IJCSE International Journal of Computer Sciences and Engineering Open Access

Survey Paper Volume-5, Issue-10 E-ISSN: 2347-2693

Cloud Scheduling using Meta Heuristic Algorithms

A. Jain[1*], A. Upadhyay[2]

[1]*Dept. of Computer Science and Engineering, IES-IPS Academy, RGPV, Indore, India
[2] Dept. of Computer Science and Engineering, IES-IPS Academy, RGPV, Indore, India

*Corresponding Author: jainakhil615@gmail.com, Tel.: +91-89895-02226

Available online at: www.ijcseonline.org

Received: 25/Sep/2017, Revised: 08/Oct/2017, Accepted: 19/Oct/2017, Published: 30/Oct/2017

Abstract— Cloud computing has transformed into a well-known in area of high performance, cloud computing as it offers on-request access to shared pool of resources over web in a self-service, dynamically scalable. One of the important research issues which need to be focused for its efficient performance on task scheduling which plays the key role for increase the efficiency of whole cloud computing facilities. implies that to assign best suitable resources for the requested task to be execute with the various parameters like time, cost, scalability, makespan, reliability, resource utilization, accessibility, throughput etc. In this paper, we give survey and relative studies of a few task scheduling using metaheuristic algorithms for cloud computing.

Keywords: Cloud Computing, Task Scheduling, Meta-heuristic, hyper heuristic, PSO, GA, ACO.

I. Introduction

The cloud computing system is the propel advancement of distributed computing, parallel processing and grid computing, and is web based computing. Cloud computing is a new type of shared infrastructure which can connect huge pools of systems, provides users with a variety of storage and computing resources via the internet on the basis of Pay-As-Per-Use system[2]. Task scheduling plays the main concern for the execution of performance driven applications.

Cloud computing is creating a boom in present scenario and become a basic service like electrical energy, water phone that a common man is dependent on. This area is gaining popularity and becomes a boom presently due to its wide applicability like client server, browser dependent programming and distribution of resources to achieve reliability and scalability.

Resources of cloud are used by client-server based system over digital networks or over the public internet itself. Five specific qualities that are define cloud computing [2]:

- on-demand self-service
- wide-ranging of network access
- resource pooling
- rapid elasticity
- measured service

That's cloud computing. However, we examine the design that are commonly divided into three service models Cloud Software as a Service (SaaS), Cloud Platform as a Service (PaaS), Cloud Infrastructure as a Service (IaaS) and, four deployment models Private cloud, Community cloud, Public cloud, Hybrid cloud that exhibited by a given cloud structure[3].

1.1. Scheduling On Cloud

Find an optimal solution to schedule a given set of tasks *T = {T1, T2, T3….,Tn}* to a given set of machines *M = {M1, M2, M3, ……., Mm}* subject to a predefined set of measurements. For an example, one of the mostly used estimations for the scheduling issue is the makespan *Cmax*(S). Mathematically, the scheduling problem is to. The scheduling problem on cloud can be formulated for the solution *S* as follows:[33]

$$minimize\ f(s) = C_{max}(S) + \sum_{i=1}^{n}\sum_{j=1}^{m} C_{i,j}$$
$$subject\ to\ \ C_{max}(S) \leq U(S),$$
$$C(S) \leq B(S),$$

Where: f(S) Objective function, Cmax(S) Completion time of the last task (also called makespan), n Number of tasks, m Number of machines, $C_{i,j}$ Cost of processing the ith task on the jth machine, U(s) Number of overdue tasks, B(s) Restriction on the budget for the tasks of s.

