

# Comparative Based Study of Scheduling Algorithms for Resource Management in Cloud Computing Environment

Rajesh Verma

Dept. of Computer Application, International Institute of Professional Studies, DAVV, Indore, India

Available online at [www.isroset.org](http://www.isroset.org)

Received: 08 May 2013

Revised: 28 May 2013

Accepted: 06 June 2013

Published: 30 June 2013

**Abstract**— Cloud Computing (CC) is emerging as the next generation platform which would facilitate the user on pay as you use mode as per requirement. It provides a number of benefits which could not otherwise be realized. The primary aim of CC is to provide efficient access to remote and geographically distributed resources. A scheduling algorithm is needed to manage the access to the different resources. There are different types of resource scheduling technologies in CC environment. These are implemented at different levels based on different parameters like cost, performance, resource utilization, time, priority, physical distances, throughput, bandwidth, resource availability etc. In this research paper various types of resource allocation scheduling algorithms that provide efficient cloud services have been surveyed and analyzed. Based on the study of different algorithms, a classification of the scheduling algorithms on the basis of selected features has been presented.

**Keywords**-Cloud computing; Scheduling algorithms; Resource allocation; Open source; virtual machine

## I. INTRODUCTION

Cloud computing (CC) has arisen out of developments in grid computing, virtualization and web technologies. CC is a scalable distributed computing environment in which a large set of virtualized computing resources, different infrastructures, various development platforms and useful software's are delivered as a service to customers as a pay as you go manner usually over the Internet [1]. CC consists of infrastructure, platforms, and applications. Cloud technology helps business organizations, academic institutions, government organizations in cutting down operational expenses. The significant features of CC include lower cost, incremental scalability, reliability and fault-tolerance, service-oriented, utility-based, virtualization and SLA [2] [3]. Cloud computing have been used for various areas like education, business, health etc. It is required to have an efficient use of resources by different application areas [4] [5].

Cloud computing environments can be built on different system infrastructures like on physically located grids (grid-based), geographically distributed services (service-based), commercial cloud computing infrastructure (business-based) etc. Cloud enables on-demand provisioning of computational resources, in the form of virtual machines (VMs) deployed in a cloud provider's datacenter. Cloud computational resources are shared among different cloud consumers who pay for the services according to the usage of resource. The allocation of resource and proper scheduling has significant impact on the performance of the system. The primary aim of CC is to provide efficient

access to remote and geographically distributed resources. An efficient scheduling is needed to manage the access to the different resources. It also provides proper load balancing while resource allocation. There are different types of resource scheduling in CC that are based on different parameters like cost, performance, resource utilization, time, priority, and physical distances, throughput, bandwidth, and resource availability.

### A. Taxonomy of Cloud Computing

Cloud computing provides various services related to software, platform, infrastructure, data, identity and policy management. Cloud Computing building blocks consists of three layers IaaS, PaaS & SaaS as shown in Fig. 1. Software as a service Documents, applications, data, information Platform as a service Identity and access management, developing tools, data management Infrastructure as a service Virtualization, management, utility computing, connectivity Hardware and software environment

