International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-11 Issue-1, October 2021

Comparison of Algorithms for Workflow Applications in Cloud Computing

Sriperambuduri Vinay Kumar, M. Nagaratna

Abstract: ***Cloud computing model has evolved to deliver resources on pay per use model to businesses, service providers and end-users. Workflow scheduling has become one of the research trends in cloud computing as many applications in scientific, business, and big data processing can be expressed in the form of a workflow. The scheduling aims to execute scientific or synthetic workloads on the cloud by utilizing the resources by meeting QoS requirements, makespan, energy and cost. There has been extensive research in this area to schedule workflow applications in a distributed environment, to execute background tasks in IoT applications, event-driven and web applications. This paper focuses on the comprehensive survey and classification of workflow scheduling algorithms designed for the cloud.***

Keywords: ***cloud computing, makespan, scheduling, scientific workflow, workflow.***

I. INTRODUCTION

With the enormous growth of the internet and wide range usage of social networking applications and IoT applications, the data proliferation has become a challenge to store, manage and perform analysis. Cloud computing has become the platform of choice, as it offers pay-per-use resources with elastic provisioning and availability. The widespread deployment of applications on cloud posed a challenge to allocate the resources for an application meeting the QoS constraints. As per authors in [1] meeting provider's resource to cost and energy requirements are a challenging problem in scheduling. Most of the applications consist of set cooperative tasks that require a good amount of computing capacity to run the application. An application can be represented in the form of Directed Acyclic Graph (DAG) called as workflow. These kinds of applications are generally either big data related workflows, web service based workflows or scientific workflows. Many authors have been studying the workflow scheduling for the past several years. There has been a considerable amount of work that has been carried out in the literature, but there is still scope for research in the light of IoT, Fog computing posing a different challenge in scheduling. The workflow scheduling problem being an NP- Hard [2], authors have applied various static and dynamic strategies to solve the problem. The current review presents (a) the mathematical model of workflow (b) taxonomy for workflow scheduling algorithms based on various objectives and constraints chosen.

Manuscript received on September 02, 2021.
Revised Manuscript received on September 20, 2021.
Manuscript published on October 30, 2021.
* Correspondence Author
Sriperambuduri Vinay Kumar*, Research Scholar, JNTUH College of Engineering, Hyderabad, India. Email: vinayperambuduri@gmail.com
Dr. M. Nagaratna, Professor, JNTUH College of Engineering, Hyderabad, India. Email: mratnaraju@jntuh.ac.in

II. MATHEMATICAL MODEL OF WORKFLOW

A workflow is run on a VM instance in the cloud environment. A workflow consists of a set of tasks. These tasks are represented by nodes and dependencies between the tasks are represented by the edges in the form of a DAG shown in Fig.2.1. A workflow W (J, E) consists of J tasks J= $\{J_1, J_2, J_3 \ldots, J_n\}$, and directed edges E, represented by (J_i, J_k) where J_i, $J_k \in J$. Edge E represents the communication time between tasks Ji and Jk. A node with no outgoing edge is referred to as a task in an exit state. The time taken to send data from task J_i to the other task J_k referred to as communication time (CT). CT is dependent on whether the tasks are running on the same server in a Virtual Machine (VM) or server running on different VMs. The workflows are managed and executed on a server by a system referred to as a workflow management system [3]. Pegasus and DAGMan are workflow generators that generate the workflows to be submitted by the user to the cloud [20].

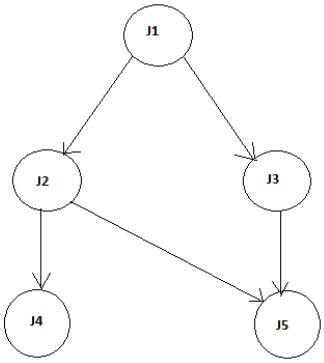

Fig. 1. An example of a Workflow.

III. TAXONOMY OF SCHEDULING ALGORITHMS

There are different types of scheduling algorithms implemented for scheduling workflows taking into consideration varied objectives and constraints. Below are some of the widely popular scheduling algorithms implemented for Workflow applications.

a) PSO algorithm: Creates a model to assign the tasks to the processor in a way that the total execution cost is reduced/ minimized, considering its computation and communication costs. The tasks are mapped to cloud resources optimally. To optimize the overall cost, the costs are updated in each scheduling loop [4].

b) Integer Linear Program (ILP): ILP utilizes the resources of IaaS providers and SaaS for execution of workflow tasks. The algorithm could produce low cost solutions to short deadlines and is able to fulfill QoS constraints by the VMs [5].

c) Scalable HEFT: Applied a two-phase strategy to schedule workflows. One task prioritizing phase and two resource selection phases. It could optimize the execution time and increase the scalability of resources [6].