$$minimize\ f(s) = C_{max}(S),$$

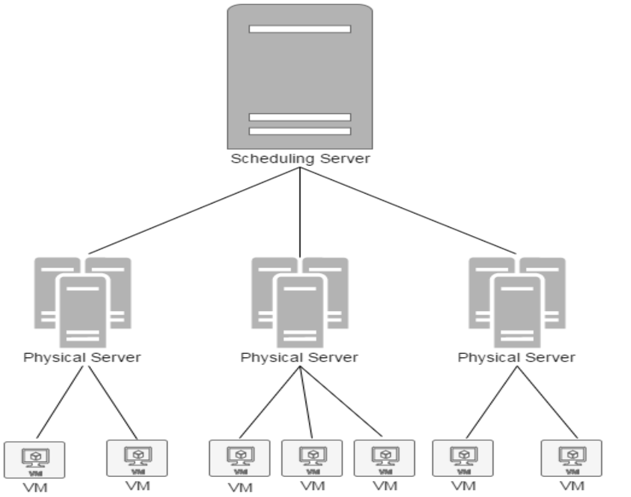

Figure 1.1 Scheduling Architecture in Cloud Computing Environment [7]

Optimization criteria [6]

i. **Makespan:** It is the completing time of the last task. The most common improvement criterions while scheduling task is minimization of makespan

$$Makespan = max_{i\ \in task}\{F_i\}$$

Where f_i denotes the finishing time of task *i*.

ii. **Flow Time:** It is the sum of completing time of all the tasks. For reducing the flowtime, scheduling tasks should be executed in ascending order of their processing time.

$$Flow\ time = \sum_{i\ \in tasks} F_i$$

Where F_i represents the completing time of task *i*

iii. **Waiting time:** Waiting time is the difference between the task execution start time and submission time of the task.

$$Wating\ time_i = S_i + B_i$$

Where S_i is start time and B_i is submission time of task *i*.

iv. **Turnaround time:** This monitors to how long it takes for a task to finish execution since its submission. It is the total of waiting time and execution time of task.

$$Turnaround\ time_i e = W_i + E_i$$

Where W_i is waiting time and E_i is execution time of task *i*

v. **Economic cost:** It indicates the total amount the user needs to pay to service provider for resource utilization.

$$Economic\ cost = \sum_{i\ \in resourc} \{C_i * T_i\}$$

Where C_i denotes the cost of resource *i* per unit time and T_i denotes the time for which resource *i* is utilized.

1.2. Scheduling Methods For Cloud Computing

The majority of the task scheduling issues on cloud computing environment are still either NP-hard or NP-finish [13]. As indicated by our perceptions, most traditional scheduling algorithms (e.g., exhaustive and deterministic algorithms) are widely utilized on the present cloud computing environment because they are basic and simple to execute. Unfortunately, these scheduling algorithms are unsuitable for large scale or complex scheduling issues in light of the fact that the results of these scheduling methodologies are for usually far from optimal.

A promising research direction, which applies modern meta-heuristics algorithms, such as simulated annealing (SA) [33], genetic algorithm (GA) [8], particle swarm optimization (PSO) [19], bat Algorithm [23], ant colony optimization (ACO) [14], and cuckoo search algorithm [27] to scheduling on cloud computing systems, has attracted several researchers from different research domains. The extended adaptation of heuristics joins more than one heuristics into a single algorithm to use their qualities for scheduling, this type of arrangement called hybrid heuristic scheduling algorithm [29].

In cloud environments two players one is cloud service providers and another cloud users. Providers hold huge computing resources in their large datacenters and provide resources on rent to users on a pay-per-usage basis. Users who have an applications with fluctuating loads and rent resources from providers to run their applications. In maximum cases, the connection among providers and users happen as appeared in figure 1.2

Rest of the paper is organized in the following manner, Section I contains all the introductory requirements to understand the domain area. It also provides the detailed explanation about cloud computing, its types, cloud services, cloud scheduling, cloud optimization criteria on cloud computing, Section II presents a survey on scheduling algorithms available with this domains such as GA, ACO, PSO, BAT. In this work a wide variety of existing mechanism, algorithms and architectures is studied, Section III we conclude the survey work done by us.