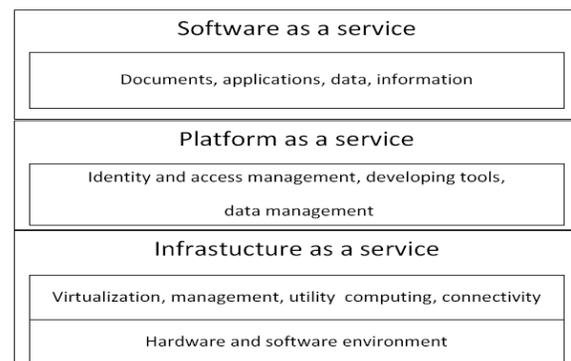


Figure 1. Cloud computing service layers

Corresponding Author : Rajesh Verma

IaaS delivers hardware and associated software as a service. It consists of servers (stand alone and virtualized), storage, network, operating systems, virtualization technology and file system. Virtualized infrastructure provides the abstraction necessary to ensure that an application or business service is not directly tied to the underlying hardware infrastructure. Platform as a service provides application development and deployment platform as a service to developers over the web. This platform consists of infrastructure software, and typically includes a database, middleware and development tools. Identity and access management is also implemented at this layer and offered to the user as a service. Software as a service layer provides the user custom build applications such as CRM, ERP. These are built and deployed on cloud by developers. Some SaaS providers run on another cloud provider's PaaS or IaaS service offerings. Some of SaaS applications are Oracle CRM on Demand, Salesforce.com, Net suite etc. Level of sharing IaaS PaaS SaaS IPaaS Public cloud Hosted virtual Private cloud Private Cloud` Pure Cloud Market Dynamic Infrastructure Services Integration as a service Dynamic Apps Service Dynamic BPO service Extended Cloud Market Infrastructure virtualization tools Middleware virtualization tools Apps virtualization tools BP virtualization tools Infrastructure Middleware Application Information and processes Business Values

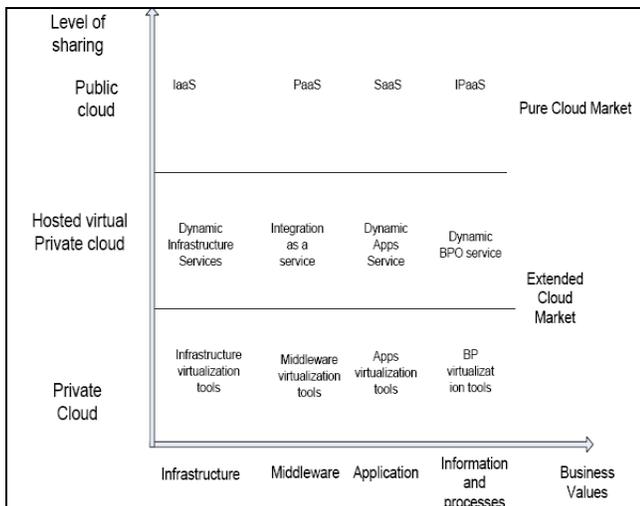


Figure 2. Types of cloud and services provided

**B. Types of Cloud Computing**

Cloud Computing environment can be classified in many ways. Based on outsourcing the services and deployment as shown in Fig.2 . Cloud computing environment can be classified as public cloud, private cloud and hybrid cloud. A public cloud sells services to anyone on the Internet. A private cloud is a proprietary network or a data center that supplies hosted services to a limited number of people. A hybrid cloud is a combination of public and private clouds. For example, a business may choose to provide some

resources internally while choosing to outsource some externally.

**C. Scheduling in Cloud Computing**

Scheduling algorithm is the method by which threads, processes or data flows are given access to system resources (e.g. VM, processor time, communications bandwidth). The process of scheduling is very significant in the CC environment for efficiently using the distributed resources. The speed, efficiency, utilization of resources in optimized way depends largely on the type of scheduling algorithm selected for the CC environment. This is usually done to load balance a system effectively or achieve a target quality of service. Scheduling is the process of deciding the distribution of the resources between varieties of possible tasks. There are certain factors that scheduler is mainly concerned. These include throughput, latency, turnaround, response time and fairness. Throughput is number of processes that complete their execution per time unit. Latency is a measure of time delay experienced in a system. Turnaround is total time between submission of a process and its completion. Response time can be defined amount of time it takes from when a request was submitted until the first response produced. Fairness is equal CPU time to each process generally according to each process' priority.

There are various issues pertaining to scheduling for various systems. The applications require different optimization criteria. In Batch systems criteria required are throughput and turnaround time. In Interactive system criteria required are response time, fairness, and user expectation However in Real-time systems meeting deadlines is important criteria. The scheduling algorithms shall be selected in a way it satisfies the required criteria for efficient resource allocation and better services.

A comprehensive study has been carried out to study and analyze the working of different scheduling algorithms. Based on this study different selected significant factors are being identified for comparative analysis of the algorithms.