d) Improved Genetic algorithm: The approximate optimal scheduling is generated using the automated scheduling policy. A solution was determined that satisfies all the user preferred QoS constraints, including the constraints such as improved CPU utilization [7].

e) Multiple QoS constrained scheduling algorithm: In this scheduling algorithm [8], different QoS constraints are considered. The main four factors, i.e., scheduling success rate, QoS requirements, mean execution cost, mean execution time are the factors to impact on the makespan and the total cost of execution of a workflow.

f) Deadline and Budget Distribution based Cost-Time optimization (DBD-CTO): In [9] DBD-CTO, two main constraints are considered – target, deadline. Considering these two constraints the computation cost of completion of tasks in timeline is minimized.

g) Revised Discrete PSO: In this algorithm, execution time and data transfer cost are taken into consideration to schedule the tasks. Initialization of a swarm is performed through the Greedy randomized adaptive search process. Particle's position is computed based on the selection of pairs with higher probability and learning from previous positions. This one performed well on the makespan, saving the cost, and cost optimization ratio, over the other algorithms [10].

h) Improved cost-based scheduling: Authors have grouped the user tasks based on resource capabilities. The tasks were executed on these resources that met the capabilities thus improving the computation/communication ratio. [11].

i) Heuristic based Genetic algorithm: The authors in [12] have assigned the priority to the tasks in a synthetic workflow using bottom-level (b-level) and top-level (t-level). The priorities improved the diversity of the population and then used to initialize the population of HGAs. The performance is much better than the regular standard Genetic algorithm (SGA) [12].

j) Improved Round Robin algorithm: the algorithm assigns jobs to the respective cloud nodes initially, however, once the nodes enter the idle state the order in which the jobs are assigned, is changed. It ensures that the job is not allocated to the first node always thereby improving the turn-around time, avg. waiting time and response time. [13].

k) Multi-Objective Privacy Aware Workflow Scheduling Algorithm MOPA: In this algorithm, authors modeled the problem by employing an encoding strategy that includes privacy protection constraints along with workflow schedule. Pareto trade-off solutions were achieved with minimal cost and execution time. The quality of the solutions of this algorithm are found to outperform both the algorithms MNSGA-II and MMOPSO [19][14].

l) Deadline Constrained workflow scheduling Algorithm: It has reduced the cost of execution by running the algorithm in two phases. In phase one K-means clustering technique was applied for consolidation of VM's based on speed. In phase two, level-based scheduling using a partition method is applied to dynamically allocate the VMs or resources with deadline as constraint. A data recovery mechanism is also provided by using a centralized storage model. The time complexity of this algorithm is $O(n^2)$ which is polynomial time complexity [15].

m) Completion Time Driven Hyper-Heuristic (CTDHH) Algorithm: In this algorithm, heuristic and meta-heuristics are applied on a large search space to find solutions. It employed four heuristic algorithms as a low-level strategy that effectively guides the search process. The results depict that it is more efficient than the existing approaches [16].

n) Hybrid Algorithm: The hybrid algorithm applies preprocessing steps to before applying PSO. The tasks are first sorted based on the large number of descendants are placed in list one. A second list is prepared taking into consideration processing power. Parent tasks that require high processing nodes are executed to minimize dependencies. Later, children are processed based on their position. The algorithm then uses PSO for mapping the tasks in both lists to the resources. It could reduce the cost, execution time and was able to balance the load among the nodes. [17].

o) Cost and Energy Aware Scheduling Algorithm (CEAS): It utilizes three sub algorithms, where first one maps tasks to VMs using sub-makespan constraint, second one, uses two tasks merging method, third one proposes VM reuse policy and fourth uses slack reclamation to save energy. It could reduce the computation cost and total energy consumption with each sub-algorithm taking polynomial time [18].

p) Neural Network based MOEA: It employs NSGA-II algorithm with neural networks to predict changing objectives and failures. They have determined the effects of varying frequency of resource failures and the effects on the performance of algorithms when the number of objectives are increased. The proposed algorithm exploits the history of Pareto-optimal set (POS) to estimate POS after the change occurrence [21].

IV. RESULT OF COMPARISON OF ALGORITHMS

The Table-I below shows, the results of comparison of various algorithms. The parameters considered for comparison in Table-I are based on objectives considered, limitations of the algorithm/method and environment used to perform the experiments on. Table-II is based on QoS parameters.

