



# Review of Efficient Resource Scheduling Algorithms in Cloud Computing

**Nimisha Singla***Computer Science and Engineering Department  
Thapar University, India***Seema Bawa***Computer Science and Engineering Department  
Thapar University, India*

---

**Abstract** - In this paper the computability of the various resource scheduling algorithms in different aspects are described. Which algorithm is best topology wise, aspect and many others ways and why. This describes the optimal algorithm for various activities that include resource scheduling as an efficient ways to allocate resources in cloud computing which are proposed till now. The energy efficient, dynamic priority based, virtual resources model and so on. These include the latest algorithms that are devised to work over cloud and help in optimal resource scheduling.

**Keywords** - Energy efficient, dynamic priority allocation, virtual resources, high performance, parallel genetic, fault tolerance

---

## I. Introduction

Computing based on the internet sharing resources is called cloud computing. Cloud computing is the fastest growing technology in today's era offering ubiquitous services to users. The need of services to the lowest level is in demand. Nowadays everybody is not 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 managed. It delivers applications which are accessible from web browsers, desktop and mobile apps. Optimization of energy efficiency in cloud computing is necessary. It is a large scale computing using virtual resources. Its popularity is increasing 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 must in the cloud. Every server has limited resources so jobs/requests needs to be scheduled. Each application in the cloud computing 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 system. 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.

## II. Need for Resource Scheduling

There is enormous need for the cloud services to the schedule the resources as this scheduling will further followed by the job/task scheduling inside of the resources. There may be many 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 balanced among the resources of the same type. For the above parameters there need to be a procedure or function 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 best for the fully customizable configuration environment for application.

## III. Related Work

There have been a lot of work done on resource scheduling in cloud computing. New algorithms and management techniques for resource scheduling in cloud computing are being advised 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 job/task scheduling algorithms are introduced. The resource scheduling is been a tough job in cloud specially as it is the one which decides which process will be allocated to which resource and for how much time.

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 customer and providers. These parameters are execution time, policy, virtual machine, utility function, gossip, application, auction, hardware resource dependency, SLAs. While making a strategy the allocation methods should keep into consideration resource contention, fragmentation, over provisioning and under provisioning [4]. The various task scheduling methods in cloud computing are Cloud Service, User Level, Static and Dynamic, Heuristic, Workflow and Real Time scheduling [1]. Some of the scheduling algorithms in cloud whether or task or job or workflow or resources are Compromised-Time-Cost, Particle Swarm Optimization based Heuristic, Improved cost based for tasks, RASA workflow, Innovative transaction intensive cost constraint, SHEFT workflow, Multiple QoS Constrained for Multi- Workflows [2]. There are also the workflow scheduling algorithms that are described some of which are ant colony, deadline constrained, market oriented hierarchical etc [3]. These surveys concluded that there is still a need for reliable and available resource scheduling algorithms as none of them concentrates on both parameters.

#### **IV. Algorithms Introduced with Domains**

##### *4.1. Algorithm based on Energy Efficient Optimization Methods*

This algorithm is being implemented in Hadoop distributed file system with Energy Management and Regulation also called as GreenHDFS. This algorithm concentrates on usage of the resources that are not fully utilized while execution of the environment. Due to fast advancement in technology the old methods of saving energy has been challenging. The works introduced till now are taken into account 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 indirect 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 [5].

##### *4.2. Dynamic priority scheduling algorithm (Service request scheduling)*

This algorithm is applied on three tier containing service providers, resource providers and consumers. 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: service request scheduling and resource 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 [6].

##### *4.3. Non-Dominated Sorting Genetic Algorithm II*

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 [7].

#### *4.4. Optimizing Virtual Machine for High Performance Computing*

It is a HPC aware novel scheduler implemented on Open Stack Scheduler. It is topology aware and homogeneously allocating virtual machines. Cloud computing is of the lot of help to those who cannot afford large clusters has replaced supercomputers in some cases. Commodity interconnects performance variability and performance virtualization which indicates that cloud is suited for some HPCs. There are only few efforts on virtual machine algorithms that take into account the HPC. Open stack and Eucalyptus provide a minor effect of HPC. HPC aware strategies (topology awareness and hardware awareness) have been implemented which improves performance by allowing cloud providers to better utilize the infrastructure making more profits. Open stack is a scheduler which selects a physical resource where VM is provisioned. Open stack receives VM request as part of RPC message. Host capability is an important input to scheduler which contains list of physical servers and their capabilities. The scheduling algorithm contains 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 [8].

