# DIVERSIFIED MULTI-OBJECTIVE WORKFLOW HEURISTIC AND META-HEURISTIC SCHEDULING ALGORITHMS FOR CLOUD ENVIRONMENTS

Meenakshi Bist[1], Parag Jain[2], Amit Agarwal[3]
[1]SGRRITS, Dehradun, [2]RIT, Roorkee, [3]UPES, Dehradun

**Abstract**

**Internet based computing that provide computing resources on demand at a price is cloud computing, now the question arises how does the cloud allocates infrastructure or resources to tasks , it is done with the help of scheduling. Diversified multi-objective workflow scheduling are discussed and reviewed. It is proposed that various multi-objective workflow scheduling be simulated in open source cloud platforms such as Open Stack, Cloud Stack, Open Nebula etc.**

**Introduction:**

Cloud computing has become a noteworthy technology trend, generating revenue in billions and many experts expect that Cloud computing will be the next big thing in the Information Technology. It has become a service provider for the companies which spend frugally. With the cloud computing technology, users use a variety of devices, including PCs, laptops, latest mobiles, and PDAs to access various applications, programs, storage, and application-development platforms over the Internet via services offered by cloud computing providers such as Google, Amazon, Microsoft etc. Advantages of the cloud computing technology include minimizing cost, excessive availability of resources, and easy scalability.

Scheduling is the method that maps the execution of interdependent tasks on the distributed set of resources. Scheduling mechanism takes into account numerous factors such as locality, energy efficiency, cost, reliability, performance and many more. Scheduling can be done in single cloud or multi-cloud scenarios. In workflow scheduling we allocate resources to each task of the workflow and determine the order of execution so that one or more than performance criterions are met. Workflow scheduling may be based on heuristics such as Min-Min, Max-Min, HEFT(Heterogeneous-Earliest-finish -time) . they might be based on meta-heuristics such as Genetic algorithm, Simulated annealing and many more. Workflow scheduling in clouds is a NP-Hard problem.

**Related Work**

Since my topic is to prove a framework for diversified and improved Multi-objective workflow scheduling for clouds I had to review few other objectives under consideration such as Deadline –constrained co-evolutionary, Energy aware workflow scheduling, Cost Trade-off, deadline constrained for multi-core resources, budget constrained scheduling algorithm, BOT(Bag of Task) workflows, yet many criterions on which workflow scheduling can be enhanced. I have considered few more criterions on which performance of workflow scheduling in cloud environments can be enhanced. Current related work for searching more objectives on which enhanced multi-objective workflow scheduling can be framed is given below:

Aarti et.al. [20] Categorised Scheduling into three categories. The first is dynamic scheduling such as round robin based resource selection, Dynamic resource allocation etc. The second category comprises of Agent based scheduling algorithm such as IMAV(Intelligent Multi-Agent for Virtualization), ARAM(Agent based resource allocation model) ,adaptive resource allocation model, Market based model etc. Category third comprised of Cost Optimization based scheduling algorithms such as CTC(Compromised-time-cost scheduling algorithm), Optimal resource allocation technique etc.

Elzeki et.al. [21] proposed improved Max-Min for cloud environments. Max-Min algorithm is the one in which large tasks are executed fist and then the small ones. In this scenario the small tasks had to wait for a longer period of time, while Min-Min executes smaller tasks first than the larger ones. Improved Max –Min algorithm uses both Max –Min and Min –Min. What the improved algorithm does is that in original algorithm, it selects the task with maximum execution time and assign it to the resource with minimum completion time is replaced by select the task with maximum completion time and then assign it to be executed by resource with minimum execution time. Improved Max-Min is compared with Min-Min, Max-Min, RSA and was found that proposed algorithm schedules task with same make span or less than others.