**II. PREVIOUS WORK**

Resource allocation and scheduling of resources have been an important aspect that affect the performance of networking, parallel, distributed computing and cloud computing. Many researchers have proposed various algorithms for allocating, scheduling and scaling the resources efficiently in the cloud. These includes first come first serve, min-min max min, ant colony, round robin, earliest dead line first, hybrid heuristic, back tracking, task duplication, genetic algorithm, loss and gain simulated annealing, ant colony , greedy etc. Various modified scheduling algorithms like Improved Genetic Algorithm, Modified ant colony optimization, Extended Min-Min have

also been proposed by researchers [6-10][12][15][16][18][19][24].

Energy-efficient (Green) resource allocation policies and scheduling algorithms for cloud computing are proposed. The algorithm uses the prediction in making turning off/on the number of running servers/hosts in order to eliminate the idle power consumption [20-22].

Scheduling strategies in this cloud computing scenario should satisfy the objectives of customer as well as of service providers. Authors have addressed specific problem related to scheduling of consumers' service requests (or applications) on service instances made available by providers taking into account costs incurred by both consumers and providers as the most important factor [11][13][14][16][31].

There are two major types of workflow scheduling algorithms for grid and cloud workflow management systems. These are best-effort and QoS constraint based algorithms. Best-effort based scheduling attempts to minimize the execution time, ignoring other factors such as the monetary cost of accessing resources and various users QoS satisfaction levels. Users QoS (Quality of Service) requirements like deadline and budget have been addressed in these existing grid and cloud workflow management systems [30].

Based on the literature review, various CC Scheduling algorithms as mentioned above have been identified. Some scheduling algorithms have further been modified by the researchers. A detailed discussion of these algorithms is presented in section III.

### III. SCHEDULING ALGORITHMS

The efficiency of task scheduling has a direct impact on the performance of the entire cloud environment; many heuristic scheduling algorithms were used to optimize it. Scheduling in cloud computing environment is performed at various levels like workflow, VM level, task level etc. The scheduling algorithms are also divided according to scheduling policies i.e. preemptive or non-preemptive. In a non-preemptive (FCFS, SJF) pure multiprogramming system the short-term scheduler lets the current process run until the task is not finished. Preemptive policies (RR), on the other hand, force the currently active process to release the CPU on certain events or priority, such as a clock interrupt, some I/O interrupts or a system call. The classification presented in this research work is based on different parameters like time, cost, workflow, energy etc. The Performance of the scheduling algorithms can be estimated based on different parameters like deadline

constraint, CPU load, CPU power usage, remaining time of the task etc.

A detailed description of various algorithms is given below. Few algorithms (FCFS, round robin, min-min max min, ant colony, genetic etc.) are basic algorithms which are been extensively used for distributed systems. Recently some new modified algorithms have been proposed. These include extended min-min max min, improved genetic, modified ant colony etc.

#### *First-Come-First-Serve (FCFS) –*

FCFS for parallel processing and is aiming at the resource with the smallest waiting queue time and is selected for the incoming task. The Cloud Sim toolkit supports First Come First Serve (FCFS) scheduling strategy for internal scheduling of jobs. Allocation of application-specific VMs to Hosts in a Cloud-based data center is the responsibility of the virtual machine provisioner component. The default policy implemented by the VM provisioner is a straightforward policy that allocates a VM to the Host in First-Come-First-Serve (FCFS) basis [6][7]. The disadvantages of FCFS is that it is non-preemptive. The shortest tasks which are at the back of the queue have to wait for the long task at the front to finish. Its turnaround and response is quite low.

#### *Round robin-*

Round Robin (RR) algorithm focuses on the fairness. RR uses the ring as its queue to store jobs. Each job in a queue has the same execution time and it will be executed in turn. If a job can't be completed during its turn, it will be stored back to the queue waiting for the next turn. The advantage of RR algorithm is that each job will be executed in turn and they don't have to be waited for the previous one to get completed. But if the load is found to be heavy, RR will take a long time to complete all the jobs [6]. The CloudSim toolkit supports RR scheduling strategy for internal scheduling of jobs [9]. The drawback of RR is that the largest job takes enough time for completion.

