International Journal of Modern Computer Science (IJMCS) ISSN: 2320-7868 (Online)
Volume 4, Issue 4, August, 2016

Task Scheduling: The Technological Changes

Gaurav Chaudhary1[st]
Research Scholar
CGC-College of Engineering, Mohali, 140307, India
Email: gaurav.chaudhary668@gmail.com

Mr. Manish Mahajan 2[nd]
Assist. Professor and H.O.D. Computer Science
CGC-College of Engineering, Mohali, 140307, India
Email: hod.coecse@cgc.edu.in

Abstract--Cloud computing is creating as a perspective for immense scale data-oriented applications. Cloud computing is based on the headway of distributed and grid computing and virtualization. As per the cost of every job in cloud resources is assorted with each other, scheduling of jobs/tasksis not same as in standard scheduling techniques. Job/Task scheduling is a vital and critical problem of Cloud computing. Everyjob/task scheduler in Cloud computing have to meet the satisfaction cloud customers with the doled outQoS and upgrade advantages of cloud suppliers. Developments are important to ride the unavoidable tide of progress. The time of cloud based multimedia applications has prompted a tremendous increment in the no. of requests on cloud. The expanded no. of requests on cloud prompts an expanded workload, making workload balancing a vital QoS Parameter. Workload Balancing additionally prompts a sensible utilization of assets like electricity and so on and therefore advances the idea of Green IT. Job/task scheduling Cloud computing is a NP-hard issue for which, various methods and models have been developed and proposed, ranging from low level to high level execution of tasks in multiple processors. In this paper different task scheduling techniques are said that have been made over the earlier years to meet the vital QoS and changing customer needs.

Keywords-- Cloud Computing, Distributed Computing, Grid Computing, Virtualization, Task Scheduling.

I. INTRODUCTION

In view of the development of system virtualization technologies and Internet, one other computing stage has risen as Cloud Computing. Cloud computing works for providing resources to user as services by making use of the Internet and Virtualization [1]. Cloud computing is another strategy in the IT circle. It is an expansion of grid, parallel and distributed computing. It gives speedy, highly secured and advantageous computing and storage service cantered by Internet. Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) are the three sorts of services that are given by Cloud computing. All of these services and platforms are available to the users as an On-Demand model, which is fit for making utilization of shared and consequently distributed IT assets like server machines, information storage systems, network, applications thus on with the assistance of internet [2].

Cloud Computing is another worldview which empowers relationship to grow or cut back their usage of IT assets considering their necessities without endeavoring any attempts in foundations. As in cloud computing every resource resides in a shared network so an administrative system is necessary which can dispense jobs to different units at whichever point necessary, manage and maintain requests, re-figure the responses reverted by the units, oversee the availability of data to all the units and so forth. We live in the period of computing where advancement is a ceaseless procedure subsequently the cloud foundations are additionally developing. Task Scheduling is a champion amongst the most basic activities of the asset organization, if workload is not supervised accurately it can incite tremendous concedes in techniques [3]. Giving resources to cloud clients is a non-trivial task which is a NP-hard issue. An un-optimized allocation will negatively affect the general performance of the system particularly the operation cost and revenue. Keeping in mind the end goal is to make more feasible cloud foundation energy utilization by the data centers turns into an important component which is more difficult to unravel. The unlimited resource limit of data centers fulfills a large portion of clients' requests. Nonetheless, these data centers expend a tremendous amount of energy and they are one of the variables bringing on the greenhouse effect. It has been noticed that the utilization of power by data centers has crossed 10.000.000.000 kilowatt-hours (kWh) and has created more than 40.568.000 tons of CO2 emission, by 2011[4].

Cloud computing gives utilization readiness, lower capital use, region autonomy, resource pooling, broad network access, unwavering quality, adaptability, flexibility, and simplicity of support. The scheduling of a workflow in cloud computing systems is clearly difficult and complex when the virtualized systems are used to execute a large number of jobs. Consequently, various methods and models have been developed and proposed, ranging from low level to high level execution of tasks in multiple processors [7]. Starting late, various papers are distributed which use transformative algorithms like particle swarm optimization (PSO), different cost based algorithms, bee colony algorithm (BCA) ant colony optimization (ACO), simulated annealing (SA) and genetic algorithm (GA) out of which genetic algorithm (GA) has been shown extremely effective [9].