#### *4.5. Scheduling with Parallel Genetic Algorithm (PGA)*

This algorithm was devised to solve the problem of Unbalance Assignment problem to achieve the maximum efficiency. The existing strategies are not good to handle the scheduling so the GA turns out to be a good choice in case of scheduling. PGA improves performance and scalability. It can be implemented on parallel mainframes and heterogeneous computers. This algorithm helps in finding the best possible scheduling sequence on IaaS (Infrastructure as a Service) cloud giving better results than Rank algorithm, Round Robin algorithm, greedy technique, PBS and SGE. This algorithm contains three main steps: firstly system sets an idle resource and VM list, update each time VM requests occur. Secondly run PGA to get optimal allocation sequence. Thirdly allot physical machines to VM requests. Chromosome representation, fitness function and design are the according to the requests. The PGA operators include some of the parameters like topology, migration rate, and migration scheme and migration interval. It can run on more than one virtual machine at any instant. This algorithm till now has been implemented on Windows XP using Java Genetic Algorithm Package (JGAP). PGA improves the speed of resource allocation then Round Robin and Greedy algorithm [9].

#### *4.6. Balance Reduce Algorithm (BAR) (Fault Tolerant)*

This algorithm is based on data locality driven reducing network access thus reducing bandwidth usage and job completion time. This algorithm also handles the machine failure. Initial local task allocation in balanced phase takes place and then job execution time can be reduced by matching initial task allocation in reduced phase. The machine failure is handled by algorithm similar to primary backup approach.

This algorithm helps in reducing the required output by using the special key. In this algorithm the large computations can be broke into small pieces and run parallel in multiple machines inexpensively so a machine failure may lead to whole computation failure and thus leading to large job completion time if server fails. The BAR algorithm has two phases: Balance (all tasks allocated to preferred servers and make the data local i.e. read from servers local disk) and Reduce (balanced allocation is modified to minimize job completion time). Fault tolerant function is a reactive measure and solves failures. In this algorithm the failure detection needs to be done with some mechanism or device and it tolerates the failure of one node in the system. The fault tolerant algorithm reduces the make span to a quantifiable time from infinite time and tasks on the failed machine are rescheduled to other server for completion. As for the cloud based services are more of the data processing and network bandwidth is limited so the scheduling based on data locality is crucial [10].

#### *4.7. Heavy Traffic Optimal Algorithm*

The join-the-shortest-queue routing and power-of-two-choices routing with MaxWeight scheduling is optimal in throughput and they are queue length optimal in high traffic loads. Calculating the exact queue length is quite difficult so the system in heavy traffic regime (exogenous arrival rate is almost same as boundary of capacity region) was studied. Use of state space collapse (multi dimensional state reduces to single dimension) was there. The algorithm is applied on multiple models supported by multiple servers. Above models assume system is work conserving. Then the result converges to regulated Brownian notation and simple path optimality in scaled time. The method in the heavy traffic optimality is simpler and also in unscaled time consists of three steps: lower bound (weighted sum of expected queue

length by comparing with a single server queue), state-space collapse (state of system collapsing to single dimension, queue length in particular direction increases and in perpendicular direction it is bounded) and upper bound (obtained by natural Lyapunov function). Heavy traffic is obtained when upper and lower bound coincides. The solution contains one routing and one scheduling algorithm. This is the stochastic model for load balancing and scheduling in clusters. The JSQ and MaxWeight is throughput optimal and traffic optimal when all servers identical. And also the power-of-two-choices is also heavy traffic optimal [11].

#### *4.8. Optimized Resource Scheduling*

The optimization can be done on the basis of SLA (Service Level Agreement) as the resource scheduling problems are NP hard problems. The stochastic integer programming which further uses Grobner bases theory to extend Minimized Geometric Buchberger Algorithm is applied addressing SLA aware resource composition problem. This optimal solution is based on reasonable short time. This whole model used is further proposed for an optimal solution of resource composition model [12].

#### *4.9. Cost Based RS paradigm*

In this scheduling technique the resources are allotted as leveraging marketing theory to make the maximum use of the resources available. This algorithm as well as protocol is designed for IaaS. The allocation is according to the resource availability and price. This algorithm reduces overhead of running the algorithm in cloud environment resulting in perfect balance of complexity and performance. This whole algorithm is implemented on a private cloud environment [13].

#### *4.10. Double Auction-based Scheduling of scientific applications in Distributed Grid and Cloud environments*

This paper introduced a negotiation protocol between scheduler and resource manager using market-based Continuous Double Auction model. Different scheduling strategies are analyzed and based on it the results are demonstrated. It manages the access of resources in open market in order to minimize the cost of application execution. The self limitation based scheduling brings small improvements. Improvement in execution is achieved by aggressive strategy. The strategy can be chose by scheduler's attitude for risks [14].