#### *Earliest Deadline First –*

Earliest Deadline First (EDF) or Least Time to Go is a dynamic scheduling algorithm used in real-time operating systems. It places processes in a priority queue. Whenever a scheduling event occurs (task finishes, new task released, etc.) the queue will be searched for the process closest to its deadline, the found process will be the next to be scheduled for execution [8].

#### *Adaptive Scheduling Algorithm(ASA)-*

The Adaptive Scheduling Algorithm (ASA) is proposed to find a suitable execution sequence for workflow activities by additionally considering resource allocation constraints and dynamic topology changes. The idea of ASA is to divide the scheduling process into two phases: a logical

partition step and a physical allocation step. The key motivation behind this multi-stage procedure is to reduce the complexity of scheduling problem. The logical partition step identifies partitions of workflow activities based on similar attribute values and the physical allocation step assigns these partitions by using constraint programming. Meanwhile, network changes are considered in ASA to make it a robust and adaptive scheduling procedure [10].

#### *Back-tracking Algorithm –*

The back-tracking algorithm [12] assigns available tasks to the least expensive computing resources. An available task is an unmapped/unscheduled task whose parent tasks have been scheduled. If there is more than one available task, the algorithm assigns the task with the largest computational demand to the fastest resources in its available resource list. The heuristic repeats the procedure until all tasks have been mapped. After each iterative step, the execution time of the current assignment is computed. If the execution time exceeds the time constraint, the heuristic back tracks the previous step, removes the least expensive resource from its resource list and reassigns tasks with the reduced resource set. If the resource list is empty the heuristic keeps back tracking to the previous steps, reduces corresponding resource list and reassigns the tasks.

#### *Deadline Distribution Algorithm –*

The deadline distribution algorithm [13] tries to minimize the cost of execution while meeting the deadline for delivering results. This algorithm partitions a workflow and distributes the overall deadline into each task based on their workload and dependencies. It uses synchronization Task Scheduling (STS) for synchronization tasks and Branch Task Scheduling (BTS) for branch partition respectively. Once each task has its own sub-deadline, a local optimal schedule can be generated for each task. If each local schedule guarantees that their task execution can be completed within their sub-deadlines, the whole workflow execution will be completed within the overall deadline. Similarly, the result of the cost minimization solution for each task leads to an optimized cost solution for the entire workflow. Therefore, an optimized workflow schedule can be constructed from all local optimal schedules. The schedule allocates every workflow task to a selected service such that it can meet its assigned sub-deadline at a low execution cost.

#### *Profit-driven Service Request Scheduling –*

Authors have presented two sets of profit-driven service request scheduling algorithms. These algorithms are devised incorporating a pricing model using process sharing (PS) and two allowable delay metrics based on service and application. Authors have demonstrated the efficiency of consumer applications with interdependent services. The

evaluation of the algorithm was done on the basis of maximum profit and utilization [14].

#### *Greedy algorithm –*

Greedy algorithm works in phases. In each phase, a decision is made that appears to be good (local optimum), without regard for future consequences. When the algorithm terminates, the local optimum should be equal to the global optimum. Otherwise, a suboptimal solution is produced. It is used to generate approximate answers, rather than exact one which need more complicated algorithms. Greedy algorithm is used in open source Eucalyptus, nimbus for scheduling [15].

#### *Ant colony –*

In Ant colony algorithm, after initialization of the pheromone trails, ants construct feasible solutions, starting from random nodes, and then the pheromone trails are updated. At each step ants compute a set of feasible moves and selects the best one (according to some probabilistic rules) to carry out the rest of the tour. The transition probability is based on the heuristic information and pheromone trail level of the move. The higher value of the pheromone and the heuristic information, the more profitable it is to select this move and resume the search [16] [17].

