TASK SCHEDULING IN CLOUD COMPUTING ENVIRONMENT USING HYBRID GENETIC ALGORITHM AND ARTIFICIAL BEE COLONY

MOHD IZUDDIN BIN SIPLUK

FACULTY OF COMPUTING AND INFORMATICS

UNIVERSITY MALAYSIA SABAH

2022

UMS

UNIVERSITI MALAYSIA SABAH

TASK SCHEDULING IN CLOUD COMPUTING ENVIRONMENT USING HYBRID GENETIC ALGORITHM AND ARTIFICIAL BEE COLONY

MOHD IZUDDIN BIN SIPLUK

THESIS SUBMITTED IN PARTIAL FULFILLMENT FOR THE BACHELOR OF COMPUTER SCIENCE WITH HONOURS (NETWORK ENGINEERING)

UMS
UNIVERSITI MALAYSIA SABAH

FACULTY OF COMPUTING AND INFORMATICS

UNIVERSITY MALAYSIA SABAH

2022

UNIVERSITI MALAYSIA SABAH

NAME : MOHD IZUDDIN BIN SIPLUK

MATRIC NUMBER : BI18110248

TITLE : TASK SCHEDULING IN CLOUD COMPUTING ENVIRONMENT USING HYBRID GENETIC ALGORITHM AND ARTIFICIAL BEE COLONY

DEGREE : BACHELOR OF COMPUTER WITH HONOURS (SOFTWARE ENGINEERING)

VIVA'S DATE : 22 FEBRUARY 2022

CERTIFIED BY;

1. **SUPERVISOR** Signature

ENCIK NORDIN SAAD

DECLARATION

I hereby declare that the material in this thesis is my own except for quotations, equations, summaries, and references, which have been duly acknowledged.

22 FEBRUARY 2022

IZUDDIN SIPLUK
BI18110248

ACKNOWLEDGEMENT

Alhamdulillah, all praises to Allah S.W.T. Finally, I completed my writing related to Task Scheduling in Cloud Computing Environment Using Hybrid Genetic Algorithm and Artificial Bee Colony. Thanks to the family for never ceasing to motivate and encourage me to complete the writing of this Final Year Project Thesis.

Immense appreciation to University Malaysia Sabah, lecturers of the Faculty of Computing and Informatics for all knowledge given for almost four years of studies in this university. Not to forget Mr Nordin Saad as my Final Year Project supervisor, as well as AP Dr Azali Saudi, who gave me a lot of guidance completing this report and project.

Furthermore, I would like to thank both examiner and panel Dr Lai Po Hung and Dr Ervin Gubin Moung, who also gave some guidance and constructive comment for a better project.

And for all my fellow friends who are constantly struggling and hardworking with me, thank you for helping and giving full support for me in this survival. And I wish you all the best in the future.

MOHD IZUDDIN BIN SIPLUK
22 FEBRUARY 2022

ABSTRACT

The fundamental issue with cloud computing is task scheduling and decreasing system performance. An efficient task-scheduling technique is essential to increase system performance. Existing task-scheduling algorithms are primarily concerned with task resource requirements, CPU memory, execution time, and cost. These, on the other hand, do not examine network bandwidth. In cloud computing systems, task scheduling is essential. Task scheduling cannot be done based on a single criterion but rather under a set of rules and regulations that we might refer to as a cloud user-provider agreement. This agreement is more than the user's expectations about the providers' service quality. Providing high-quality services to users following the consensus is a crucial duty for providers, juggling a vast number of other responsibilities. The task scheduling problem can be thought of as discovering or discovering an optimal mapping/assignment of a series of subtasks of various tasks over a set of available resources (processors/computer machines) to fulfil the intended task goals. During the methodology chapter, a comprehensive investigation has been done to ascertain the proposed method that can be adopted such as algorithms involved, project flow, and simulation. This is essential to produce a system that has a feature such as web-based system that is able to generate a report from the simulation. In this project, a comparative evaluation of selected algorithms is done to ascertain their applicability, practicality, and adaptability in a cloud scenario. At the end of the project, the author will attempt to suggest an algorithm that can be utilized to expand the present platform further. As a result, cloud providers will be able to provide higher-quality services.