Retrieval Number: 100.1/ijeat.A31391011121
DOI: 10.35940/ijeat.A3139.1011121
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Table- I: Comparison of algorithms

Algorithm	Development Environment used	Objectives considered	Drawbacks/ Limitations
Particle Swarm Optimization	Amazon EC2	• Time • Cost Optimization • Resource utilization	• Local Optimum in high-dimensional space • The Convergence rate is low in iterative process
Integer Linear Program	Java	• Makespan • Cost • Time	• Does not consider linear effects. • Risk of the high dimensionality of the problem
Scalable HEFT	CloudSim	• Scalability • Execution time	• Not effective for small sized workflows as that of larger workflows
Improved Genetic Algorithm	Eucalyptus	• Execution time • CPU utilization • Resource utilization	• There is a chance of premature convergence of the solution.
Multiple QoS-constrained scheduling algorithm	CloudSim	• Time • Cost • Makespan • The success rate of scheduling	• Does not perform good w.r.t mean execution time on the smaller workload. • As the concurrent workflows' size increases, the mean execution cost gives results like the other algorithms
DBD-CTO algorithm	Java	• Cost • Time	• The success rate is not more than other methods. • Still has a scope for improvement in the execution time and execution cost
Revised Discrete PSO	Amazon Elastic Compute Cloud	• Makespan • Cost Optimization	• Sensitive to the bias associated with sampling. • It has high variance in terms of performance
Improved cost based scheduling algorithm	CloudSim	• Cost • Performance	• Could not handle complicated scenarios such as the dynamic nature of cloud • Does not consider other QoS attributes
Heuristic based Genetic algorithm	Java	• Execution cost • Execution time • Data transmission cost	• Does not consider other QoS factors • Results are not very comparable to other popular algorithms
Improved Round Robin	Java	• Time • Makespan	• Does not consider the priority of the processes/ tasks. • If priority is considered, then it may lead to longer waiting periods of normal priority processes
MOPA	CloudSim	• Execution time • Monetary cost • Minimizing security overhead • Maximizing total security	• Other objective such as energy consumption is not considered
Deadline constrained PPDPS	Amazon EC2 Cloud	• Cost-efficient with deadline as the constraint • Dynamic Provision of VMs	• Need to analyze the problems that might occur in multi clouds that uses p2p communication.
Hyper-Heuristic Cost Optimisation	Java	• Cost Optimization • Completion time	• Takes more time to initialize than the other algorithms.
Hybrid Algorithm	CloudSim	• Minimize execution time • Reduce cost • Load balance	• Not considers energy consumption • Fault tolerance not considered
Cost and Energy Aware Scheduling Algorithm	CloudSim	• Minimize execution cost • Reduce energy consumption • Deadline constrained	• Not considered the energy consumption of hard drive and memory.
Neural Network based MOEA	CloudSim	• Reduce Makespan • Cost • Energy • Maximize Reliability • Utilization	• Fault tolerance and priority are not considered

Table- II: Comparison of algorithms based on QoS parameters

Algorithms/ Performance Metrics	Execution Time	Cost Optimization	Resource Utilization	Makespan	Scalability	Scheduling success rate	Data transfer cost	Privacy/ Security features	Energy Consumption
Particle Swarm Optimization	Y	Y	Y	Y					
Integer Linear Program	Y	Y		Y					
Scalable HEFT	Y			Y	Y				
Improved Genetic Algorithm	Y	Y		Y					
Multiple QoS constrained scheduling algorithm	Y	Y		Y		Y			
DBD-CTO algorithm	Y	Y							
Revised Discrete PSO	Y	Y	Y	Y					
Improved cost based scheduling algorithm	Y	Y		Y					
Heuristic based Genetic algorithm	Y	Y					Y		
Improved Round Robin	Y			Y					
MOPA	Y	Y		Y				Y	
PPDPS	Y	Y	Y	Y	Y				
Hyper-Heuristic Cost Optimization		Y		Y					
Hybrid Algorithm	Y	Y							
Cost and Energy Aware Scheduling Algorithm	Y	Y		Y					Y
Neural Network based MOEA	Y	Y	Y	Y					

Y – Does Optimize

V. CONCLUSION AND FUTURE WORK

To summarize, in this paper we have analyzed and compared existing approaches to schedule the workflow applications in cloud. The analysis done is represented in a table with the algorithm used, objectives considered, and development environment along with another table indicating the QoS parameters. Based on the analysis the optimized areas in workflow scheduling are execution time, cost optimization, makespan, resource utilization while considering deadline constraint. There is scope for further research in this area with objectives as reliability, privacy/security, backup, energy consumption with constraints as fault tolerance, the budget, and the priority.