Kushwah et.al.[22] simulated ACO under fault tolerance. It means that correct and continuous operations are performed even in the case of faulty components. Effective error and latent error computing are the two different phases of fault tolerance. Many different fault-tolerant techniques used in cloud computing. One self healing based on divide and conquers where applications running on different VMs and if individual instances fail, they are automatically taken care of. Second is Replication is done using Hadoop , Amazon EC2, HA-Proxy like tools. third is Task Resubmission where the lost tasks are resubmitted . last one is Job Migration where if the machine fails then by using HA-Proxy task are migrated to working machines.

Monte-Carlo method used for reliability-aware workflow scheduling by Rehani et.al.[23] proposed that monte- carlo can correctly model a complex system and give results that are near to complex system operations. This method can also minimize computation time by using divide and merge pattern for parallelization. The author proposed FARS( Failure Aware Resource Scheduling) algorithm in which a cloud is modelled which is used to simulate cloud environment by using Monte Carlo Simulation(MCS) clubbed together with Weibull distributed failures. In the proposed algorithm MCFE(Monte Carlo Failure Estimation) mechanism is used to check the availability and non-availability states for each Virtual Machines. FARS Algorithm is an extension of the famous HEFT algorithm. The proposed algorithm is compared with HEFT using cloudsim toolkit[24] using makespan as their performance metrics. FARS algorithm performed better than HEFT as author increases the value of CCR. When the number of task increases, failures too increase and thus makespan of HEFT also increases. As the task graph increases the FARS performs better. FARS provides reliable allocation of tasks to various resources by using the statistics provided by MCFE algorithm.

Scientific workflows such as Montage, Cybershake, Sipht etc. are specific type of WFMS(Work flow Management systems). A scientific workflow system is which sketches the execution of sequence of computational tasks in a scientific application. Jain et. al.[24] applied all four algorithm ( FCSS, Round Robin, Min-Min, Max-Min) on different scientific workflows and compared with respect to execution time and cost to find results.

Ali et.al . [25] proposed grouped task algorithm that is scheduled to , CC network by applying QoS to user. The proposed algorithm used the methods of improved cost-based-algorithm, TS algorithm and Min-Min algorithm.GTS algorithm divides the task into categories such as long, urgent and user tasks .the performance metrics taken into consideration for these three algorithms were latency for long tasks, execution time span load balancing. The objective of this algorithm gets minimum execution time to all tasks with low latency to tasks with high priority.

Sonia et.al. [1] in her work conveyed the use of Energy aware workflow scheduling. In this type of scheduling the energy consumption of cloud computing resources is considerably reduced by using the parameters of QoS(Quality of Service). In this the author used a hybrid PSO algorithm which helps in reducing cost and enhances the makespan. It also used the DVFS (Dynamic Voltage and Frequency Scaling) technique to considerably minimize the energy consumption .the paper compared the algorithm DVFS-MODPSO with HEFT algorithm given by Topocoglu et. al.[17].the results of the proposed algorithm came out to be better than the HEFT algorithm.

Sanjaya et.al.[2] proposed two phased MOTS(Multi objective task scheduling algorithm for Hetrogeneous multi-cloud environments) the author compared their algorithm with CMMS[18] and PBTS[19] on the criterions of makespan and execution time.

Heyang Xu et. al.[3] proposed algorithm MTCT( Min – Min Based Time and Cost Trade-off) which showed that fault recovery has an impact on the performance of workflow scheduling.
Zhaomeng et. al.[4] proposed EMO (Evolutionary Multi-objective Optimization) algorithm , it solved workflow scheduling on IaaS (Infrastructure as a Service) platform.

Minxian et.al. [5] in his paper proposed a detailed classification which on load balancing algorithm for virtual machine placements in cloud datacenter.

In the paper by Khalili et.al[6], the author used GWO(Grey Wolf optimizer) and worked on dependency graph of workflow tasks. The GWO algorithm exhibits the hunting mechanism used by grey wolves in nature, four types of grey wolf are considered for the experiment. They are alpha, beta, delta and omega, which are used for revivifying the leadership hierarchy.