In this paper several task scheduling algorithms are listed to understand how the things have been changed along the past several years. This paper consist of six sections, in the first section the abstract of the work is given then in the second section contains the introduction then the literature survey and then in the next section all the task scheduling techniques are listed and after that section the conclusion of the paper is mentioned followed by the references to all the papers used in the making of this paper.

II. LITERATURE SURVEY

Market-based asset utilization has become much thought from the exploration community because it is a critical issue of tremendous scale appropriated systems like clouds [5]. Buyyaet al. exhibited in [17] a method for asset deployment in grid using economic-based thoughts which also includes product market, contract Internet models, bargain modeling, posted price modeling, and so forth. This method has been realized in Nimrod-G [15], which is an economy driven asset manager that deals with all operations related to remote execution including asset disclosure, trading, economic based scheduling and a customer'sQoSrequirements. Silva et al. begat in [8] a model on utility computing base for resources allocation. This model streamlines the quantity of systems designated to process jobs and accelerate the execution within an economic restriction. Regardless, every one of them make the inalienable supposition that a market-based method is fundamentally better which is adhoc as the allocation depends upon various parts other than demand and supply, for instance, bandwidth, communication delays, server speeds, and so forth. Moreover, they concentrate just on boosting the aggregate revenue of asset proprietor without considering the energy utilization of resources apportioned to customers.

Closeout based asset utilization has been associated in computing resulting to 1968 when Sutherland made use of auctions for giving the CPU time in a single computer system[18]. Starting now and into the foreseeable future, specialists have tested the closeout models for resource utilization in computational bases. Gomoluch and Schroeder [6] reproduced the double closeout tradition for asset utilization in grids. Their outcomes exhibit that the double closeout tradition beats the standard round-robin approach. In [16], Das and Grosugiven a combinatorial closeout based asset utilization tradition in which a customer bids and announces a price esteem for each and every possible outcomes of assets needed for its execution in grid. The tradition incorporates an approximation algo for dealing with the combinatorial closeout issue. They used a method where various grid suppliers can give particular sorts of physical computing assets. In [19], Zaman and Grosu showed other combinatorial closeout method for assets of a single supplier (i.e., allot the virtual machine instances to customers).

Pandey, Linlin [20] have built up a Particle Swarm Optimization (PSO) based methods to timetable applications to cloud assets that considers both data transmission and calculation cost. Mezmaz, Melab [21] have proposed another parallel bi-objective hybrid genetic algorithm based on DVS (Dynamic Voltage Scaling) that considers, makespan, and in addition energy utilization. Abrishami and Naghibzadeh [22] have given another QoS-based task scheduling algo by utilizing the idea called PCP (Partial Critical Paths), which works for minimizing the expanse of work process execution while satisfying a customer described due date. Wang, Zeng [23] have presented a scheduling algorithm called Cloud-DLS (Dynamic Level Scheduling) and a Bayesian strategy based subjective trust model by planning the present DLS algorithm [24]. Wu, Liu [25] have presented a market-oriented class-conscious scheduling method. Liu, Qu [26] have presented a fuzzy clustering strategy to process the cloud assets in advance. They also have presented another directed acyclic graph based scheduling algo called EFTD algorithm (Earliest Finish Time Duplication) for distributed cloud systems by consolidating the rundown scheduling with the job duplication scheduling approach. Choudhary and Peddoju [27] have presented a scheduling algo which considers the genuine difficulties of job scheduling. As per this algo, the input work process are combined on the premise of the work process prerequisite like slightest of the time taken for execution or most reduced cost and organized. By then, asset decision is done on the premise of job requirements using a greedy approach. At long last, Jung, Lim [28] has proposed the work process scheduling plot that decreases the job waiting time when an occasion happens the out-of-bid circumstance.