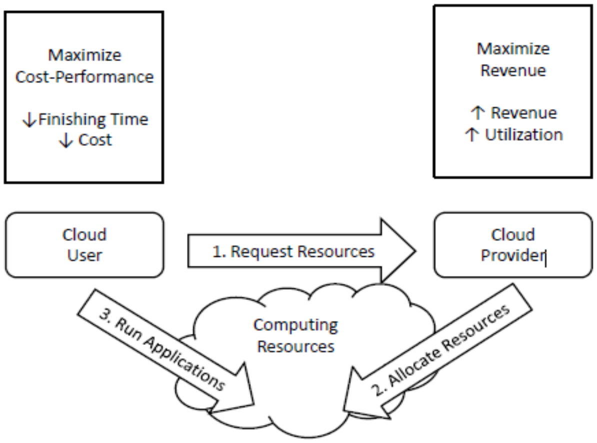

Figure 1.2 Cloud Usage Scenario

II. LITERATURE SURVEY

In the literature we study about various task scheduling algorithm in cloud computing, so many task scheduling algorithm were proposed in the past.
We study this algorithms in four category:

1. Traditional (Heuristic)
2. Meta-Heuristic
3. Hybrid Heuristic
4. Hyper Heuristic

A. *Traditional Algorithms:*

H. Chen. et al. [4] have studied the comparative performance regarding then that of traditional (heuristic) algorithms for scheduling such as FCFS, Opportunistic Load Balancing (OLB), Minimum Execution Time (MET), Minimum Completion Time (MCT), Min-Min, Max-Min, etc. They had also introduced the concept of Min-Min algorithm in order to overcome the makespan and rise the resource utilization. Their results had shown that the simple Min-Min algorithm gave a much better schedule which minimizes the makespan than the other algorithms.

Heuristics algorithms is mostly used to search an optimal solution for the scheduling. Heuristic algorithms are used in both the cases of static and dynamic. Static heuristic is utilized when the number of tasks to be finished are known in earlier. Dynamic heuristic can be used when the task arrival is dynamic in nature. Min-min scheduling worked on Minimum Completion Time (MCT) that is used to assign tasks to the resources having minimum expected completion time.

S. Devipriya et al. [5] proposed a simple changes of Max-min algorithm is based on expected execution time as an alternative of complete time. Thus the tasks within cloud using Improved Max-min can attain lower makespan rather than original Max-min.

B. *Meta-Heuristic Algorithms:*

According to **M. Kalra** and **S. Sing [6]** Meta-heuristic based methods have been showed to achieve close optimal solutions within reasonable time as compared to traditional algorithms. They gave a broad review and relative analysis of different scheduling algorithms for cloud environments in view of three prominent meta-heuristic techniques that are Ant Colony Optimization (ACO), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), and two new methods: League Championship Algorithm (LCA) and BAT algorithm.

In general, finding best schedule for cloud using the traditional sequential method is an NP-hard problem whereas heuristic methods will provided near best result for difficult problems. But the Ant Colony Optimization (ACO) under meta-heuristic is inspired by the behavior of actual ants to finding the shortest path among their colonies and a source of food. This novel method was presented by **M. Dorigo,** and **T. Stützle [14]** in 1992 and called Ant system. ACO approach are useful for solving discrete optimization problems. It has been effectively applied for resolving task scheduling in grid and cloud and many more environment.

K. Zhu et al. [9] focused on multi-agent genetic algorithm (MAGA) to achieve the best load balancing between VMs. MAGA is a hybrid algorithm which has combining GA with multi-agent techniques, MAGA is a kind of enhanced hybrid GA, significantly enhanced convergence time and optimization results as compared to that of traditional GA. MAGA has obvious superiority, mainly when handling very large-scale, high dimensional, complex, and dynamic optimization problems with easy. MAGA balanced both CPU consumption and memory usage among VMs. After then **Z. Zheng et al. [10]** offered Parallel Genetic Algorithm (PGA). The PGA performs 1.5 times faster than original GA when two threads are used and 2.7 times faster when four threads are used. The resource utilization rate is much better than first fit and Round Robin algorithm. It can easily employed on assorted computer networks or on parallel mainframes.