ABSTRAK

PENJADUALAN TUGAS DALAM PERSEKITARAN PENGKOMPUTERAN AWAN MENGGUNAKAN ALGORITMA GENETIK HIBRID DAN KOLONI LEBAH BUATAN

Isu asas dengan pengkomputeran awan adalah penjadualan tugas dan penurunan prestasi sistem. Teknik penjadualan tugas yang cekap adalah penting untuk meningkatkan prestasi sistem. Algoritma penjadualan tugas sedia ada terutamanya berkaitan dengan keperluan sumber tugas, memori CPU, masa pelaksanaan, dan kos. Ini, sebaliknya, tidak memeriksa jalur lebar rangkaian. Dalam sistem pengkomputeran awan, penjadualan tugas adalah penting. Penjadualan tugas tidak boleh dilakukan berdasarkan kriteria tunggal tetapi sebaliknya di bawah satu set peraturan dan peraturan yang mungkin kami rujuk sebagai perjanjian pembekal pengguna awan. Perjanjian ini lebih daripada jangkaan pengguna mengenai kualiti perkhidmatan penyedia. Menyediakan perkhidmatan berkualiti tinggi kepada pengguna berikutan konsensus adalah tugas penting bagi penyedia, mengimbangi sejumlah besar tanggungjawab lain. Masalah penjadualan tugas boleh dianggap sebagai menemui atau menemui pemetaan / tugasan yang optimum satu siri subtugas pelbagai tugas melalui satu set sumber yang ada (pemproses / mesin komputer) untuk memenuhi matlamat tugas yang dimaksudkan. Siasatan menyeluruh telah dilakukan untuk memastikan kaedah yang dicadangkan yang boleh diguna pakai seperti algoritma yang terlibat, aliran projek dan simulasi. Ini adalah penting untuk menghasilkan sistem yang mempunyai antara muka pengguna dan boleh menjana laporan. Pada akhir projek, penulis akan cuba mencadangkan algoritma yang boleh digunakan untuk mengembangkan platform ini lebih jauh. Akibatnya, pembekal awan akan dapat menyediakan perkhidmatan berkualiti tinggi.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Cloud computing is a new technology that assures end-users access to cloud resources such as storage, processing, memory, and applications [1]. Enterprises use cloud computing to conduct computationally heavy operations, and the computing resources are dispersed across the internet and can be accessed anywhere. Many stakeholders may be found in a cloud computing environment, with cloud service providers, brokers, and service consumers among the most prominent [2]. The cloud service provider and the cloud service consumer are the two key entities immediately affected by task scheduling. Both parties have their own goals in mind, but the best performance is a win-win situation for everyone.

The cloud broker [3] serves as a middleman between the cloud service provider and the user. The goal of a cloud service provider is to have a high degree of customer satisfaction. Because cloud networks give services to users on-demand, resource availability is required [4]. Cloud service providers ensure that their cloud systems are wholly functional and optimized to meet the needs of their valued customers. Cloud services are divided into three categories based on how they are delivered in hardware and software. Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) are the terms used to describe these services [1, 2, 3].

i. In IaaS, the customer is given access to the cloud infrastructure, including networks, storage, processing, and applications.
ii. PaaS provides the user with an internet-based platform for the development of apps. Developers can develop and test their apps on tools available in cloud machines.
iii. In SaaS, software applications are delivered to users via the internet, eliminating installing software on their computers.

Cloud service providers seek optimal resource use, improving the user experience by offering the quality of services [5] that adhere to QoS parameters. Much research has been done in this area to determine the best algorithm for mapping tasks to virtual machines, and this study proposes an evolutionary approach for identifying the appropriate task-to-VM mapping.

1.2 Problem Statement

In cloud computing, the task scheduler must choose an optimal mapping of jobs to VMs from an abundant search space of alternative mappings, discussing factors such as time of execution and number of tasks. A few known algorithms have been tested on big datasets based on genuine cloud traces. Existing schedulers that use various meta-heuristics can be enhanced for better task scheduling, and it is essential to achieve early convergence and better time execution.

1.3 Objectives

i. To implement the hybrid genetic algorithm and artificial bee colony algorithm using CloudSim simulator.
ii. To develop a web system for managing the outcomes from the CloudSim simulation.
iii. To simulate and evaluate the best duration of proposed algorithm in CloudSim simulator with the help of web system.