#### *Genetic Algorithm-*

Genetic algorithms are stochastic search algorithms based on the mechanism of natural selection strategy. It starts with a set of initial solution, called initial population, and will generate new solution using genetic operators. The advantage of this technique is it can handle a large searching space, applicable to complex objective function and can avoid trapping by local optimum solution. Authors have developed a genetic algorithm, which provide a cost-based multi QoS scheduling in cloud [18][29].

#### *The match-making algorithm (MMA) –*

The MMA, first those hosts that do not meet the VM requirements and do not have enough resources to run the VM are filtered out. The "RANK" expression is evaluated upon this list using the information gathered by the monitor drivers. Any variable reported by the monitor driver can be included in the rank expression. Open Nebula comes with a match making scheduler that implement the Rank scheduling policy. The goal of this policy is to prioritize those resources more suitable for the VM. Those resources with a higher rank are used first to allocate VMs [27].

#### *Modified Best Fit Decreasing (MBFD) algorithm*

In modification (MBFD) all VMs are sorted in decreasing order of current utilization and each VM is allocated to a host that provides the least increase of power consumption due to this allocation. This allows leveraging heterogeneity

of the nodes by choosing the most power-efficient ones [21].

#### *Green Scheduling Algorithm –*

A Green Scheduling Algorithm which makes use of a neural network based predictor for energy savings in Cloud computing. The predictor is exploited to predict future load demand based on collected historical demand. The algorithm uses the prediction in making turning off/on decisions to minimize the number of running servers [22].

#### *EMM (Extended Min-Min) Scheduling Algorithm –*

Algorithm is designed for task scheduling with the main purpose to achieve maximum throughput in the group. This algorithm is executed periodically to provide dynamic scheduling in a batch mode. The EMM algorithm extends the original Min-Min algorithm and is more suitable for instance-intensive workflows. In workflow systems, resources are needed to execute tasks, and for every resource, it can only be occupied by a task at one time. The resources are allocated to tasks by assigning time slots to selected tasks. When it is decided to schedule a task (i.e. the task of the workflow instance) on resource, it will assign a suitable time slot of this resource to it. Of course, the data dependency and control dependency should be preserved simultaneously. The element in matrix means the estimated execution time for particular task on resource. If its value equals to -1, it means that task cannot be executed on resource. The original values of this matrix are estimated, but can modify the values at runtime according to the history execution record for better estimation later [24].

#### *Improved Genetic Algorithm (IGA)-*

An optimized scheduling algorithm to achieve the optimization or sub-optimization for cloud scheduling problems is proposed by researchers. They investigated the possibility to allocate the Virtual Machines (VMs) in a flexible way to permit the maximum usage of physical resources and used an IGA for the automated scheduling policy. The IGA uses the shortest genes (A gene is a molecular unit of heredity in a living organism) and introduces the idea of Dividend Policy in Economics (The policy a company uses to decide how much it will pay out to shareholders in dividend) to select an optimal or suboptimal allocation for the VMs requests. The problem of service request scheduling in cloud computing systems has been addressed. They considered a three-tier cloud structure, which consists of infrastructure vendors, service providers and consumers. The scheduling of consumers' service requests (or applications) on service instances is done taking costs incurred by both consumers and providers as the most important factor. It includes the development of a pricing model using processor-sharing (PS) for clouds application services. They developed two sets of profit-driven scheduling algorithms with key characteristics of

service requests including precedence constraints. The first set of algorithms explicitly takes into account not only the profit achievable from the current service, but also the profit from other services being processed on the same service instance. The second set of algorithms attempts to maximize service-instance utilization without incurring loss this implies the minimization of costs to rent resources from infrastructure vendors [25].

#### *Modified ant colony optimization algorithm –*

An initial heuristic algorithm to apply modified ant colony optimization approach for the diversified service allocation and scheduling mechanism in cloud paradigm is proposed. The optimization method is aimed to minimize the scheduling throughput to service all the diversified requests according to the different resource allocator available under cloud computing environment [26].