III. III. DIFFERENT TASK SCHEDULING TECHNIQUES

A. *Genetic Algorithm Based Scheduling*

As we probably aware as of now a cloud client achieves a SLA (Service Level Agreement) with a supplier to process a job. A SLA document comprises of client requisites like time and budget constraints of the job, which demonstrate adequate due date and final expanse value of the client. QoS parameters like throughput and response time can also be contained in a SLA document. A cloud supplier needs to pay attention on client requisites and virtual machine status before designating work process to virtual machines. This genetic based task scheduling algorithm can be utilized to take care of different task issues. In the proposed job scheduling method, the job scheduler starts the GA scheduling function to timetable work process based on information of work process and virtual machines. The GA scheduling function makes a large set, an arrangement of jobs, and ascertains the set by utilizing the fitness function considering fulfillment client requests and the availability virtual machine. The function rehashes repeating set to output the best job plan. To the GA scheduling function the restart operation is additionally forced for the change in quality of work process plans [1].

B. *Activity Based Cost Algorithm*

Activity based costing is a strategy for finding both the cost of the assets and the computational performance. In cloud computing, every application run on a virtual platform, where the assets will be spread fundamentally. Each application is entirely assorted and is free and has no association with any other by any stretch of the imagination, for example, some require additional processor time to process complex jobs, and some might require extra memory to store information, etc. Assets are surrendered on jobsexecuted on each individual unit of service. Remembering the final objective to gauge direct costs of applications, every single utilization of assets (like CPU, memory and Input/yield cost and so on) ought to be calculated. Right when the immediate data of every particular assets cost has been calculated, more precise cost and advantage examination based on it can be generated. The scheduler recognizes number of jobs, average MI of jobs, deviation percentage of MI granularity size and CPU overhead of the extensive number of jobs. Assets are picked, jobs are arranged by priority, and they are set in three particular records based on three levels of priority - high, medium and low priority. Presently task grouping algois associated with the above records with a particular final objective to distribute the job-groups to different available assets [2].

C. *Tree Based Scheduling Technique Using K-way Trees*

The cloud includes vast number of virtual machines with different limit of processing, the route in which the jobs are given to the virtual machines impacts the time of execution and moreover the time the cloudlets are in waiting state. As demonstrated by this protocol the cloudlets and the virtual machines are at first arranged in descending order. The cloudlets are arranged based on their number of instructions and the virtual machines are arranged according to the MIPS value which states what number of instructions that respective virtual machine can execute per second of time. The pre-processing of both the virtual machines and the cloudlets is a bit of this algo for the reason that organizing the immense cloudlets over the minimal one [3].

D. *Green Resource Allocation Auction Model*

As indicated by this method, we expect that there are two sorts of elements in cloud: clients and suppliers. For effortlessness, concentrate looking into it of a single supplier attempting to assign resources to clients. Supplier has k sorts of resources signified by set. For every kind of resource, there are an aggregate of unit instances prepared for utilization. To understand thisthe instances of the standard virtual machine (VM) by Amazon EC2 can be considered which are small, medium, large and extra-large.

Relating to every sort of assets, there is a coefficient ei∈R+ corresponds to the unit cost for energy required or expended per unit time. For Instance, a "small" Amazon EC2 VM case which is running Linux-based OS utilizes 150W and costs $0.085 every hour while an "extra-large" case with Windows utilizes 630W and costs $2.88 every hour. Without losing sweeping statement, we accept that the base time of ei is equivalent to the span of an auctioning period. This will bar the time constraints from the method and disentangle it. There are n clients, each asking for an immense amount of resources and states a worth that shows the amount she/he will pay for that bundle. Given the arrangement of clients and their bids, then the focus and purpose of this closeout based issue are to (i) decide the arrangement of champs and (ii) figure the installment to be paid by every triumphant user [4].

E. *Cuckoo Search Algorithm*

CSA algo works as per the commit brood parasitic nature of some cuckoo species in union with the Lévy flight phenomenon of few birds.