K. Dasgupta et al. [11] focused on resource load balancing while task scheduling on cloud using GA. Experimental outcomes shows that they reduce 25–26% response time

Table 1: Comparison of various Meta-heuristic task scheduling algorithms

Author	Strategy	Performance Criteria	Tasks Nature	Environment
J. Gu. et al. [8]	GA with Random initial population and selection operator but Crossover and Mutation operator is based on fitness ratio	Makespan	Independent	Cloud Computing (Cloud simulation using cloudsim tool)
Z. Zheng et. al. [10]	PGA with Based on best-fit and round-robin method for initial population, Random Gene Selection Crossover operator	Makespan, Load Balancing on Resources	Workflow	Cloud Computing (Cloud simulation using cloudsim tool)
K. Dasgupta et al. [11]	GA with Random initial population and selection operator based on fitness value, Fuzzy based crossover Operator	Makespan, Cost, Load Balancing	Independent	Cloud Computing(using Cloudsim in MATLAB)
M. Shojafar [12]	Permutation Based Representation of GA with random initial population and RouletteWheel Selection Operator	Makespan, Load Balancing, Resource Utilization,	Workflow	Grid Simulation Environment (Gridsim toolkit)
K. Kaur et al [13]	Permutation Based Representation GA Using LJFP and SJFP for initial population, Two - Point Crossover operator and Swap for mutation operator	Makespan and Execution Cost	Independent	Cloud Simulation Environment (using Cloudsim in java)
M. A. Tawfeek et al. [15]	ACO	Makespan	Independent	Cloud Computing (Cloud simulation using cloudsim tool)
P. Mathiyalagan et al.[16]	Changes in pheromone updating in ACO	Makespan	Independent	Grid Simulation Environment (GridSim toolkit)
Wen et al. [28]	Combined ACO with PSO algorithm	Makespan, Resource Utilization	Workflows	Cloud Computing (Cloud simulation using cloudsim tool)
H. Liu et al. [21]	1 * n Vector Representation with Random initial population	Makespan Average Resource Utilization	Independent	Grid Simulation Environment (Gridsim toolkit)
L. Zhang et al. [22]	Matrix Representation Using SJFP for initial population	Makespan	Independent	Cloud Computing (Cloud simulation using cloudsim tool)
X. S. Yang [24]	Bat Algorithm	Makespan	Independent	Cloudsim
L. Jacob [25]	Bat Algorithm	Optimal Resources, Minimal workflow cost	Workflow	Cloud Computing (Cloud simulation using cloudsim tool)

and FCFS algorithms with one datacenter and 75 VMs. The when compared with SHC (Stochastic Hill Climbing), RR suggested algorithm is performed on various other scenarios and their results prove that it better than that of FCFS, RR and SHC.

M. A. Tawfeek et al. [15] focused on the objective function for minimization of makespan. They have taken a constraint of visiting each VM once for each ant and heuristic function is based on expected execution time and transfer time of task. The proposed idea is implemented on Cloudsim simulator with the number of tasks varying from 100-1000. ACO is compared with RR and FCFS algorithms and simulation results demonstrate that the number of task increases, ACO takes less time as compared to RR and FCFS. Result shows that the around 29-32% reduction in makespan for 1000 tasks. **P. Mathiyalagan et al. [16]** offered ant colony optimization algorithm with some modification by updating the pheromone rule which resolves the cloud task scheduling problem efficiently as compared of the existing ACO. An effective task scheduling method requires not only meeting the user's needs but also improving the efficiency of the entire system. The ACO has been used for all types of scheduling problems, achieving promising results.