1.4 Project Scope

The scope of this project is to study and analyse the proposed Hybrid GAABC scheduling algorithm with the existing scheduling algorithm in CloudSim to compare the completion time of jobs. In terms of application, this project is assumed to be utilized in a networking environment such as JTMK UMS, and the target user is a network administrator. A network administrator can adopt this system to manage the task scheduling algorithm to produce better output for utilizing the virtual machine.

1.5 Limitation of works

This project focuses on large-scale cloud data centres, virtualized servers, data centre network topologies, featured clouds, and user-defined policies for assigning hosts to virtual machines and policies for assigning host resources to virtual machines. It is difficult and expensive to implement in the real world. However, the user can better understand the project by simulating the data centre using CloudSim. Aside from that, one of CloudSim's flaws is the absence of a graphical user interface (GUI). However, CloudSim is the most extensively utilized simulation tool for cloud research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter covers the fundamental concepts of the technologies involved and relevant studies and a comparison to existing systems. The objective of a literature study is to help ascertain the appropriate strategy and approach for project advancement and gain an improved understanding of the project.

2.2 What is CloudSim?

CloudSim is an extendable simulation toolkit for seamless cloud computing system modelling and simulation modelling. Because it does not offer a ready-to-use environment for executing a scenario with input, CloudSim is not a framework. CloudSim users must first generate the cloud scenarios they wish to assess, then set the necessary output and input parameters. Cloud Sim was produced in the CloudBus Laboratory at the University of Melbourne's Computer Science and Software Engineering Department [6].

Using CloudSim as a cloud simulator, run through the scheduling criteria CPU utilization. It is best if the CPU is kept as active as possible. The optimal CPU utilization ranges from 40% (lightly loaded) to 90% (heavily loaded) [7]. Before allocating resources to a job, a user needs to estimate all accessible resources and the work at hand to attain the best possible CPU utilization.

2.3 Review of Related Works

Zhong Zong et al. (Zong, 2020) propose pooling dynamic fusion mission planning methodology, genetic approach, and the ant-colony system in 2020. Cloud computing data and storage facilities adopt less energy as a consequence of this. The test results

indicate that the suggested task programming approach would considerably reduce the amount of time and energy spent on cloud computing devices. [8]

Vijayalakshmi A. Lepaksh et al. (Vijayalakshmi A. Lepaksh, 2020) proposed an Efficient Resource Allocation with Score (ERAS) for task scheduling in cloud environments in 2020, which takes into consideration Virtual Machines (VM) temporary operational availability by proposing various kinds of delays and using EFT to set the processor for task scheduling to a normalized score [9]. The findings imply that the improved dependability of the ERAS algorithm delivers superior efficiency than current approaches that just consider EFT for allocations.

Sanj M S et al. (Sanaj M S, 2020) proposed an upgraded Round Robin (ERR) technique to boost efficiency without affecting traditional RR functionality in 2020. The proposed technique is implemented and tested utilizing the CloudSim toolbox [10]. In comparison to classical RR, the findings indicate that the overall waiting time for tasks in a certain number of cloudlets in ERR is minimised under the indistinguishable conditions.

Yong Shi and et al. (Yong Shi, 2020) introduced a BMin algorithm to improve the effectiveness of the Min-min algorithm in 2020. The proposed technique is assessed using the cloudsim simulation programme, and the findings imply that it minimises completion time, maximises throughput, and enchances resource load balancing. [11]

Raja Masadeh et al. (Raja Masadeh, 2019) proposed the vocalisation algorithm for humpback whale optimization in 2019. (VWOA). The VWOA is a cloud system that simulates the vocalisation behaviour of humpback whales and is employed to optimise work scheduling. The VWOA scheduler is based on a proposed multi-objective model. It cuts down on completion time, costs, and energy consumption while maximising resource utilisation [12]. The VWOA scheduler outperformed the normal whale optimization algorithm (WOA) and round-robin (RR) algorithms in terminology of cost, makespan, scope of imbalance, resource utilisation, and energy consumption in the test data.

