

# Multi-objective based Cloud Task Scheduling Model with Improved Particle Swarm Optimization

Chaitanya Udatha, Gondi Lakshmeeswari  
Department of Computer Science and Engineering  
GITAM (Deemed to be University)  
Visakhapatnam, India

**Abstract**—Now-a-days, advanced technologies have emerged from the parallel, cluster, client-server, distributed, and grid computing paradigms. Cloud is one of the advanced technology paradigms that deliver services to users on demand by cost per usage over the internet. Nowadays, a number of cloud services have rapidly increased to facilitate the user requirements. The cloud is able to provide anything as a service over web networks from hardware to applications on demand. Due to the complex infrastructure of the cloud, it needs to manage resources efficiently, and constant monitoring is required from time to time. Task scheduling plays an integral role in improving cloud performance by reducing the number of resources used and efficiently allocating tasks to the requested resources. The paper's main idea attempts to assign and schedule the resources efficiently in the cloud environment by using proposed Multi-Objective based Hybrid Initialization of Particle Swarm Optimization (MOHIPSO) strategy by considering both sides of the cloud vendor and user. The proposed algorithm is a novel hybrid approach for initializing particles in PSO instead of random values. This strategy can obtain the minimum total task execution time for the benefit of the cloud user and maximum resource usage for the benefit of the cloud provider. The proposed strategy shows improvement over standard PSO and the other heuristic initialization of PSO approach to reduce the makespan, execution time, waiting time, and virtual machine imbalance parameters are considered for comparison results.

**Keywords**—Cloud computing; task scheduling; cloud service provider; virtual machines; PSO; multi-objective; cloud service broker

## I. INTRODUCTION

The rapid growth of internet data processing prompted the creation of cloud computing systems. Cloud computing is critical for providing technology-based services through the use of the internet. It gives access to computing resources like storage, network and data without requiring active user control. Cloud environments can provide three distinct services: SaaS, PaaS, and IaaS. SaaS (Software as a Service) is the top layer service that distributes software to consumers. SaaS allows users to utilize software straight from the cloud without the need to install anything locally; you can access it immediately from the cloud.

The middle one is PaaS (Platform as a Service), it allows users to develop and deploy their own applications on top of the provided platform. Finally, IaaS (Infrastructure as a service) is the bottom layer service, the capability to deliver

services as servers, storage and operating system and compute resources. Cloud providers use virtualization technology to provide consumers with computational resources virtually. Optimal task scheduling strategy is essential in the multi-tenant cloud computing model for enhancing the performance of a cloud environment. An efficient scheduling strategy enables the best virtual machine (VM) allocation to the required tasks in a way that to attain the required quality of service. The purpose of optimal allocation of tasks to VMs that fits certain criteria to obtain a specific objective as a result, the scheduling algorithm is a vital part of any cloud architecture.

### A. Scheduling in Cloud Environment

Nowadays, everyone is trending towards advanced technology to save management efforts, time, and personnel. Cloud Computing is a new paradigm for hosting services and delivering those through the internet. Cloud dynamically provisions platform, infrastructure, and software applications as services to the cloud users based on the pay-as-you-go model, which means charging per usage. The cloud is a metaphor for the internet and virtualization technology is the key concept used to deliver services through the cloud by maintaining data integrity. Cloud environment is a pay as you go service model and it is an important aspect to business owners to compute huge amounts of data. Scheduling helps in better utilization of resources optimally. Thus, scheduling is the heart of cloud computing for the management of resources effectively.

Scheduling is categorized into two distinct levels:

- 1) The first level of Scheduling under IaaS is the Task or Workflow scheduling
- 2) The next level is the Virtual machine scheduling under IaaS.

Fig. 1 indicates that the primary type of scheduling technique used in cloud computing is further divided into two types: workflow task scheduling and independent task scheduling.

- Workflow task scheduling enables the tasks in a specific order because the tasks are interdependent, like a parent-child relationship.
- Independent task scheduling is converse to workflow scheduling in which all tasks are independent and there is no dependency among the set of tasks.

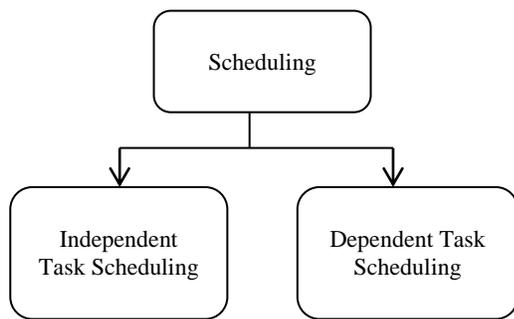


Fig. 1. Task Scheduling Categories.