#### *4.11. Algorithm based on Trust Degree*

This algorithm takes into consideration the functional characteristics and provides better stability and low risk whils completing tasks. It reduces threshold and risks in small and medium enterprises. The trust degree is determined by execution time and reliability. Scheduling logs stores trust degree at any time and sort it decreasingly and then the computer slots are called according to whose trust degree is greater first. This algorithm is stable and reliable [15].

#### *4.12. PSO based hierarchical strategy*

The Particle Swarm Optimization (PSO) based algorithm includes both transmission cost and present load. A novel inertia weight is also included for getting the global as well as local search effectively. This all supports minimizing inter network costs and balancing the load. This algorithm optimizes schedules of workflow application in cloud. The proposed algorithm yields effective performance on scheduling algorithm [16].

#### *4.13. Dynamic resource scheduling and workflow management*

An economic algorithm with business parameters determines the trade off between effectiveness and performance. An market oriented workflow architecture is introduced to meet customer demands and enhances the efficiency of the algorithm. This enhancement is done to improve dynamic algorithm with already predictive resource mechanisms. This solution helps in sustaining the consumers operations with different priorities [17].

#### *4.14. SLA Restriction for cloud based in Pareto Optimality $M \times N$ production*

This method is economic based with realization of optimal allocation of resources over a cloud. The strategy focuses on resource scheduling in Cloud Bank model solving problem of resource allocation among self interested individuals. It tries maximizes the benefits of the parties involved. Its allocation is dynamic and takes into consideration the time consumption nature of the scheduling as to cater to the needs of the users and it is one of the important parameter in scheduling algorithms. The user requirements are characterized by a utility function [18].

#### *4.15. Loyalty based resource allocation*

The trust concept is introduced in architecture and loyalty which improves the successful transaction rate of the system while meeting the requirements. Using Master Slave framework a role based access control is proposed considering the trust of the node and meets the requirements using the services. The unreliability of hardware should be provided by highly reliable software. It assesses the real time condition of the system and allocates resource according to condition. This dynamic feedback mechanism provides stability and reliability of services [19].

#### *4.16. Dynamic and integrated load balancing algorithm*

It treats CPU, bandwidth and memory for physical as well as virtual machines. Total measurement of cloud datacenter 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 [20].

*4.17. 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 [21].

*4.18. 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 [22].

*4.19. Green power management for virtual machines*

It includes 3 phases: Virtualization management, Dynamic resource allocation mechanism 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 considering perfect smooth virtual changes [23].

*4.20. 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 [24].

*4.21. 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 features. It takes into consideration resource prices and execution time [25].

*4.22. Smart Dynamic resource scheduling algorithm*

This algorithm contains 2 steps mainly prediction technique to fits the cloud data center well and novel and efficient migration technique to strike a balance between cost, instantaneity and efficiency. In this management prototype resource scheduling is just a module which has high forecast accuracy and can deal with load balancing and load consolidation. This paper presents a dynamic scheduling strategy which employs Single Exponential Smoothing (SES) algorithm for prediction of resource utilization and Vector Projection (VP) for second step. It doesn't consider spending of migration [26].

*4.23. Genetic algorithm with MultipleFitness*

It is a pre migration strategy which considers 3 parameters: disk I/O rate, network throughput and CPU utilization. For optimal solution a hybrid of genetic algorithm and knapsack problem is considered. This algorithm raises resource utilization and lower energy consumption cost by runtime resource scheduling under cloud environment. It smoothes load utilization [27].

*4.24. Hybrid multidimensional algorithm for network aware scheduling*

This distributed resource allocation algorithm is capable of handling multiple resources requisites for tasks that are arriving to computing environment. The tradeoff between execution time and cost of data intensive is considered by taking performance parameters at system and network level on economic and computational basis. It has an advantage of knowledge of grid infrastructure [28].

*4.25. Pricing algorithm*

Cloud bank as resource agency provides analysis and guidance for participants and a price update iterative algorithm analysis the historical utilization ratio and iteration current prices. It also gets the availability of resources and final price to consumers. This algorithm is designed to safeguard the interests of the participants in cloud. With this resources achieve macro control. It is not adaptive as it cannot adapt rapid changes of demand and supply. It reduces the cost of providers, maximizes revenue and is more conducive to keep the providers interests [29].