#### *Improved activity based cost algorithm –*

It makes efficient mapping of tasks to available resources in cloud. This scheduling algorithm measures both resource cost and computation performance, it also improves the computation/communication ratio by grouping the user tasks according to a particular cloud resource's processing capability and sends the grouped jobs to the resource [31]. The scheduling algorithms for execution of workflow in cloud computing environment are proposed [11], [28], [32]. These work flow algorithms are discussed below:-

#### *Particle Swarm Optimization algorithm–*

K. Liu et al. used the heuristic to minimize the total cost of execution of application workflows on Cloud computing environments. They calculated the total cost of execution by varying the communication cost between resources and the execution cost of compute resources [11].

#### *Compromised – time - cost scheduling algorithm –*

S Pandey et al. presented a compromised-time-cost scheduling algorithm in which they considered cost-constrained workflows by compromising execution time and cost with user input [28].

#### *Market Oriented Hierarchical Scheduling*

Zhangjun Wu et. al. proposed a market-oriented hierarchical scheduling strategy which consists of a service-level scheduling and a task-level scheduling. The service-level scheduling deals with the Task-to-Service assignment and the task-level scheduling deals with the optimization of the Task-to-VM assignment in local cloud data centers. It can be used to optimize the time and cost simultaneously [32].

## IV. RESULT AND DISCUSSIONS

There are various factors that determine which algorithm is best for Cloud environment for providing efficient services.

There are various algorithms used for scheduling cloud computing environment for resource allocation which are based on various factors like cost, throughput, time etc. Various Cloud providers offer paid extra resources to users in an on demand manner. Therefore, scheduling policies should consider resources' prices, deadline and time constraints.

Some of the scheduling algorithms are based on cost factor. These include deadline distribution algorithm, backtracking, and improved activity based cost algorithm, compromised-time-cost etc. The algorithms like Extended Min-Min, modified ant colony optimization are based on throughput factor. Earliest deadline, FCFS, Round robin is time based. Time Optimization scheduling policy minimizes the application completion time where as Cost Optimization scheduling policy minimizes the cost incurred for running the application.

Different scheduling algorithms have been used by different providers for scheduling at different levels in cloud computing environment for generating better results and optimized resource utilization. The major cloud providers have been listed in Table 1.along with the kind of algorithms used by them for resource allocation and load balancing. Some of the open source providers are Eucalyptus, Open nebula etc.

On the basis of above study various selected factors have been identified to classify the algorithms. Scheduling algorithms can be classified according to following criteria like time, cost, energy etc as shown in table 2. Some of the scheduling algorithm can optimize the time span where as other can minimize the cost or energy consumption. So based on the customer or service provider requirements various algorithms can be used for enhancing the efficiency and also to get optimized resource allocation and load balancing depending on various factors.

TABLE I. SCHEDULING ALGORITHMS USED IN DIFFERENT CLOUD ENVIRONMENT

Cloud provider	Open source	Algorithms used for scheduling
Eucalyptus	Yes	Greedy first fit and Round robin
Open Nebula	Yes	Rank matchmaker scheduling, preemption scheduling
Rackspace	yes	round robin, weighted round robin, least connections, weighted least connections
Nimbus	yes	Virtual machine schedulers PBS and SGE
Amazon EC2	No	Xen ,swam, genetic
Red Hat	yes	Breath first ,Depth first
Luna cloud	yes	Round robin

TABLE 2 FACTOR BASED CLASSIFICATION OF SCHEDULING ALGORITHMS

Time	Cost	Energy/ Others ( Queue, rank etc)
Round robin	Dead line distribution	Modified Best fit Decreasing
Earliest Dead Line first	Compromised –time cost	Green Scheduling Algorithm
Back tracking	Genetic Algorithm	FCFS
Modified ant colony optimization	Improved activity based cost	Adaptive Scheduling Algorithm
Extended Min-Min	A Particle Swarm Optimization	-
Market Oriented Hierarchical Scheduling	Improved Genetic Algorithm	-
Compromised time cost	Profit-driven service oriented algorithm	-