Scheduling can be achieved with either static or dynamic scheduling approaches. The resources and scheduling strategy is pre-determined in static scheduling. Whereas in dynamic scheduling, the resources are allocated at the time of execution according to requirement and resource allocation can be modified during execution. Each scheduling technique can be achieved through different categories of heuristic, meta-heuristic, and combination of both approaches. Heuristic scheduling techniques are the most common type of scheduling methodology. Static scheduling is done by using heuristic methods, to give a single static solution. Cloud task allocation strategy is a type of NP-hard problem. Meta-heuristic algorithms are required to solve problems that are multidimensional in nature. They provide multiple solutions dynamically. Several researchers proposed multi-stage hybrid meta-heuristic algorithms to obtain better performance, combining heuristic and meta-heuristic approaches.

The following is a proposed work summary of the key contribution of task scheduling to the current literature:

- MOHPSO solution for optimum task scheduling strategy employs a hybrid approach that combines two heuristic methodologies such as shortest job first and minimum execution time, to initialize the particles in PSO with a good starting point to explore the search space more efficiently instead of random values.
- Creating a multi-objective scheduling technique that reduces both task execution and waiting time for the benefit of the user in reducing the cost of application based on pay per usage policy and improving resource utilization, maximizing the profit for cloud provider by reducing makespan and degree of VM imbalance.
- The proposed method was implemented in CloudSim framework by extending JSWan package and validating the proposed method with multi-objective-based standard PSO and SJF-PSO methods.

Section II focuses on the literature review, Section III focuses on the proposed MOHPSO model for scheduling, the outcome of the proposed method is compared and analyzed in Section IV, and the conclusion in Section V.

## II. RELATED WORK AND BACKGROUND

Researchers sought to discover acceptable scheduling mappings in the cloud environment using various methodologies based on heuristic and meta-heuristic

approaches. Many authors have improved scheduling strategy by novel variants in nature-inspired algorithms to enhance the global search capability of traditional standard techniques to avoid premature convergence. However, the focus on scheduling using multi-objective-based algorithms was minimal.

Bangyal, Waqas Haider, et al. [1] survey provides a complete overview of the different PSO and DE initialization procedures based on the Sobol, Halton, and random distribution families of quasi-random sequences. The fundamental purpose of the proposal was applied to various meta-heuristic approaches. It provides future work directions for the researchers.

Alsaidy et al. [2] proposed heuristic initialized PSO [3] [4] [5] outperforms among other approaches by considering of convergence and load balance, but it is a single objective based solution not able to satisfy the targets of multiple objectives by considering both cloud provider and user.

Bangyal et al. [6] proposed an enhanced version of bat algorithm using torus walk instead of uniform walk for improving local search and chaotic mapping [7] introduced for inertia weight to explore more in global search space for hyper dimensional global optimization problems. TW-BA is useful for the researchers to propose a new variant to all traditional nature-inspired algorithms.

Zhou, Zhou, et al. [8] introduced a unique variation of GA using a greedy approach to optimize scheduling strategy, which converges solution with very few iterations. This approach had considered only the makespan as a fitness function.

Ngatman et al. [9] survey on modified PSO compared to traditional PSO to solve issues of random initialization of population for convergence of best solution by exploring the search space effectively. This survey is useful for many authors to propose the advancement of PSO by considering the study.

Zhang et al. [10], [11] survey provided a thorough examination of PSO. PSO advancements by initializing with chaotic and quantum behavior, analyzed PSO with different population topologies, hybridization and extensions by discussing multiple objectives [12] and theoretical analysis of PSO were considered in various computing environments for targeting researchers from all engineering fields.

MOPSO [12] [13] [15] based new task scheduling model by E. S. Alkayal et al. [14] on a ranking strategy to achieve minimum waiting time and maximum throughput for only heterogeneous tasks but there is a chance of performance degradation for homogeneous tasks.

The proposed scheduling technique makes three main assumptions:

- The first assumption is that each task is an independent task.
- The second assumption is that users can submit  $n$  tasks, and are executed on  $m$  virtual machines, mapping tasks to VMs.

- The final one is that there is no task migration across virtual machines. That is a task cannot be assigned to multiple virtual machines at the same time.

A major challenging issues of scheduling is to schedule, distribute varying number of tasks to multiple virtual machines (VMs) and minimizing the turnaround time of a task. In the cloud environment which consists of number of data centers.