Chirkin et.al. [7] speaks of runtime estimate that tell us the quality of scheduling. When workflow execution time is estimated , one has to take into account the following factors. First is the dependency between the executed tasks. Second is the heterogeneity of the task. Third is dependency among computing resources. Runtime estimation is also used to give price of the workflow execution when computing resources are leased. The scheduler uses the estimates as random variables and the gives the total information about them. The workflow execution time include extra cost of data transfer, allocation of resources and others. It helps to validate an efficient algorithm to estimate the workflow execution time. On the other hand it helps to implement an estimating system that can be planted into existing schedulers.

Lu et.al.[8] used co-evolutionary approach to adjust the cross-over and mutation probability . This helps in accelerating the convergence and prevents prematurity. This algorithm is compared with Random, HEFT , PSO and Genetic algorithms. The author proposes CGA2(CGA with adaptive penalty function) for Constrained Scientific workflow scheduling . It puts to use adaptive mutation and cross-over probabilities which is based on co-evolution research . Experiments produced results better than PSO, GA, HEFT and Random Scheduling algorithms.

Leena et.al.[9] proposed algorithm for workflow scheduling in hybrid that minimizes both cost and time. Hybrid cloud combines both Public and Private Cloud. BIP(Binary Integer Programming) and Bi-Objective Optimization is considered for mathematically formulating the problem.

Rezaeian et. al.[10] proposes how to make decisions about scheduling sensitive tasks on private cloud while it puts non-sensitive tasks on public cloud . this is done to reduce the makespan , while budget limitation demanded by the user is satisfied. Proposed algorithm shows that the execution of sensitive tasks on private cloud , which helps to achieving at least 7% lower makespan.

Rana et.al.[11] discussed the different stages for executing workflow. First one is the resource provisioning phase based on QoS(Quality of Service) parameter where the computing resources are selected. Second one is task submission phase where a schedule of tasks that are deployed to suitable resources selected in the above phase is created. It focuses on optimizing more than one scheduling parameters such as execution time ,cost, deadline and cost, budget and deadline. Scheduling of workflows in IaaS cloud is a deadline constrained problem. Since Scheduling is NP-Complete problems so there are chances to improve the results of implemented problems. The author proposed an "Enhanced Max- Min" algorithm. EMM selects the resources for scheduling tasks depending on the execution time of resources. It did not consider the properties of the while allocating resources to tasks. The author extended the 'Enhanced Max-Min' algorithm to a Multi-objective Workflow Scheduling which included the parameters such as memory of resources , CPU speed by prioritizing the resources. The resources of scheduling tasks of workflows are chosen based on priority of the resource.

Geeta et.al.[12] proposed the comparison of cloudlet Scheduling algorithms. The author talked of various Scheduling QoS metrics such as execution time , turnaround time, response time, fairness, fault-tolerance, resource utilization, latency etc. discussion on different types of cloudlet scheduling such as real –time , static ,dynamic, heuristic, workflow cloud service scheduling was done. Various linear algorithms like FCFS, RR, Best-Fit, Worst-fit, priority scheduling are reviewed.

Deldari,et.al.[13] proposes workflow scheduling algorithm that minimize the execution cost while considering a user deadline constrained for multi-core resources on cloud. Since multi-core resources have higher leasing cost so a CCA(Cluster Combining algorithm ) has been proposed so that the problem of workflow scheduling on the multi-core cloud has been removed. The research has been divided into two parts, firstly the workflow is clustered by a clustering algorithm. In the Second part the author chooses the best cluster combination available with the help of novel scoring approach which maps cluster tasks on multi-core processing resources. This is done step by step. The CCA algorithm consists of two phases. First phase is called pre-clustering here workflow is divided into different clusters and each cluster is executed on a single-core processing resource, whereas in the second phase which is called combining and mapping, a priority is assigned to primary clusters. These clusters are then combined so that the total cost of the workflow is minimized. The utilization of the processing resource is minimized and makespan meets its defined deadline. After the combination of clusters is performed , the scheduling algorithm decides the processing resource most appropriate for executing the resulting cluster. After cluster combination a mapping phase is performed which maps the tasks of the resulting cluster on the cores of the resources. The proposed algorithm compared with HCOC(hybrid cloud optimized cost ) by Biltencourt et.al.[14].