W. N. Chen et al. [17] offered a workflow scheduling algorithm based on Ant Colony System (ACS) algorithm with some novel features for its improvement. They designed to reduce the cost while meeting the time limit. For resolving this, they implemented two types of pheromone are well-defined, one for minimization of makespan and other for minimization of cost. They have characterize three sorts of heuristic information to control the ants to finding their search directions. In every cycle, every ant used one heuristics type and one pheromone type in view of the probabilities controlled by adaptively and balanced parameters in the algorithm. **J. Bagherzadeh, M. MadadyarAdeh [18]** have presented the idea of one-sided beginning ants to enhance ACO. Their approach uses the results of deterministic algorithms for one-sided initial ants. Creators have likewise viewed as standard deviation of task in addition to pheromone, heuristic data and expected that time would execute a task on a given machine. Their test results show makespan lessening of 33% and 20% in examination with MaxMin and MinMin individually in consistent, low task and low machine heterogeneity condition. **K. Li et al. [19]** presented Load Balancing Ant Colony Optimization (LBACO) algorithm, for autonomous (independent) tasks scheduling with the purpose of reducing makespan and even load across all VMs. They have also intended the degree of imbalance to measure the imbalance among VMs. Their test results demonstrate that the LBACO has reduced the average makespan by 63% and level of imbalance by 47% approximately in correlation with FCFS algorithm for the 500 independent tasks with the same parameter settings in CloudSim. In this manner it has performed much better to FCFS algorithm

In the category of meta-heuristic one of the most famous algorithm Particle Swarm Optimization (PSO) is an evolutionary computational method presented by **J. Kennedy** and **R. Eberhart [20]** in year of 1995 inspired by social behavior of the particles. Each particle is associated with position and velocity and moves through a multi-dimensional search space direction. In every iteration, every particle changes its velocity based of its best position and the position of the best particle of the entire population. PSO consolidates local search strategies with global search techniques and attempting to adjust exploration and exploitation. **H. Liu et al. [21]** used fuzzy matrices to represent position and velocities of particles. The component in each matrix indicates fuzzy relation among resource and job i.e. the degree of membership that the resource would execute the job in the feasible schedule solution space. **L. Zhang et al. [22]** proposed to apply Variable Neighborhood Search (VNS), a local search algorithm, after every emphasis of PSO to improve the exploitation of searching space.

Getting motivation from echolocation behavior of bats, **X. S. Yang [24]** introduced a novel optimization algorithm, BAT algorithm in 2010. Bats are uses echolocation to guess the distance of their prey. They fly randomly with a velocity, position, frequency, loudness and pulse emission rate to seek for their prey. When they are hunting for their prey, they can modify their frequency, loudness and pulse rate of emission based on the distance amid them and the prey. This behavior of bats has been used to formulate in BAT algorithm. **L. Jacob [25]** used BAT algorithm for scheduling on cloud pointing to reduce makespan and concluded that it has high accuracy, efficiency and convergence rate when compared to other algorithms like GA and PSO. **S. Raghavan et al. [26]** have also used Bat algorithm for solving workflow scheduling problem on cloud targeting to decrease processing cost of the over-all workflow. The result show better in terms of processing cost when compared with Best Resource selection (BRS) algorithm.

The newest improvement in the field of bio-inspired algorithms the Cuckoo Search is used for optimization problem, it has been seen that the performance of the cuckoo search is more than other meta-heuristic algorithm. The Cuckoo Search is created from the lifestyle of Cuckoo bird. Cuckoo birds leave their eggs in the nests of other birds with astounding capacities. **S. Joshi** and **S. Kour [27]** focused on cuckoo search algorithm which comes under the evolutionary algorithms for VM consolidation problem which is similar to bin packing problem and belongs to the NP hard problems class. They focused on minimum power consumption and resource wastage on cloud datacenters.