*4.26. Load adaptive model based on ant colony algorithm*

This algorithm monitors real timely virtual machines on performance parameters and schedules fast resources using any colony algorithm. It is made accordingly to bear load on a load free node to meet the changing load requirements improving resource utilization's efficiency. The detection of overload exceeds the threshold limit. This algorithm finds

the nearest idle node and allows it to bear some load meeting the performance and resource requirements of load thus achieving the goal of load balancing [30].

## V. Conclusion

As applicability of cloud computing nowadays is increasing and lots and lots of work is being done to reduce the cost incurred in providing services and making large profits so efficient scheduling is a must. There has been multiple scheduling algorithms introduced till now and every algorithm has its own applicability environment, characteristics, pros and cons. In these there has been no algorithm which takes care of reliability and performance without a fault. These algorithms are till now the optimal ones and further research has been done. There is need of more energy efficient algorithms in the future as cloud computing applicability is increasing more energy will be needed for the further increasing load in use and more of the services will be provided in future.

## References

- [1] Sujit Tilak, Prof. Dipti Patil, 'A Survey on Various Scheduling Algorithms in Cloud Environment', International Journal of Engineering Inventions, ISSN: 2278-7461, www.ijejournal.com, Volume 1, Issue 2 (September 2012), PP: 36-39.
- [2] Yogita Chawla and Mansi Bhonsle, 'A Study on Scheduling Methods in Cloud Computing', International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), ISSN 2278-6856, www.ijettcs.org, Volume 1, Issue 3, September – October 2012, PP: 12-17.
- [3] Amid Khatibi Bardsiri and Seyyed Mohsen Hashemi, 'A Review of Workflow Scheduling in Cloud Computing Environment', International Journal of Computer Science and Management Research, ISSN 2278-733X, www.ijcsmr.org, Vol. 1 Issue 3 October 2012, PP: 348-351.
- [4] V.Vinothina, Sr.Lecturer, Dr.R.Sridaran, Dr.PadmavathiGanapathi, 'A Survey on Resource Allocation Strategies in Cloud Computing', (IJACSA) International Journal of Advanced Computer Science and Applications, www.ijacsa.thesai.org, Vol. 3, No.6, 2012, PP: 97-104.
- [5] Liang Luo, Wenjun Wu and Dichen Di, Fei Zhang, Yizhou Yan, Yaokuan Mao, 'A Resource Scheduling algorithm of Cloud Computing based on Energy Efficient Optimization Methods', IEEE 978-1-4673-2154-9, Vol. 12 (2012).
- [6] Zhongyuan Lee, Ying Wang, Wen Zhou, 'A dynamic priority scheduling algorithm on service request scheduling in cloud computing', 2011 International Conference on Electronic & Mechanical Engineering and Information Technology, IEEE 978-1-61284-088-8, Vol. 11 (2011), PP: 4665-4669.
- [7] Jianfeng Zhao, Wenhua Zeng, Min Liu, Guangming Li, 'Multi-objective Optimization Model of Virtual Resources Scheduling Under Cloud Computing and It's Solution', 2011 International Conference on Cloud and Service Computing, IEEE 978-1-4577-1637-9 (2011), PP: 185-190.
- [8] Abhishek Gupta, Laxmikant V. Kale, 'Optimizing VM Placement for HPC in the Cloud', ACM 978-1-4503-1754-2, September 21, 2012, San Jose, California, USA, PP: 1-6.
- [9] Zhongni Zheng, Rui Wang, Hai Zhong, Xuejie Zhang, 'An Approach for Cloud Resource Scheduling Based on Parallel Genetic Algorithm', IEEE 978-1-61284-840-2 (2011), PP: 444-447.
- [10] Simy Antony, Soumya Antony, Ajeena Beegom A S, Rajasree M S, 'Task Scheduling Algorithm with Fault Tolerance for Cloud', International Conference on Computing Sciences, IEEE 978-0-7695-4817-3 (2012), PP: 180-182.
- [11] Siva Theja Maguluri and R. Srikant, Lei Ying, 'Heavy Traffic Optimal Resource Allocation Algorithms for Cloud Computing Clusters', ARO MURI W911NF-08-1-0233 and NSF grant CNS-0963807.
- [12] Qiang Li, Yike Guo, 'Optimization of Resource Scheduling in Cloud Computing', 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, IEEE 978-0-7695-4324-6/10, 2010, PP: 315-320.
- [13] Zhi Yang, Changqin Yin, Yan Liu, 'A Cost-based Resource Scheduling Paradigm in Cloud Computing', 12th International Conference on Parallel and Distributed Computing, Applications and Technologies, IEEE 978-0-7695-4564-6/11, 2011, PP: 417-422.
- [14] Radu Prodan, Marek Wiczcerek, Hamid Mohammad Fard, 'Double Auction-based Scheduling of Scientific Applications in Distributed Grid and Cloud Environments', J Grid Computing (2011) 9, PP:531–548.
- [15] Mingshan Xie, Mengxing Huang, and Bing Wan, 'A Resource Scheduling Algorithm Based on Trust Degree in Cloud Computing', Software Engineering Research, Management and Applications, 2012, PP: 177–184.
- [16] Hongli Zhang, Panpan Li, Zhigang Zhou, and Xiangzhan Yu, 'A PSO-Based Hierarchical Resource Scheduling Strategy on Cloud Computing', ISCTCS 2012, CCIS 320, PP: 325–332, 2013.
- [17] Xuelin Shi and Ying Zhao, 'Dynamic Resource Scheduling and Workflow Management in Cloud Computing', WISE 2010 Workshops, LNCS 6724, PP: 440–448, 2011.
- [18] Huixi Li and Hao Li, 'A Research of Resource Scheduling Strategy with SLA Restriction for Cloud Computing Based on Pareto Optimality M×N Production Model', WISM 2011, Part I, LNCS 6987, PP:155–165, 2011.
- [19] Yanbing Liu, Shasha Yang, Qingguo Lin, and Gyoung-Bae Kim, 'Loyalty-Based Resource Allocation Mechanism in Cloud Computing', Recent Advances in CSIE 2011, LNEE 125, PP: 233e–238.
- [20] Wenhong Tian, Yong Zhao, Yuanliang Zhong, Minxian Xu, Chen Jing, 'A DYNAMIC AND INTEGRATED LOAD BALANCING SCHEDULING ALGORITHM FOR CLOUD DATACENTERS', Proceedings of IEEE CCIS2011, 978-1-61284-204-2/11, 2011, PP:311-315.