Cuckoo Breeding Behavior: Levy flight phenomenon is used by Cuckoos to lay their eggs in common nests. They sometimes select a nest where the owner bird simply has its own eggs. To build the probability of hatching their own eggs they may evacuate alternate eggs. The patterns and colors of the eggs of initial species are sometimes imitated by a few cuckoos. Cuckoos tries to minimize the chances of the eggs being deserted to amplify their re-profitability. Accordingly, at whatever point a host bird finds a peculiar egg, will either leave the nests or go to another spot to assemble another one or simply discard them.

Lévy Flights Phenomenon: The common nature of animals to hunt sustenance is in a quasi-random way. By and large, the scavenging way of an animal is viably a random walk in light of the fact that the following stride depends on both the present area/state and the move probability to the following area. Numerically, it is anything but difficult to model the determination of next location as it depends on probability. Different studies have demonstrated that the ordinary attributes of Lévy flights are exhibited in the flight example of numerous animals. Therefore, such examples are actualized for optimization and ideal pursuit, and experimental results demonstrate its promising capability.

Cuckoo Search Algorithm (CSA): CSA is a meta-heuristic way that simulates the common nature of cuckoos and take after the accompanying tenets:

- At a time every cuckoo lays a single egg and drops it in a nest which is chosen randomly.
- Only the perfect quality eggs are taken to the next stage which is browsed out of the best nests.
- Suppose n is the quantity of accessible hosts which is foxed, and the probability of finding an outsider egg by the host is Pa [0, 1].
- In request to fabricate another nest in another area, the host bird can either forsake the entire nest or discard the egg [7].

Algorithm 1. Pseudo code of the CSA

Objective function $f(x)$, $x=(x_1, \ldots, x_d)$;
Initial a population of n host nests xi (I =1,2,…,n);
while (t < Maximum Generation) or (stop criterion);
Get a cuckoo (say I) randomly and generate a new solution by Lévy flights;
Evaluate its quality/fitness; F_i
Choose a nest among n (say j) randomly;
if ($F_i > F_j$),
Replace j by the new solution; end
Abandon a fraction (Pa) of worse nests
[and build new ones at new locations via Lévy flights];
Keep the best solutions (or nests with quality solutions);
Rank the solutions and find the current best;
End while
Post process results and visualization

F. A Modified Genetic Algorithm

In SGA, the initial set is produced randomly. As the outcomes are created randomly, there are fewer risks of them being proficient and consequently the youngster outcomes produced from these wasteful solutions won't be much powerful. As per this work initial population is made with the techniques created by Enhanced Max-Min algo. At that point we have connected the processes of crossover and mutation on this elementaryset and applied the new created schedules to the set. This procedure is rehashed until we get the size of the set as 100. At last the schedule from the wholeset which shows minimum makespan is picked. In enhanced max-min, biggest job is booked to slowest asset and rest of all jobs is executed by speedy assets. In this particular instance, general makespan is expanded. Since here the estimation ofmakespan is reliant on anticipated execution time of biggest task. In the event that this reliance is expelled, at exactly that point, makespan can be optimized for this particular case. So as opposed to relegating biggest task to slowest resource, in the event jobs scheduled in a manner that the slowest asset is allotted to the job with average execution time this can optimize the result [9].

G. MQoS-GAAC algorithm

Due to the solving efficiency in the later period of genetic algorithms is not high, so it is easy to lead to a huge amount of common iterations. And in the preliminary stage of ACO, lacking of pheromone will lead to accumulate pheromone for a long time in the initial search. Thus, combining GA and ACO and integrating QoS proposed MQoS-GAAC algorithm (genetic algorithm and ant colony optimization algorithm with the multi-dimensional QoS constraints). The algorithm is divided into three parts, details are as follows:

Part 1: Make full use of global fast search of GA and convert the obtained global search information into the initial pheromone of ACO.

Part 2: Judge the appropriate time for GA and ACO integration.

Part 3: Use the positive feedback of ACO and the characteristics of efficient convergence to finish the optimal cloud task scheduling which meets the QoS constraints [11].