SHAN CHEN PANG et al. (SHANCHEN PANG 2019) proposed an efficient hybrid algorithm for Distribution Estimation Algorithm and Genetic Algorithm (EDA-GA) to solve the difficulty of multi-objective task scheduling with the objective of reducing task execution time and optimizing the potential to balance the load of the

environment. The proposed method first employs EDA operations to generate a suitable solution, then employs a genetic algorithm to generate further solutions that are dependent on the optimum solution chosen in the prior stage in order to broaden the remit of the search. The data demonstrate that the suggested method has a rapid convergence incidence and a potent search capability. [13]

Shengmei Liu and Yari Yin (Shengmei Liu, 2019) developed the discrete particle swarm optimization method (IDPSO) by utilizing a sinusoidal strategy-based dynamic inertia weight optimization technique to make particles adaptive to different stages in their quest for the best international solution [14]. In terms of completion time and convergence, the IDPSO technique exceeds the DPSO and FCFS algorithms.

In 2019, Abdulsalam Alsmady et al. (Abdulsalam Alsmady, 2019) suggested a Memetic Algorithm (MA) to address cloud workflow-scheduling, utilizing cost and time as two aims for optimising scientific workflow scheduling in a cloud system [15]. The (MA) algorithm employed the local search algorithm for hill climbing as an further operator for the Genetic Algorithm (G.A.) to boost individual solutions during an international search.

[16] This study implemented a Grey Wolf Optimization (GWO) algorithm with fitness function modifications by generating it manage multi-objectives in single fitness; the makpane and cost are the goals exploited in fitness to answer the uncertainty of job scheduling (Abdullah Alzaqebah, 2019). The proposed method's principal purpose is to diminish both cost and time to market. The CloudSim findings suggest that the proposed MGWO algorithm exceeds both the classic GWO algorithm and the Whale Optimization Algorithm (WOA) in terminology of cost, make space, and performance.

Ping Zhu et al. (Ping Zhu, 2019) [17] proposed a power-aware and real-time scheduling (PRTS) way to reduce workflow expenses and energy usage in 2019. The proposed technique has two parts: scheduling the most economical virtual machines based on the crucial path without missing the deadline, tracking dynamic slack, and reclaiming it to adopt the energy-saving DVFS technique.

In 2019, A.M.Senthil Kumar and et al. (A.M.Senthil Kumar, 2019) reported successful hybrid task scheduling utilising Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) approaches to reduce total execution time. When compared to the standard genetic algorithm, Min-Min, and Max-Min algorithms, PSO enabled GA

achieved higher outcomes in the hybrid Genetic Algorithm - Particle Swarm Optimization (GA-PSO algorithm. [18]

[19] Binary coded chromosomes for resource scheduling is recommended as an technique for premature optimization of the threat of cross-over mutation in adaptive genetic algorithms by Fang Yiqiu and colleagues (X. X. Fang Yiqiu, Ge Junwei, 2019). (AGA). When compared to the regular genetic algorithm, the evolved genetic algorithm (AGA) outperforms it (SGA). The CloudSim tool's findings imply that the upgraded approach has a good planning algorithm.

In 2019, Negar Chitgar and et al. (Negar Chitgar, 2019) implemented a enhance for scheduling workload in the cloud system based on virtual machine grouping. The proposed approach seeks to improve the performance of the cloud environment by reducing make span and response time and maximising virtual machine use. The suggested approach outperforms other existing algorithms on a range of performance parameters. [20]

A new preparation method was integrated in 2018 by Shuang YIN (Shuang YIN, 2018), which adopts the doublefitness algorithm-load balance and the cost completion genetic algorithm (LCGA) [21]. The schedule guarantees a load balance, lowering the mission's expense of completion. The LCGA algorithm, which is comparable to the load-balancing genetic algorithm (LGA) and the task-complete cost genetic algorithm (CGA), indicates that the performance and optimization scheme of a programming technique can be applied in a simulation experiment.

In 2018, Fang Yiqiu and Li Xiaosheng (L. X. Fang Yiqiu, 2018) presented a real-time ant colony method for a virtual machine (L. X. Fang Yiqiu, 2018). (VM-ACO). The method takes into consideration time in mandate to complete the task's load balance. In terms of resource state and polling, task latency, time to complete the work, and load balancing, the data demonstrate that the VM-ACO technique exceeds the ant colony algorithm.

