way for balancing load on both client and server sides.
10	Green power management for virtual machines	Utilization of servers	It achieves overload avoidance and green computing.
11	Component based resource allocation	CPU, memory and network bandwidth	It treats multidimensional resource for both physical machines and virtual machines (VMs) for different scheduling objectives (algorithms).
12	Pareto based optimal scheduling	Resource price and execution time.	It presents PANDA framework for cloud bursting.

6. Advantages and Limitations

There are many advantages in resource allocation while using cloud computing irrespective of size of the organization and business markets. But there are some restrictions as well, since it is an emerging technology. Let's have a comparative look at the advantages and limitations of resource allocation in cloud environment [15] [16].

6.1 Advantages

1) The biggest benefit of resource allocation is that user neither has to install hardware nor software to access the applications, to develop the application and to host the application over the internet facilities.
2) The next major benefit is that there are no constraints of place and medium. We can reach our applications and data anywhere in the world, on any system.
3) The user does not need to expend on software and hardware systems.
4) Cloud providers can share their resources over the net during resource paucity.

6.2 Limitations

1) Since users rent resources from remote servers for their purpose, they don't have control over their resources.
2) Migration problem occurs, when the user wants to switch to some other provider for the better storage of their data. It is not easy to transfer huge data from one provider to the other.
3) In public cloud, the client's data can be susceptible to hacking or phishing attacks. Since the servers on cloud are interconnected, it is easy for malware to spread.
4) Peripheral devices like scanners or printers might not work with cloud. Many of them require software to be installed locally. Networked peripherals have not greater problems.
5) More and deeper knowledge is must for allocating and managing resources in cloud, since all knowledge about the working of the cloud mainly depends upon the cloud service providers.

7. Conclusion

Scheduling is one of the most important task in cloud computing environment. In this paper we have analyzed various scheduling algorithm and tabulated various parameters. We have noticed that disk space management is critical issue in virtual environment. Existing scheduling algorithm gives high throughput and cost effective but they do not consider reliability and availability. So we need to concentrate on algorithms that improves availability and reliability in cloud computing environment.

References

[1] Olivier Beaumont, Lionel Eyraud-Dubois, Christopher Thraves Caro, and Hejer Rejeb, “Heterogeneous Resource Allocation under Degree Constraints”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 5, May 2013.

[2] Mohamed Abu Sharkh, Manar Jammal, Abdallah Shami, and Abdelkader Ouda, “Resource Allocation in a Network-Based Cloud Computing Environment: Design Challenges”, IEEE Communications Magazine, 0163-6804/13, November 2013.

[3] Bernardetta Addis, Danilo Ardagna, Barbara Panicucci, Mark S. Squillante, and Li Zhang, “A Hierarchical Approach for the Resource Management of Very Large Cloud Platforms”, IEEE Transactions on Dependable and Secure Computing, vol. 10, no. 5, September/October 2013.

[4] Qian Zhu, and Gagan Agrawal, “Resource Provisioning with Budget Constraints for Adaptive Applications in Cloud Environments”, IEEE Transactions on Services Computing, vol. 5, no. 4, October-December 2012.

[5] Imad M. Abbadi, and Anbang Ruan, “Towards Trustworthy Resource Scheduling in Clouds”, IEEE Transactions on Information Forensics and Security, vol. 8, no. 6, June 2013.

[6] Hamzeh Khazaei, Jelena Mi_si_c, Vojislav B. Mi_si_c, and Saeed Rashwand, “Analysis of a Pool Management Scheme for Cloud Computing Centers”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 5, May 2013.

[7] Sheng Di, and Cho-Li Wang, “Dynamic Optimization of Multiattribute Resource Allocation in Self-Organizing Clouds”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 3, March 2013.

[8] Zhen Xiao, Weijia Song, and Qi Chen, “Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 6, June 2013.

[9] Ying Song, Yuzhong Sun, and Weisong Shi, “A Two-Tiered On-Demand Resource Allocation Mechanism for VM-Based Data Centers”, IEEE Transactions on Services Computing, vol. 6, no. 1, January-March 2013.

[10] Kyle Chard, and Kris Bubendorfer, “High Performance Resource Allocation Strategies for Computational Economies”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, January 2013.

[11] Hong Xu, and Baochun Li, “Anchor: A Versatile and Efficient Framework for Resource Management in the Cloud”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 6, June 2013.

[12] Mayank Mishra, Anwesha Das, Purushottam Kulkarni, and Anirudha Sahoo, “Dynamic Resource Management Using Virtual Machine Migrations”, IEEE Communications Magazine, 0163-6804/12, September 2012.

[13] Sharrukh Zaman, and Daniel Grosu, “A Combinatorial Auction-Based Mechanism for Dynamic VM Provisioning and Allocation in Clouds”, IEEE Transactions on Cloud Computing, vol. x, no. x, xxxx, 10.1109/TCC.2013.9.

[14] Wenhong Tian, Yong Zhao, Minxian Xu, Yuanliang Zhong, and Xiashuang Sun, “A Toolkit for Modeling and Simulation of Real-Time Virtual Machine Allocation in a Cloud Data Center”, IEEE Transactions on Automation Science and Engineering, 10.1109/TASE.2013.2266338.

[15] Luiz F. Bittencourt, Edmundo R. M. Madeira, and Nelson L. S. da Fonseca, “Scheduling in Hybrid Clouds”, IEEE Communications Magazine, 0163-6804/12, September 2012.

[16] Kyle Chard, Kris Bubendorfer, Simon Caton, and Omer F. Rana, “Social Cloud Computing: A Vision for Socially Motivated Resource Sharing”, IEEE Transactions on Services Computing, vol. 5, no. 4, October-December 2012.

[17] M. Reza Hoseiny Farahabady, Young Choon Lee, and Albert Y. Zomaya, “Pareto-Optimal Cloud Bursting, IEEE Transactions on Parallel and Distributed Systems”, 10.1109/TPDS.2013.218.

[18] Chun-Wei Tsai, and Joel J. P. C. Rodrigues, “Metaheuristic Scheduling for Cloud: A Survey”, IEEE Systems Journal, 10.1109/JSYST.2013.2256731.

[19] Jens Buysse, Konstantinos Georgakilas, Anna Tzanakaki, Marc De Leenheer, Bart Dhoedt, and Chris Develder, “Energy-Efficient Resource-Provisioning Algorithms for Optical Clouds”, J. OPT. COMMUN. NETW, vol. 5, no. 3, March 2013.

[20] Yu Hua, and Xue Liu, “Scheduling Heterogeneous Flows with Delay-Aware Deduplication for Avionics Applications”, IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 9, September 2012.

Author Profile

V. Manimaran completed his B.E. degree in Computer Science and Engineering from Nandha Engineering College, Erode, India in 2011. He is currently doing his M.E (Computer Science and Engineering) in Nandha Engineering College (Autonomous), Erode, India.

S. Prabhu completed his B.E. degree in Computer Science and Engineering from SSM College of Engineering, Komarapalayam, Salem, India in 2008. He completed his M.E degree in Computer Science and Engineering from Kongu Engineering College (Autonomous), Erode, India in 2010. He is currently pursuing his Ph.D. Programme on the area of cloud computing. Presently he is working as Assistant Professor in Computer Science and Engineering Department in Nandha Engineering College (Autonomous), Erode, India.