Cai et.al.[15] discussed algorithm on BOT(Bag of Task) workflows . BoT workflows are widespread in various big data analysis fields. Very few algorithms are catering to BoT workflows. Existing algorithm in this field fail to consider the stochastic task execution times of BoT workflows. This leads to increased resource renting cost and deadline violations.

Ramezani et.al.[16] proposed a MO-LB(Multi-objective Load Balancing System) which transfers extra workload from a set of VMs which are allocated to Physical Machine to other clustered VMs. It contains CPU usage prediction (CUP), which does the following, first it predicts VMs performance and secondly it find out the most appropriate VMs that can execute extra workload.

**Discussion**

Scheduling in cloud environments are implementing using various parameters such as cost, throughput, trust, resource utilization, computational time, degree of imbalance, compromised time, latency, priority, performance , bandwidth , SLA (Service Level Agreements), availability of resources, QoS, energy efficient, adaptive, dynamic, hybrid, simulated annealing based, deadline based, market oriented, profit driven, data security and many more to come in the future. After analysing various multi-objective scheduling algorithms, like versions of HEFT (MOHEFT, SHEFT), Min-Min and improved Min-Min, Max-Min and Max-Min with other constraints, Genetic Algorithm, simulated annealing clubbed with various constraints using private ,public or hybrid cloud platforms. Various bio-inspired algorithms such as ACO and its improved version, Frog-Leaping algorithms , Honey –Bee Optimization, firefly algorithms, grey wolf algorithms etc.

**Conclusion:**

In numerous and diversified parameters are use to improve the performance of work-flow scheduling. There are many parameters such as SLA, energy efficiency, latency and many more can be combined with bio-inspired workflow scheduling, linear scheduling, dynamic scheduling, genetic algorithms, ACO, PSO etc. The scope of combining four or more parameters or mathematical functions so as to achieve better solutions than the previous multi-objective cloud scheduling algorithms. The algorithms reviewed in this paper were simulated on CloudSim, WorkflowSim, MATLAB etc. But we can also use various open source cloud platforms such as

Cloudstack, OpenNebula, OpenStack, Delta, GreenCloud simulators and many more.