Mehran Ashouraei and et al. (Mehran Ashouraei, 2018) developed a parallel genetic algorithm-based technique for scheduling jobs with priority in cloud systems to effectively employ resources and avoid resource waste. To eliminate work failures, this scheme is borne out by boosting the load balance while picking excellent resources for short-term arrival activities. [22]

In 2018, N. Gopalakrishnan and C. Arun introduced a hybrid Genetic Gray Wolf Optimization Algorithm (GGWO) to resolve and enhance work scheduling challenges (N. GOBALAKRISHNAN, 2018). [23] In comparison to QWO and GA algorithms, the attained result indicates that the proposed method reduces computing time, migration costs, and load utilisation.

A Multi-Objective QoS strategy was provided by Danlami Gabi and et al. (Danlami Gabi, 2017) in 2017 to solve customer expectations utilizing execution time and cost parameters. Cloud Scalable Multi-Objective Cat Swarm Optimization (CSO) based Simulated Annealing (SA) (CSM-CSOSA) techniques were suggested to solve the problem. To improve the exploration capability of the local search, the Taguchi Orthogonal approach is employed to improve the simulated annealing and incorporate the proposed algorithm into it. [24]

[25] A different approach for task scheduling utilizing the Flower Pollination Algorithm (TSFPA) was integrated in 2017 by Jaspnder Kaur and Brahmalen Kaur Sidhu (Sidhu, 2017). The algorithm seeks to shorten the time it takes to complete a task (makespan). When the efficiency of the suggested method (TSFPA) was compared to that of other approaches such as the genetic algorithm (GA), First Come First Serve (FCFS), and Round Robin (RR) scheduling strategy, the suggested method (TSFPA) outperformed the other algorithms in terminology of makespan.

To overcome the scheduling problem, Xing Jia Wei and et al. [26] introduced a Simulated Annealing Multi-Population Genetic Technique (SAMPGA) algorithm in 2017. SA is put into SAMPGA to stop the local optimum and enhance the international optimum efficiency. Simultaneously, to discover a relevant solution and enhance convergence time, a family evolution method based on the adaptive mechanism in MPGA is proposed. The simulated result demonstrates that the SAMPGA performs well in terms of reducing job completion time, cost, and balance load.

In 2017, Ashish Gupta and Ritu Garg (Ashish Gupta, 2017) proposed a meta-heuristic method based on the ant colony optimization algorithm to handle job scheduling in cloud systems, with the primary objectives of minimising makespan/computation time and optimising load balancing [27]. According to the findings, the proposed load balancing ant colony optimization algorithm (LB-ACO) outperforms the NSGA-II method in terms of makespan and load balancing.

[28] Hu Yao et al. (Hu Yao, 2017) suggested a "three-stage selection procedure," and the "total-division-total" genetic approach was devised to improve genetic strategy. The CloudSim tool's findings imply that the improved algorithm outperforms a simple genetic algorithm (SGA) in terms of task completion time. It's a dependable task scheduling approach for cloud computing.

A pre-allocation Ant Colony Optimization (PACO) method for job scheduling in the cloud system was proposed by Ruonan Lin and Qiang Li (Ruonan Lin, 2016). The modified Ant Colony Optimization (ACO) method and the template size are utilized to individual schedule tasks in this algorithm. Within the simulated software, the proposed technique performs well. The experiments indicates that PACO Scheduling productivity improves work performance. [29]

[30] WANG Bei and Li Jun (WANG Bei, 2016) proposed a Multi-Population Genetic Algorithm (MPGA) for load balancing, which is utilized to handle task complexities in cloud systems and avoid premature convergence. Several strategies in this investigation are dependent on which min-min and max-min approaches were utilised to initialise the population. The metropolis condition is then employed to filter the progeny in authority to determine whether or not the defective person will be allowed. The population diversity can thus be retained, and the optimum local can even be avoided. The collected findings demonstrated that the MPGA was capable of obtaining good task scheduling results, such as cost and execution time minimization.

In 2016, Zhifeng Zhong and colleagues (Zhifeng Zhong, 2016) proposed a Greedy Particle Swarm Optimization (G&PSO) method for task scheduling resolution [31]. The findings reveal that the proposed algorithm increased each virtual machine's performance, comprising local and international search capabilities, a faster convergence rate, and a more equally allocated workload. As a result, the Greedy Particle Swarm Optimization beats the traditional PSO in terminology of resource utilization.