C. Hybrid Meta-heuristic Algorithms:

M. Kalra and **S. Singh [6]** conclude meta-heuristic techniques are generally slower than traditional algorithms and the generated solutions may not be optimal, thus most of the research done is toward improving the convergence speed and quality of the solution. These issues have been started by modifying the transition operator, pre-processing the input population or by taking hybrid approaches in meta-heuristic techniques. Hybrid algorithm means combines two or more heuristics or meta-heuristic into a single algorithm to leverage their strengths for scheduling, called hybrid scheduling algorithms. **M. Shojafar et al. [12]** presented a grouping of GA and fuzzy idea called FUGE that expects to reduce makespan, cost and level of imbalance in cloud while task scheduling. Fuzzy hypothesis is used to analyze the fitness estimation of chromosomes and for crossover operation. It is additionally contrasted and GA as far as makespan and execution cost. It gives an improvement of around 45% regarding execution cost and around half as far as makespan over GA. **Wen et al. [28]** proposed that ACO algorithm combined with PSO to increase performance. The proposed idea not only increases the convergence speed, it also increase resource utilization ratio, but also stays away from falling into local optimum solution. A hybrid of PSO and Multi-Objective Bat Algorithm is discussed by **Dr. S. George [29]** for profit intensification in cloud. PSO is used for local search and BAT methodology for global update, M/M/m queuing model is used to achieve multi-server structure and resources are allocated considering service charge and business cost to exploit profit.

D. Selvaraj and **G. S. Sadasivam [30]** developed Hybrid approach with grouping of PSO and GA, and called HPSO-GA for solving the task scheduling. The novel HPSO-GA implements GA processes such as crossover and mutation in PSO to increase efficiency of resource utilization and finish tasks within time limit. **R. Raju [31]** proposed Hybrid algorithm which combined the advantages of ACO and Cuckoo search algorithm and focused on decreasing the makespan. Obtained results shows that proposed algorithm performs better than the traditional ACO algorithm in terms of performance of the algorithm and makespan.

D. Hyper Heuristic Algorithm:

Hyper heuristic have ability to find the right method or sequence of heuristic in a given situation rather than trying to solve the problem directly.

According to **M. Gendreau [32]** hyper-heuristic are categorized into two types with learning and without learning. Hyper-heuristics without learning include approaches that use a few heuristics, but select the heuristics to call as per a predetermined arrangement. The hyper-heuristics with learning include techniques that dynamically change the preference of every heuristic based on their historical performance, guided by some learning component.

According to **C.W. Tsai et al [33]** hybrid-heuristic algorithms may have a higher chance to find a better result than a single heuristic does, it generally takes a longer computation time than heuristics at each iteration of the convergence process. They proposed another way to combine two or more heuristic, meta-heuristic algorithms that is a novel heuristic scheduling algorithm, called hyper-heuristic scheduling algorithm (HHSA), A hyper-heuristic is a high-level approach that tries to computerize the suitable mix of low level heuristics to effectively solve the specific problem. The fundamental thought of hyper-heuristic is to use "one and only one" heuristic algorithm at every emphasis of process.

HHSA outperforms many other scheduling methods, namely, Min-min, Max-min. FIFO, SA, GA, PSO, and ACO [33].

III. CONCLUSION

As the cloud computing changing day by day a lot of new difficulties are rising. One of them is the task scheduling for a cloud computing condition. In this paper, we have studied on the different existing resource allocation and job scheduling algorithms which can fundamentally enhance the effectiveness of the uses of resources as per certain situation in cloud environment and also tabulated their parameters such as Performance, Nature of Tasks and Environment. The fundamental goal of the scheduling is to expand usage of resources and to decrease makespan.

A lot of algorithms are proposed to achieve effective scheduling, but since the task scheduling is heuristic problem the more research can be done in this field and more optimized solutions can be achieved.

REFERENCES

[1] R. Buyya, C. S. Yeoa, S. Venugopal, J. Broberg, I. Brandic, *"Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility*", Future Generation Computer Systems the International Journal of eScience, 2009.

[2] S. Kumar, R. H. Goudar, "*Cloud Computing – Research Issues, Challenges, Architecture, Platforms and Applications: A Survey",* International Journal of Future Computer and Communication, Vol. 1, No. 4, December 2012.

[3] R. Nallakumar, N. Sengottaiyan, S. Nithya "*A Survey of Task Scheduling Methods in Cloud Computing*" International

Journal of Computer Sciences and Engineering (IJCSE), Vol. 02, No. 10. Oct 2014.