H. Bag-of-Task Scheduling

A bag-of-tasks application is modeled as a job J that can be divided into n parallel tasks {$task_1$, task2, ⋯, taskn} without precedence constraints and inter-task communications. This independent feature makes bag-of-tasks applications suitable to be executed on multiple resource providers to downgrade the final execution time. The total execution time of J is called the schedule length or the Makespan, which is the duration from the start of the first job to the completion of the last job. Scheduling and executing bag-of-tasks in cloud are taken out on a large pool of virtualized computing resources, which are assigned according to slots such as time slot in Amazon EC2 and Microsoft Azure. In this case, Makespan not only measures system performance but also affects users' bill of renting computing resources. Therefore, the optimization of Makespan is still an essential goal of scheduling.

Task scheduling can have multi-objectives, the scope of which is defined by constraints. When energy consumption is taken into account, energy can be saved by exploring the potentials of energy-efficient task arrangements within deadlines. Since tasks exhibit an immense variety in genuine execution time, and by and large, just expend a little part of their worst case execution time, unused time slice can be viewed as slack and distributed by tasks to run slower for energy efficiency. Figure 1 exhibits the component of slack reclamation. Job J comprises of two autonomous tasks task1 and task2. Fig. 1(a) outlines that the completion time of J is a long way from due date. Subsequently, in Figure 1(b), task1 and task2 are scheduled to execute at lower frequencies by DVFS to spare energy [12].

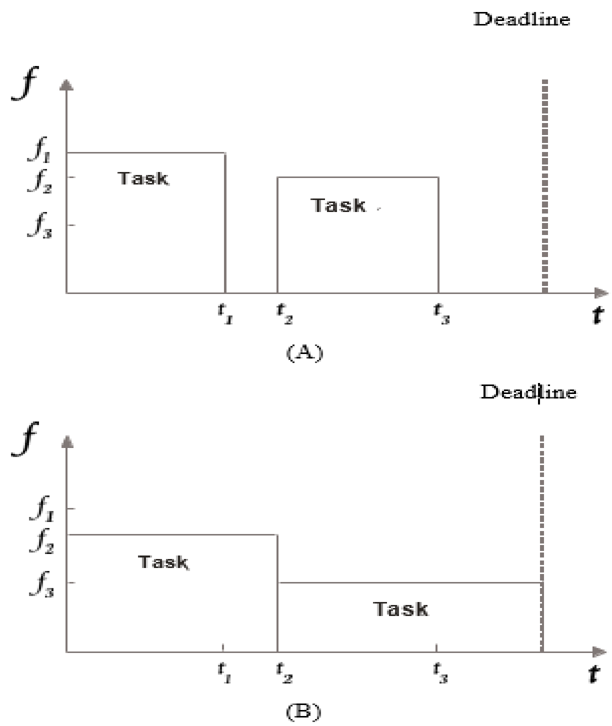

Figure 1: Slack Reclamation. (A) Before reclamation. (B) After reclamation.

I. Load Balancing Resource Scheduling

Resources are categorized on the basis of two pricing schemes: the Reservation Scheme and the On-Demand Scheme. These resources are further classified into three classes with the Virtual Machine (VM) instances having different resource costs and different service times respectively [13]. The research work presented in this paper considers the importance of load balancing in the scheduling process and presents a new load balanced resource scheduling (LBRS) algorithm.

The proposed approach performs the following steps:

- Compute the cost per service rate of each reserved Class i VM instance, q_i, and arrange it in ascending order
- Schedule the reserved VMs on the basis of order computed in step 1
- If all reserved VMs are consumed, calculate the no. of requests left to be served
- Calculate the required no. of on-demand resources from the respective available classes n the inverse proportion of the ratio of their respective service rates and cost per VM
- Optimize the ratio obtained in previous step
- Schedule the requests on the basis of above ratio
- Repeat the steps 3 to 6 till all requests are served [14].