REFERENCES

1. J. Baliga, R. W. A. Ayre, K. Hinton, and R. S. Tucker, "Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport," Proceedings of the IEEE, vol. 99, no. 1, pp. 149–167, 2011.
2. J. Ullman, "NP-complete scheduling problems," Journal of Computer and System Sciences, vol. 10, no. 3, pp. 384–393, 1975.
3. S. Pandey, D. Karunamoorthy, and R. Buyya, "Workflow Engine for Clouds," Cloud Computing, pp. 321–344, 2011.
4. S. Pandey, L. Wu, S. M. Guru, and R. Buyya, "A Particle Swarm Optimization-Based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments," 2010 24th IEEE International Conference on Advanced Information Networking and Applications, 2010.
5. T. A. L. Genez, L. F. Bittencourt, and E. R. M. Madeira, "Workflow scheduling for SaaS / PaaS cloud providers considering two SLA levels," 2012 IEEE Network Operations and Management Symposium, 2012.
6. C. Lin and S. Lu, "Scheduling Scientific Workflows Elastically for Cloud Computing," 2011 IEEE 4th International Conference on Cloud Computing, 2011. doi: 10.1109/CLOUD.2011.110.
7. H. Zhong, K. Tao, and X. Zhang, "An Approach to Optimized Resource Scheduling Algorithm for Open-Source Cloud Systems," 2010 Fifth Annual ChinaGrid Conference, 2010.
8. M. Xu, L. Cui, H. Wang, and Y. Bi, "A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing," 2009 IEEE International Symposium on Parallel and Distributed Processing with Applications, 2009.
9. Verma, Amandeep, "Deadline and Budget Distribution based Cost-Time Optimization Workflow Scheduling Algorithm for Cloud," International Journal of Computer Applications, 2012.
10. Wu, Zhangjun & Ni, Zhiwei & Gu, Lichuan & Liu, Xiao, "A Revised Discrete Particle Swarm Optimization for Cloud Workflow Scheduling," 2010 International Conference on Computational Intelligence and Security, 2010.
11. Selvarani, S., & Sadhasivam, G. "Improved cost-based algorithm for task scheduling in cloud computing," 2010 IEEE International Conference on Computational Intelligence and Computing Research, 1-5, 2010.
12. A. Verma and S. Kaushal, "Deadline constraint heuristic-based genetic algorithm for workflow scheduling in cloud," International Journal of Grid and Utility Computing, vol. 5, no. 2, p. 96, 2014.
13. F. Alhaidari and T. Z. Balharith, "Enhanced Round-Robin Algorithm in the Cloud Computing Environment for Optimal Task Scheduling," Computers, vol. 10, no. 5, p. 63, 2021.
14. Y. Wen, J. Liu, W. Dou, X. Xu, B. Cao, and J. Chen, "Scheduling workflows with privacy protection constraints for big data applications on cloud," Future Generation Computer Systems, vol. 108, pp. 1084–1091, 2020.
15. V. Singh, I. Gupta, and P. K. Jana, "A novel cost-efficient approach for deadline-constrained workflow scheduling by dynamic provisioning of resources," Future Generation Computer Systems, vol. 79, pp. 95–110, 2018.
16. E. N. Alkhanak and S. P. Lee, "A hyper-heuristic cost optimisation approach for Scientific Workflow Scheduling in cloud computing," Future Generation Computer Systems, vol. 86, pp. 480–506, 2018.
17. M. Sardaraz and M. Tahir, "A Hybrid Algorithm for Scheduling Scientific Workflows in Cloud Computing," IEEE Access, vol. 7, pp. 186137–186146, 2019.

18. Z. Li, J. Ge, H. Hu, W. Song, H. Hu, and B. Luo, "Cost and Energy Aware Scheduling Algorithm for Scientific Workflows with Deadline Constraint in Clouds," IEEE Transactions on Services Computing, vol. 11, no. 4, pp. 713–726, 2018.
19. V. Seethalakshmi, V. Govindasamy, V. Akila, G. Sivaranjini, K. Sindhuja, and K. Prasanth, "A Survey Of Different Workflow Scheduling Algorithms In Cloud Computing," 2019 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC), 2019.
20. M. Adhikari, T. Amgoth, and S. N. Srirama, "A Survey on Scheduling Strategies for Workflows in Cloud Environment and Emerging Trends," ACM Computing Surveys, vol. 52, no. 4, pp. 1–36, 2019.
21. G. Ismayilov and H. R. Topcuoglu, "Neural network based multi-objective evolutionary algorithm for dynamic workflow scheduling in cloud computing," Future Generation Computer Systems, vol. 102, pp. 307–322, 2020.

AUTHORS PROFILE

Sriperambuduri Vinay Kumar currently working as Asst. Professor in Vasavi College of Engineering, Hyderabad. He has 16 years of teaching experience. His research interests include Cloud Computing, Mobile Computing, Web Programming and Internet of Things.

Dr. M. Nagaratna currently working as Professor in Department of Computer Science and Engineering, JNTUH College of Engineering, Hyderabad. She received her Doctoral Degree in Computer Science from JNTU Hyderabad. She has over 18 years of teaching experience. Her research interest includes Computer Networks, Design & Analysis of Algorithms, Advanced Data Structures, Mobile Computing and Cloud Computing.