- [21] Lingli Deng, Qing Yu, and Jin Peng, 'Adaptive Optimal Global Resource Scheduling for a Cloud-Based Virtualized Resource Pool', STA 2011, CCIS 186, PP: 231–240, 2011.
- [22] Haihua Chang and Xinhuai Tang, 'A Load-Balance Based Resource-Scheduling Algorithm under Cloud Computing Environment', ICWL 2010 Workshops, LNCS 6537, PP: 85–90, 2011.
- [23] Chao-Tung Yang, Kuan-Chieh Wang, Hsiang-Yao Cheng, Cheng-Ta Kuo and Ching-Hsien Hsu, 'Implementation of a Green Power Management Algorithm for Virtual Machines on Cloud Computing', UIC 2011, LNCS 6905, PP: 280–294, 2011.
- [24] Sumeet S. Vernekar and Pravin Game, 'Component Based Resource Allocation in Cloud Computing', Proceedings of the InConINDIA 2012, AISC 132, PP: 907–914, 2012.
- [25] Hao Li and Guo Tang, 'Pareto-Based Optimal Scheduling on Cloud Resource', ICHCC 2011, CCIS 163, pp. 335–341, 2011.
- [26] Lei Xu, Wenzhi Chen, Zonghui Wang, Shuangquan Yang, 'Smart-DRS : A Strategy of Dynamic Resource Scheduling in Cloud Data Center', IEEE International Conference on Cluster Computing Workshops, IEEE 978-0-7695-4844-9/12 (2012), PP: 120-127.
- [27] Shi Chen, Jie Wu, Zhihui Lu, 'A Cloud Computing Resource Scheduling Policy Based on Genetic Algorithm with Multiple Fitness', IEEE 12th International Conference on Computer and Information Technology, IEEE 978-0-7695-4858-6/12 (2012), PP: 177-184.
- [28] D. Adami, C. Callegari, S. Giordano, M. Pagano, 'A Hybrid Multidimensional Algorithm for Network-aware resource Scheduling in Clouds and Grids', IEEE ICC 2012-Communication QoS, Reliability and Modeling Symposium, IEEE 978-1-4577-2053-6/12 (2012), PP: 1297-1301.
- [29] Hao Li, Jianhui Liu, Guo Tang, 'A Pricing Algorithm for Cloud Computing Resources', International Conference on Network Computing and Information Security, IEEE 978-0-7695-4355-0/11 (2011), PP: 69-73.
- [30] Xin Lu, Zilong Gu, 'A LOAD-ADAPATIVE CLOUD RESOURCE SCHEDULING MODEL BASED ON ANT COLONY ALGORITHM', Proceedings of IEEE CCIS2011, IEEE 978-1-61284-204-2/11 (2011), PP: 296-300.