[4] H. Chen. Professor Frank Wang, Dr N. Helian, G. Akanmu, "*User-Priority Guided Min-Min Scheduling Algorithm For Load Balancing in Cloud Computing",* Parallel Computing Technologies (PARCOMPTECH), National Conference, Feb 2013.

[5] S. Devipriya, C. Ramesh, "*Improved Max-Min Heuristic Model For Task Scheduling In Cloud*", in Green Computing, Communication and Conservation of Energy (ICGCE), International Conference, Dec. 2013.

[6] M. Kalra, S. Singh, "*A review of metaheuristic scheduling techniques in cloud computing*" Egyptian Informatics Journal, Vol. 16, Issue 3, pp. 275–295 Nov 2015.

[7] F. Pop, C. Dobre, V. Cristea "*Genetic algorithm for DAG scheduling in grid environments*" Intelligent Computer Communication and Processing, IEEE 5th International Conference, Aug 2009.

[8] J. Gu J. Hu, T. Zhao, G. Sun, "*A New Resource Scheduling Strategy Based on Genetic Algorithm in Cloud Computing Environment"*, Journal of Computers, Vol. 7, No.1, Jan 2012.

[9] K. Zhu, H. Song, L. Liu, J. Gao, G. Cheng *"Hybrid Genetic Algorithm for Cloud Computing Applications*", in Services Computing Conference (APSCC), IEEE Asia-Pacific, Dec. 2011.

[10] Z. Zheng ,R. Wang, H. Zhong, X. Zhang, "*An Approach for Cloud Resource Scheduling Based on Parallel Genetic Algorithm"*, in Computer Research and Development (ICCRD), 3rd International Conference, Mar 2011.

[11] K. Dasguptaa, B. Mandalb, P. Duttac, J. K. Mondald, S. Dame, "*A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing*", in International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013.

[12] M. Shojafar, S. Javanmardi, S. Abolfazli, N. Cordeschi, "*A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method*", Cluster Computing, Vol. 18, Issue 2, pp 829–844, June 2015.

[13] K. Kaur , A. Chhabra , G Singh, "*Heuristics based genetic algorithm for scheduling static tasks in homogeneous parallel system".* International Research Journal of Computer Science (IRJCS), vol. 2, issue 9, pp. 14-19, Sep 2015.

[14] M. Dorigo and T.Stützle, *Ant Colony Optimization*, The MIT Press Cambridge, 2004.

[15] M. A. Tawfeek, A. E.Sisi, A. E. keshk, F. A. Torkey, "*Cloud task scheduling based on ant colony optimization*", Computer Engineering & Systems (ICCES), 8th International Conference, Jan 2013.

[16] P. Mathiyalagan, S. Suriya, Dr. S. N. Sivanandam, "*Modified Ant Colony Algorithm for GridSchedulin*g", International Journal on Computer Science and Engineering (IJCSE), Vol. 02, No. 02, pp. 132-139, 2010.

[17] W. N. Chen, J. Zhang, Y. Yu, "*Workflow scheduling in grids: an ant colony optimization approach*", Evolutionary Computation, CEC, IEEE , Sep 2007.

[18] J. Bagherzadeh,M. MadadyarAdeh, "*An improved ant algorithm for grid scheduling problem*", in 14th International CSI Computer Conference, CSICC, Oct 2009.

[19] K. Li, G. Xu, G. Zhao, Y. Dong, D. Wang, "*Cloud Task scheduling based on Load Balancing Ant Colony Optimization*" Sixth Annual Chinagrid Conference (ChinaGrid) Conference IEEE, Aug 2011.

[20] J. Kennedy, R. Eberhart, "*Particle Swarm Optimization*", in IEEE International Conference on Neural Networks, Dec 1995.