2.4 Comparison of Related Works

Table 1.0: Comparison of Related Work

Ref.	Year	Applied Algorithm	Parameters	Finding	Tools
(Zong, 2020)	2020	Hybrid GA and ACA.	Execution time, energy	The proposed solution minimizes the amount of time and total energy consumed by jobs in cloud computing systems.	CloudSim
(Vijayalakshmi A. Lepaksh, 2020)	2020	ERAS Algorithm	Makespan, Reliability	When compared to the EFT method for allocation, the ERAS algorithm improves reliability while delivering superior results.	CloudSim
(Sanaj M S, 2020)	2020	ERR Algorithm	Waiting time, Execution time, and Residue energy.	The overall waiting time for ERR tasks decreased in comparison to RR under the same conditions. In terms of execution time and residue energy, the ERR algorithm outperforms algorithms like ACO, GA, M.P.A., Min-Min, and PSO.	CloudSim
(Yong Shi, 2020)	2020	BMin Algorithm	Completion time, Throughput, Load balancing	When compared to Min-min, the completion time is shorter, and the load balance is better.	CloudSim
(Pratisha Sarma, I. C., Anju Bala, 2019)	2019	VWOA algorithm	makespan, cost, and energy and resource utilization	The suggested algorithm outperformed the WOA and RR algorithms in terms of makespan, cost, degree of imbalance, energy consumption, and resource utilization.	CloudSim
(SHANCHEN PANG, 2019)	2019	EDA-GA algorithm	Convergence, completion time, load balancing	The EDA-GA method improves load balancing by enhancing	CloudSim

				convergence, search capabilities, and job execution time minimization.	
(Shengmei Liu, 2019)	2019	IDPSO algorithm	Completion time, converge	In terms of completion time and convergence, IDPSO outperforms DPSO and FIFO.	CloudSim
(Abdulsalam Alsmady, 2019)	2019	MA algorithm	Makespan, cost	The MA algorithm lowered makespan and beat the GA and PSO algorithms.	CloudSim
(Abdullah Alzaqebah, 2019)	2019	MGWO algorithm	Makespan, cost, degree of imbalance	Traditional Grey (GWO) and (WOA) algorithms perform better in terms of makeup, cost, and degree of imbalance than MGWO.	CloudSim
(Ping Zhu, 2019)	2019	PRTS algorithm	Energy	When compared to the ESS basic algorithm, you can save up to 12.3% on energy.	LIGO
(A.M.Senthil Kumar, 2019)	2019	(GA-PSO) algorithm	response time	The hybrid GA-PSO method outperforms GA, Max-Min, and Min-Min algorithms.	CloudSim
(X. X. Fang Yiqiu, Ge Junwei, 2019	2019	adaptive genetic algorithm (AGA.)	Completion time, load balancing	When compared to the adaptive (AGA) and (SGA) algorithms, the good effect on resource scheduling produces a more reasonable and optimal task scheduling result.	CloudSim
(Negar Chitgar, 2019)	2019	A new method for scheduling workload based on VM grouping	Makespan time, response time, resource utilization	The VM grouping approach minimises the average reaction time and makespan time. Unlike the SJF and	CloudSim

				MinMin algorithms, this approach improves the resource use ratio.	
(Shuang YIN, 2018)	2018	LCGA algorithm	load balancing, cost	The results show that the scheduling algorithm is successful, and that the optimization method is applied when the LCGA algorithm is compared to the LGA and CGA methods.	CloudSim
(L. X. Fang Yiqiu, 2018)	2018	VM-ACO algorithm	task transmission, execution time, and load balancing	In terms of task latency, time to complete the work, and load balance, it outperforms the ant colony algorithm.	CloudSim
(Mehran Ashouraei, 2018)	2018	parallel genetic algorithm	Load balancing, energy usage, migration rate, resource utilization	Improve the load balance level by choosing better tools to execute arrival jobs with a lower task failure rate in a shorter time.	Matlab
(N. GOBALAKRISHNAN, 2018)	2018	(GGWO.) Algorithm	computation time, cost, energy consumption, load utilization	GGWO will improve task scheduling over ordinary GWO and GA.	CloudSim
(Danlami Gabi, 2017)	2017	CSM-CSOSA algorithm	execution time, execution cost, QoS	The results are compared to (MOGA), (MOSACO), and (MOSACO) (MOPSO), with the suggested technique outperforming the others in terms of performance and QoS.	CloudSim
(Sidhu, 2017)	2017	TSFPA algorithm	Makespan	TSFPA performs better than GA, RR,	CloudSim