Each datacenter (DC) consists of hosts ,  $i = 1, 2, \dots, N$  are presented and it can be represented as:

$$DC = [H1, H2, \dots, Hi \dots, HN ]$$

Host(Hi) consists of VMs ,  $i = 1, 2, \dots, N$  are presented in each host and it can be represented as:

$$Hi = [VM1, VM2, \dots, VMi, \dots, VMN ]$$

Similarly each VMi consists of tasks,  $i = 1, 2, \dots, N$  are executed on each virtual machine based on task scheduling algorithm as shown in Fig. 2.

$$VMi = [T1, T2, \dots, Ti, \dots, TN ]$$

Each cloud user submitted job is considered as a task. The cloud broker acts on behalf of the user in a data center environment and abstracts VM management functions like VM creation, cloudlet assignment to these VMs, and VM destruction. CIS(Cloud Information Service) is one of the cloud entities which performs cloud resource registry and indexing. The datacenter informs CIS that they are ready to process the cloudlets. The cloud broker can communicate with this entity, which returns a list of all the VM IDs that have been registered and allocated to the tasks.

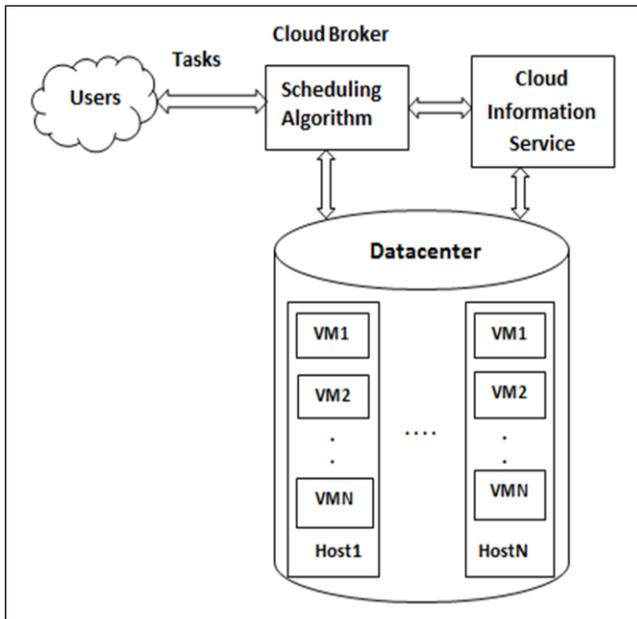


Fig. 2. Cloud Task Scheduling Model.

### III. PROBLEM DESCRIPTION FOR PROPOSED METHODOLOGY

In this section, the primary discussion is about the standard PSO method and MOHIPS0 algorithm.

TABLE I. MOHIPS0 PARAMETERS

PSO Parameter	Value
Population	50
Iterations	100
$\alpha$	0.05
$\beta$	0.7
$\gamma$	0.25
$\omega$	0.9
$k1, k2$	2.0
$r1, r2$	[0,1]

Table I lists the PSO parameters considered for implementing MOHIPS0 model. The PSO algorithm is a meta-heuristic population-based strategy for finding food sources in an optimal way influenced by the social behavior of birds flocking. PSO quickly gained popularity as a general purpose global optimizer. In this method, particles are moved across a multidimensional solution search space to find their destination. Each particle position changes in response to its own experiences as well as the experiences of others around it based on particle fitness value denoted by  $fit(X)$ . The symbol  $X(p)_i^{(t)}$  represents the location of particle position  $i$  at iteration  $t$ . It is possible to change the location of particle position  $X(p)_i^{(t)}$  by adding the new random velocity  $U(p)_i^{(t+1)}$  to the current position, as shown in the below equation:

$$X(p)_i^{(t+1)} = X(p)_i^{(t)} + U(p)_i^{(t+1)} \quad (1)$$

The particle velocity reflects the movement of particle position that is socially exchanged is given by the equation:

$$U(p)_i^{(t+1)} = \omega * U(p)_i^{(t)} + k1r1 * (X(p)_{pbesti}^{(t)} - X(p)_i^{(t)}) + k2r2 * (X(p)_{gbesti}^{(t)} - X(p)_i^{(t)}) \quad (2)$$

In (2),  $k1$  and  $k2$  are used as constant factors of a particle for personal and global influence, respectively. In this equation,  $\omega$  denotes the inertia weight of a particle, which is used to control the movement of the particle velocity.

$$X(p)_{pbesti}^{(t+1)} = \begin{cases} X(p)_i^{(t+1)}, & \text{if } f(X(p)_i^{(t+1)}) < f(X(p)_{pbesti}^{(t)}) \\ X(p)_{pbesti}^{(t)} & \text{otherwise} \end{cases} \quad (3)$$

$X(p)_i^{(t)}$  denotes local best position of the particle,  $X(p)_{gbesti}^{(t)}$  denotes the particle best position of the entire swarm globally and  $r1$  &  $r2$  denotes pseudo random values within the range between 0 and 1 at each iteration  $i$ .