V. CONCLUSION

In Cloud computing environment heterogeneous resources are usually provided as services in forms of virtual machines. To manage heterogeneous resources in optimized way efficient scheduling and load balancing is required. In this study, an effort has been made to study various scheduling algorithms in cloud computing environment. To solve the resource scheduling problem various scheduling algorithms based on various factors have been tried by various researchers. After a comprehensive study, a classification of the scheduling algorithms on the basis of selected factors like cost, time, energy etc. has been presented in this paper. This classification shall help in selecting the appropriate class of scheduling algorithms for different types of services as per the requirements of the consumers and service providers. This analysis may further be used for optimization of different algorithms for better resource management in cloud computing environment.

VI. FUTURE WORK

Future work will include implementation of various kinds of algorithms using cloud simulator and open source cloud environment. Factors based comparison of various algorithms on the basis of results obtained shall be done in further research.

REFERENCES

[1] I. Foster, Y. Zhao, I. Raicu and S. Lu, "Cloud Computing and Grid Computing 360 Degree Compared", Grid Computing Environments Workshop, Austin, 2008.  
 [2] G. Boss, P. Malladi, D. Quan, L. Legregni and H. Hall, Cloud Computing (White Paper), IBM, october2007, [http://download.boulder.ibm.com/ibmdl/pub/software/dw/wes/hipods/Cloud\\_computing\\_wp\\_final\\_8Oct.pdf](http://download.boulder.ibm.com/ibmdl/pub/software/dw/wes/hipods/Cloud_computing_wp_final_8Oct.pdf) , accessed on May. 19, 2009.