[21] H. Liu,A. Abraham, A. E. Hassanien, "*Scheduling jobs on computational grids using a fuzzy particle swarm optimization algorithm"*, Future Generation Computer Systems, Vol. 26, Issue 8, pp. 1336–1343, Oct 2010.

[22] L Zhang, Y. Chen, B. Yang, "*Task Scheduling Based on PSO Algorithm in Computational Grid*", Sixth International Conference on Intelligent Systems Design and Applications, Oct 2006.

[23] Z. Pooranian, M. Shojafar, J. H. Abawajy, A. Abraham, "*An efficient meta-heuristic algorithm for grid computing", J*ournal of Combinatorial Optimization, Vol 30, Issue 3, pp 413–434, Oct 2015.

[24] X.S. Yang, "*A New Metaheuristic Bat-Inspired Algorithm*", Nature Inspired Cooperative Strategies for Optimization of the series Studies in Computational Intelligence(NICSO) , Vol. 284, pp 65-74, Apr 2010.

[25] L. Jacob, "*Bat algorithm for resource scheduling in cloud computin*g", International Journal for Research in Applied Science and Engineering Technology (IJRASET), Vol 2, Issue 4, Apr 2014.

[26] S Raghavan, P. Sarwesh, C. Marimuthu, K. Chandrasekaran, "*Bat Algorithm for Scheduling Workflow Applications in Cloud",* Electronic Design, International Conference on Computer Networks & Automated Verification (EDCAV), Jan 2015.

[27] S. Joshi, S. Kour, "*Cuckoo search Approach for Virtual Machine Consolidation in Cloud Data Centre*", in International Conference on Computing, Communication and Automation (ICCCA), May 2015.

[28] X. Wen, M. Huang, J. Shi, "*Study on Resources Scheduling Based on ACO Algorithm and PSO Algorithm in Cloud Computing*", 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science, Oct 2012.

[29] Dr. S. George, "*Hybrid PSO-MOBA for Profit Maximization in Cloud Computing*", International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 6, No. 2, pp 159-163, 2015.

[30] G. S. Sadasivam, D. Selvaraj, "*A Novel Parallel Hybrid PSO-GA using MapReduce to Schedule Jobs in Hadoop Data Grids",* Second World Congress on Nature and Biologically Inspired Computing (NaBIC) IEEE, Dec. 2010.

[31] R. Raju, R.G. Babukarthik, D. Chandramohan, P. Dhavachelvan, T. Vengattaraman, "*Minimizing the Makespan using Hybrid Algorithm for Cloud Computing",* 3rd International Advance Computing Conference (IACC) IEEE, Feb 2013.

[32] M. Gendreau, *Handbook of Metaheuristics*, Second ed., vol. 146, New York Dordrecht Heidelberg London: Springer, 2010, pp. 449-468.

[33] C.W. Tsai, W. C. Huang, M. H. Chiang, M. C. Chiang, C.S. Yang, "*A Hyper-Heuristic Scheduling Algorithm for Cloud*", IEEE Transactions on Cloud Computing, Vol. 2, No. 2, pp 236-250, April-June 2014.

Authors Profile

Mr. Akhilesh Jain pursed Bachelor of Engineering from University Rajiv Gandhi Prodyogiki Vishwavidyalay Bhopal in 2014. He is currently pursuing Master of Engineering Department of Computer Science & Engineering from Institute of Engineering & Science, IPS Academy Indore, since 2015. His main research work focuses on Cloud scheduling Algorithms, internet computing, scheduling optimization.

Mr Arvind Upadhyay pursed Bachelor of Engineering from Rajiv Gandhi Prodyogiki Vishwavidyalay Bhopal in 2004 and Master of Engineering in 2010. He is currently pursuing Ph.D. from Devi Ahilya Vishwavidyalaya Indore and currently working as Associate Professor in Department of Computer Science & Engineering, Institute of Engineering & Science, IPS Academy Indore. He is a member of computer society of India since 2013. He has published more than 15 research papers in international journals & conferences. His main research work focuses on Cloud Scheduling & Networking. He has 13 years of teaching experience.