J. Min-Energy CSB Demand Allocation Algorithm

This algorithm explains how a CSB allocates its tenants requests to achieve the objective of minimizing the energy cost. Due to the hardness of this problem, we then propose two time-efficient algorithms: a greedy algorithm, called Balance Fit Decreasing (BaFD), based on the Best Fit Decreasing (BFD) algo, and an approximation algorithm through linear programming (LP) relaxation. BaFD is improved from the existing BFD algo, a natural non-algorithmic for one-dimensional bin packing issue. BFD assembles the objects in descending order of size. Starting from the first object, it iterates over the bins, finds out the bin that has the least amount of space left after accommodating the object. It then proceeds to the second object, and repeats the same procedure until all the objects are packed.

The greedy algorithm can efficiently get a suboptimal solution for MCD, but it has no performance guarantee for its solution, i.e., how the worst result can be compared Algorithm to the optimal solution. Hence, in this section, we devise an approximation algorithm using LP-relaxation. LP-relaxation is a technique that relaxes an NP-hard LIP into a related LP problem that is solvable in polynomial time, and the solution to the relaxed LP can be used to gain the solution to the original ILP[29].

IV. CONCLUSION

Although cloud computing is being used today at a very large scale it is still in its infancy state. Since the amount of data involved in cloud based transactions or simply cloud based work is very huge, a number of new data handling and scheduling methods are being introduced every day. In this paper a number of different task scheduling methods and models has been discussed which are used to handle and schedule the huge amount of data available on the cloud. Although the methods and models listed in this paper and the others that has been developed so far have proved to be good enough and have given fairly nice results, the amount of data over the cloud is still growing every second and so as the need of the development of new data handling and scheduling methods.

REFERENCES

[1]. Sung Ho Jang, Tae Young Kim, Jae Kwon Kim and Jong Sik Lee, "The study of genetic algorithm-based task scheduling for cloud computing." International Journal of Control and Automation vol. 5.4 pp. 157-162, 2012.

[2]. S.Selvarani, G.SudhaSadhasivam, "Improved cost-based algorithm for task scheduling in cloud computing." Computational intelligence and computing research (iccic), 2010 ieee international conference, pp. 1-5, 2010.

[3]. AkshiBhutani, IshaJauhari, Vinay Kumar Kaushik, "Optimized virtual machine tree based scheduling technique in cloud using K-way trees." Cognitive Computing and Information Processing (CCIP), 2015 International Conference on. IEEE, pp. 1-6, 2015.

[4]. Tram Truong Huu, Chen-KhongTham, "An auction-based resource allocation model for green cloud computing." Cloud Engineering (IC2E), 2013 IEEE International Conference on. IEEE, pp. 269-278, 2013.

[5]. Anubha Jain, Manoj Mishra, Sateesh Kumar Peddoju, Nitin Jain, "Energy efficient computing-green cloud computing." Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on. IEEE, pp. 978-982, 2013.

[6]. J. Gomoluch, M. Schroeder, "Market-based resource allocation for grid computing: A model and simulation," The First International Workshop on Middleware for Grid Computing (MGC'2003), Rio de Janeiro, Brazil, pp. 211–218, 2003.

[7]. NimaJafariNavimipour and FarnazSharifiMilani, "Task scheduling in the cloud computing based on the cuckoo search algorithm." International Journal of Modeling and Optimization, vol. 5.1 pp. 44, 2015.

[8]. J. N. Silva, L. Veiga, and P. Ferreira, "Heuristic for resources allocation on utility computing infrastructures," 6th International Workshop on Middleware for Grid Computing (MGC 2008), Leuven, Belgium, pp. 1 – 6, December 2008.

[9]. Shekharsingh, Mala Kalra, "Scheduling of Independent Tasks in Cloud Computing Using Modified Genetic Algorithm." Computational Intelligence and Communication Networks (CICN), 2014 International Conference on. IEEE, pp. 565-569, 2014.

[10]. Tingting Wang, ZhaobinLiu, Yi Chen, YujieXu, Xiaoming Dai, "Load balancing task scheduling based on genetic algorithm in cloud computing." Dependable, Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on. IEEE, pp. 146-152, 2014.