- [3] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-Oriented Cloud and Atmospheric Computing: Hype, Reality, and Vision", Proc. of 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08), 5-13, Dalian, China, September, 2008.
- [4] I. Gandotra, P. Abrol, "Cloud computing :A new paradigm for Education", ICAET10, 2010, pp 92.
- [5] I. Gandotra, P. Abrol, "cloud computing a new Era of Ecommerce", book chapter in Strategic Service management, Published by Excel Books, pp 228, 2010
- [6] S. Sadhasivam, N. Nagaveni, R. Jayarani, and R. Vasanth Ram, "Design and Implementation of an efficient Two-level Scheduler for Cloud Computing Environment", International Conference on Advances in Recent Technologies in Communication and Computing, 2009.
- [7] Rodrigo N. Calheiros, Rajiv Ranjan, César A. F. De Rose, and Rajkumar Buyya, "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services".
- [8] J. Singh, B. Patra, S. P. Singh, "An Algorithm to Reduce the Time Complexity of Earliest Deadline First Scheduling Algorithm in Real-Time System", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No.2, February 2011
- [9] Shin-ichi Kuribayashi, "Optimal Joint Multiple Resource Allocation Method for Cloud Computing Environments", International Journal of Research and Reviews in Computer Science (IJRRCS) Vol. 2, No. 1, March 2011.
- [10] A. Avanes and J. Freytag, "Adaptive Workflow Scheduling under Resource Allocation Constraints and Network Dynamics". Proc. VLDB Endow, 1(2), August 2008, pp 1631-1637.
- [11] K. Liu, Y. Yang, J. Chen, X. Liu, D. Yuan, H. Jin, "A Compromised-Time-Cost Scheduling Algorithm in SwinDeW-C for Instance-intensive Cost-Constrained Workflows on Cloud Computing Platform", Int. Journal of High Performance Computing Applications, Volume 24 Issue 4, 2010.
- [12] D. A. Menasc and E. Casalicchio, "A Framework for Resource Allocation in Grid Computing", Proc. of the 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, Volendam, The Netherlands, October, 2004, pp 259-267.
- [13] J. Yu, R. Buyya and C. K. Tham, "A Cost-based Scheduling of Scientific Workflow Applications on Utility Grids", Proc. of the 1st IEEE International Conference on e-Science and Grid Computing, Melbourne, Australia, December 2005, pp 140-147.
- [14] Young Choon Lee, Chen Wang, Albert Y. Zomaya, Bing Bing Zhou, "Profit-driven Service Request Scheduling in Clouds", 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010
- [15] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster, "An Open Source Solution for Virtual Infrastructure Management in Private and Hybrid Clouds", IEEE INTERNET COMPUTING, SPECIAL ISSUE ON CLOUD COMPUTING, July 2009
- [16] Nada M. A. Al Salami, "Ant Colony Optimization Algorithm", UbiCC Journal, Volume 4, Number 3, August 2009
- [17] Sachin V. Solanki, B. Minal Gour and C. Dr. (Mrs.) A. R. Mahajan, "An Overview of Different Job Scheduling Heuristics Strategies for Cloud Computing Environment", www.icett.com, 2011
- [18] J. Yu and R. Buyya, "Scheduling Scientific Workflow Applications with Deadline and Budget Constraints using Genetic Algorithms", Scientific Programming Journal, 14(3-4), 217-230, IOS Press, 2006.
- [19] R. Sakellariou, H. Zhao, E. Tsiakkouri, and M. D. Dikaiakos, "Scheduling Workflows with Budget Constraints, CoreGRID Workshop on Integrated research in Grid Computing, Technical Report TR-05-22, University of Pisa, Dipartimento Di Informatica, Pisa, Italy, November 2005.
- [20] Bo Li, J. Li, J. Huai, T. Wo, Q. Li, L. Zhong, "Ena, Cloud: An Energy-saving Application Live Placement Approach for Cloud Computing Environments", in Proc. IEEE Int. Conf. Cloud Computing, IEEE, 2009, pp 17-24.
- [21] R. Buyya, A. Beloglazov, J. Abawajy, "Energy-Efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges", in Proc. of the Int. Conf. on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010), Las Vegas, USA, July 12-15, 2010.
- [22] T. V. T. Duy, Y. Sato, Y. Inoguchi, "Performance Evaluation of a Green Scheduling Algorithm for Energy Savings in Cloud Computing", IEEE, 2010.
- [23] [http://www.rackspace.com/cloud/cloud\\_hosting\\_products/loadbalancers/technology/](http://www.rackspace.com/cloud/cloud_hosting_products/loadbalancers/technology/)
- [24] Ke Liu, Jinjun Chen, Yun Yang and Hai Jin, "A throughput maximization strategy for scheduling transaction-intensive workflows on SwinDeW-G", Concurrency Computat.: Pract. Exper. 2008; 20:1807-1820. Published online 10 July 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cpe.1316
- [25] H. Zhong, K. Tao, X. Zhang, "An Approach to Optimized Resource Scheduling Algorithm for Open-source Cloud Systems", in Proc. chinagrid2010, pp. 124-129 IEEE, 2010.
- [26] S. Banerjee, I. Mukherjee, and P. K. Mahanti, "Cloud Computing Initiative using Modified Ant Colony Framework", World Academy of Science, Engineering and Technology, 56 2009, pp 221-224
- [27] S. Pandey, Linlin Wu, S. M. Guru, R. Buyya, "A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments", in Proc. 24th IEEE Int. Conf. on Advanced Information Networking and App. IEEE, 2010, pp. 400-407.
- [28] <http://opennebula.org/documentation:archives:rel2.0:schg>
- [29] D. Dutta, R. C. Joshi, "A Genetic Algorithm Approach to Cost-Based Multi-QoS Job Scheduling in Cloud Computing Environment", International Conference and Workshop on Emerging Trends in Technology (ICWET 2011) - TCET, Mumbai, India, 2011
- [30] J. Yu and R. Buyya, "Workflow Scheduling Algorithms for Grid Computing, Technical Report", GRIDS-TR-2007-10, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, May 2007.
- [31] Selvarani, S.; Sadhasivam, G. S., "Improved cost-based algorithm for task scheduling in cloud computing", Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International conference on 2010, pp 1-5
- [32] Meng Xu, Lizhen Cui, Haiyang Wang, Yanbing Bi, "A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing", in 2009 IEEE International Symposium on Parallel and Distributed Processing.