**References:**


1. Yassa, S., Chelouah, R., Kadima, H., & Granado, B. (2013). Multi-objective approach for energy-aware workflow scheduling in cloud computing environments. The Scientific World Journal, 2013.
2. Panda, S. K., & Jana, P. K. (2015, January). A multi-objective task scheduling algorithm for heterogeneous multi-cloud environment. In Electronic Design, Computer Networks & Automated Verification (EDCAV), 2015 International Conference on (pp. 82-87). IEEE.
3. Xu, H., Yang, B., Qi, W., & Ahene, E. (2016). A Multi-objective Optimization Approach to Workflow Scheduling in Clouds Considering Fault Recovery. TIIS, 10(3), 976-995.
4. Zhu, Z., Zhang, G., Li, M., & Liu, X. (2016). Evolutionary multi-objective workflow scheduling in cloud. IEEE Transactions on Parallel and Distributed Systems, 27(5), 1344-1357.
5. Xu, M., Tian, W., & Buyya, R. (2016). A Survey on Load Balancing Algorithms for VM Placement in Cloud Computing. arXiv preprint arXiv:1607.06269.
6. Khalili, A., & Babamir, S. M. (2017). Optimal scheduling workflows in cloud computing environment using Pareto-based Grey Wolf Optimizer. Concurrency and Computation: Practice and Experience.
7. Chirkin, A. M., Belloum, A. S., Kovalchuk, S. V., Makkes, M. X., Melnik, M. A., Visheratin, A. A., & Nasonov, D. A. (2017). Execution time estimation for workflow scheduling. Future Generation Computer Systems.
8. Liu, L., Zhang, M., Buyya, R., & Fan, Q. (2016). Deadline-constrained coevolutionary genetic algorithm for scientific workflow scheduling in cloud computing. Concurrency and Computation: Practice and Experience.
9. Leena, V. A., AS, A. B., & Rajasree, M. S. (2016). Genetic algorithm based bi-objective task scheduling in hybrid cloud platform. International Journal of Computer Theory and Engineering, 8(1), 7.
10. Rezaeian, A., Abrishami, H., Abrishami, S., & Naghibzadeh, M. (2016, April). A Budget Constrained Scheduling Algorithm for Hybrid Cloud Computing Systems Under Data Privacy. In Cloud Engineering (IC2E), 2016 IEEE International Conference on (pp. 230-231). IEEE.
11. Rana, S., Choudhary, A., & Mathai, K. J. (2016, March). A critical analysis of workflow scheduling algorithms in infrastructure as a Serivce Cloud and its research issues. In Electrical, Electronics and Computer Science (SCEECS), 2016 IEEE Students' Conference on (pp. 1-6). IEEE.
12. Geetha, V., Devi, R. A., Ilavenil, T., Begum, S. M., & Revathi, S. (2016, February). Performance comparison of cloudlet scheduling policies. In Emerging Trends in Engineering, Technology and Science (ICETETS), International Conference on (pp. 1-7). IEEE.
13. Deldari, A., Naghibzadeh, M., & Abrishami, S. (2016). CCA: a deadline-constrained workflow scheduling algorithm for multicore resources on the cloud. The journal of Supercomputing, 1-26.
14. Bittencourt, L. F., & Madeira, E. R. M. (2011). HCOC: a cost optimization algorithm for workflow scheduling in hybrid clouds. Journal of Internet Services and Applications, 2(3), 207-227.
15. Cai, Z., Li, X., Ruiz, R., & Li, Q. (2017). A delay-based dynamic scheduling algorithm for bag-of-task workflows with stochastic task execution times in clouds. Future Generation Computer Systems, 71, 57-72.
16. Ramezani, F., Lu, J., Taheri, J., & Zomaya, A. Y. (2017). A Multi-Objective Load Balancing System for Cloud Environments. The Computer Journal.
17. Topcuoglu, H., Hariri, S., & Wu, M. Y. (2002). Performance-effective and low-complexity task scheduling for heterogeneous computing. IEEE transactions on parallel and distributed systems, 13(3), 260-274.

18. Delavar, A. G., & Aryan, Y. (2012). A goal-oriented workflow scheduling in heterogeneous distributed systems. International Journal of Computer Applications, 52(8).

19. Calheiros, R. N., & Buyya, R. (2014). Meeting deadlines of scientific workflows in public clouds with tasks replication. IEEE Transactions on Parallel and Distributed Systems, 25(7), 1787-1796.

20. Singh, A., & Malhotra, M. (2013). A comparative analysis of resource scheduling algorithms in cloud computing. Am. J. Comp. Sci. Eng. Surv, 1(1), 1-19.

21. Elzeki, O. M., Reshad, M. Z., & Elsoud, M. A. (2012). Improved max-min algorithm in cloud computing. International Journal of Computer Applications, 50(12).

22. Kushwah, V. S., & Goyal, S. K. (2017). A Basic Simulation of ACO Algorithm Under Cloud Computing for Fault Tolerant. In Proceedings of the International Conference on Data Engineering and Communication Technology (pp. 465-472). Springer Singapore.

23. Rehani, N., & Garg, R. (2017). Reliability-Aware Workflow Scheduling Using Monte Carlo Failure Estimation in Cloud. In Proceedings of International Conference on Communication and Networks (pp. 139-153). Springer, Singapore.

24. Jain, A., & Kumari, R. (2017). A Review on Comparison of Workflow Scheduling Algorithms with Scientific Workflows. In Proceedings of International Conference on Communication and Networks (pp. 613-622). Springer, Singapore.

25. Ali, H. G. E. D. H., Saroit, I. A., & Kotb, A. M. (2017). Grouped tasks scheduling algorithm based on QoS in cloud computing network. Egyptian Informatics Journal, 18(1), 11-19.