[11]. Yangyang Dai, Yuansheng Lou, Xin Lu, "A Task Scheduling Algorithm Based on Genetic Algorithm and Ant Colony Optimization Algorithm with Multi-QoSConstraints in Cloud Computing" Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2015 7th International Conference on IEEE, vol. 2, pp. 2015.

[12]. Yujian Zhang, Yun Wang and Cheng Hu, "CloudFreq: Elastic Energy-Efficient Bag-of-Tasks Scheduling in DVFS-enabled Clouds." Parallel and Distributed Systems (ICPADS), 2015 IEEE 21st International Conference on. IEEE, pp. 585-592, 2015.

[13]. RituKapur, "A cost effective approach for resource scheduling in cloud computing." Computer, Communication and Control (IC4), 2015 International Conference on. IEEE, pp. 1-6, 2015.

[14]. RituKapur, "A workload balanced approach for resource scheduling in cloud computing." Contemporary Computing (IC3), 2015 Eighth International Conference on. IEEE, pp. 36-41, 2015.

[15]. D. Abramson, R. Buyya, and J. Giddy, "A computational economy for grid computing and its implementation in the Nimrod-G resource broker," Future Generation Computer Systems, vol. 18, no. 8, pp. 1061– 1074, 2002.

[16]. A. Das and D. Grosu, "Combinatorial auction-based protocols for resource allocation in grids," 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05), Denver, Colorado, April 2005.

[17]. R. Buyya, D. Abramson, J. Giddy, and H. Stockinger, "Economic Models for Resource Management and Scheduling in Grid Computing," The Journal of Concurrency and Computation, vol. 14, no. 13–15, pp. 1507–1542, 2002.

[18]. I. E. Sutherland, "A futures market in computer time," Communications of the ACM, vol. 11, no. 6, pp. 449–451, 1968.

[19]. S. Zaman and D. Grosu, "Combinatorial Auction-Based Allocation of Virtual Machine Instances in Clouds," 2010 IEEE Second International Conference on Cloud Computing Technology and Science, Indianapolis, USA, pp. 127–134, 2010.

[20]. S. Pandey et al., "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments," Proc. 2010 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), 2010.

[21]. M. Mezmaz et al., "A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems," Journal of Parallel and Distributed Computing, vol. 71, no. 11, pp. 1497-1508, 2011.

[22]. S. Abrishami and M. Naghibzadeh, "Deadline-constrained workflow scheduling in software as a service cloud," ScientiaIranica, vol. 19, no. 3, pp. 680-689, 2012.

[23]. W. Wang et al., "Cloud-DLS: Dynamic trusted scheduling for Cloud computing," Expert Systems with Applications, vol. 39, no. 3, pp. 2321-2329, 2012.

[24]. A. Dogan and F. Ozguner, "Matching and scheduling algorithms for minimizing execution time and failure probability of applications in heterogeneous computing," IEEE Transactions on Parallel and Distributed Systems, vol.13, no. 3, pp. 308-323, 2002.

[25]. Z. Wu et al., "A market-oriented hierarchical scheduling strategy in cloud workflow systems," The Journal of Supercomputing, vol. 63, no.1, pp. 256-293, 2013.

[26]. Z. Liu et al., "Resource preprocessing and optimal task scheduling in cloud computing environments," Concurrency and Computation: Practice and Experience, 2014.

[27]. M. Choudhary and S. K. Peddoju, "A dynamic optimization algorithm for task scheduling in cloud environment," International Journal of Engineering Research and Applications (IJERA), vol. 2, no. 3, pp. 2564-2568, 2012.

[28]. D. Jung et al., "A workflow scheduling technique for task distribution in spot instance-based cloud," Ubiquitous Information Technologies and Applications, Springer Berlin Heidelberg. pp. 409-416, 2014.

[29]. ChenxiQiu, HaiyingShen, Liuhua Chen "Towards green cloud computing: Demand allocation and pricing policies for cloud service brokerage." Big Data (Big Data), 2015 IEEE International Conference on. IEEE, pp. 203-212, 2015.