### B. Proposed MOHIPS0 Model

One of the criteria used to classify the meta-heuristics algorithms for optimization problems by considering the no. of objectives, they are single objective, multi-objective [12] [13] and many objectives [11].

The proposed Multi-Objective based Hybrid Initialization of Particle Swarm Optimization (MOHIPS0) algorithm determines which virtual machines are most ideal for scheduling tasks and finds the most efficient task scheduling

schema. Total n tasks  $t_1, t_2, \dots, t_n$  are defined in the task scheduling model, and they must be allocated to m virtual machines (vm1, vm2,...,vmm) in order for them to be executed.

**Algorithm 1: Pseudocode for MOHIPSO**

**Input:** Tasks, Task length, VMs ,VM Processing rate  
**Output:** Task scheduled on VM  
**Start:**  
 Initialize particles Xi with SJFP-MET algorithm  
 For each iteration i  
 For each particle Xi  
 calculate TET , MS and DI applying Eqs. 3, 4 and 5.  
 calculate the fitness function fit (X) applying Eq. 6.  
 Find  $X(p)_{pbesti}$  and update the velocity value by 2  
 Update the position of the particle according to Eq. 1.  
 End  
 Find  $X(p)_{pbesti}$   
 End  
 Output the optimal task scheduled with  $X(p)_{gbesti}$

In the proposed MOHIPSO, particles are initialized with a good starting point with the help of hybrid strategy of using minimum execution time (MET) [16] and shortest job to fastest processor (SJFP) [2] to explore the search space effectively instead of random selection. In each iteration multi-objective fitness value is used to find the particle local and global best values by updating velocity randomly and tasks are scheduled on VMs based on  $X(p)_{pbesti}^{(t)}$ .

**C. Task Scheduling Problem Description**

The scheduler determines which task should be assigned to which machine.

Cloud Task Scheduling helps in:

- Reducing operational cost.
- Reducing waiting time.
- Increasing resource utilization.

Table II lists cloud simulation parameters considered for the proposed task scheduling strategy implemented on the CloudSim framework. The simulation initially starts by initializing the CloudSim clock instance and creating a data center and datacenter broker. VM and cloudlet specifications as per Table II were created and submitted to the cloud broker. MOHIPSO model is used to schedule the tasks to specific VMs based on resource availability.

The proposed work improves the PSO algorithm for scheduling tasks in a cloud environment with a multi-objective decision problem. This context mainly considers the three objectives: execution time, makespan, and degree of imbalance. It is expressed as follows:

Execution Time (ET): The time required for processing a task on a particular virtual machine.

$$ET_{ij} = \frac{T_{Leni}}{VM_j}$$

TABLE II. CLOUD SIMULATION PARAMETERS

Cloud Parameter	Value
No.of Tasks/ Cloudlets	10-50
Task Length (MI)	1100-2000
Cloudlet file size	300
Cloudlet output size	300
No.of VMs	5
VM Processing rate (MIPS)	500-900
VM RAM (MB)	512
VM Bandwidth (Mbps)	1000
Vmm Name	Xen
VM Pes Number	1
Data Center	1
Host	1

$ET_i$  denotes Task i execution time on  $VM_j$  and  $T_{Leni}$  denotes ith task length specified in MI (Million Instructions).

Total Execution Time (TET): It is the summation of all tasks processing time.

$$TET = \sum ET_{ij} \tag{3}$$

Makespan: It is the last task finishing time on virtual machine.

$$MS = \text{Max} \{ ET_{ij} \} \tag{4}$$

MS is the maximum make span of all VMs.

Degree of Imbalance (DI): The difference between the maximum and minimum execution time and the total execution time.

$$DI = \frac{\text{Max}\{ ET_{ij} \} - \text{Min}\{ ET_{ij} \}}{TET} \tag{5}$$

The fitness function in (6) is calculated based on TET, MS and DI using weighted sum method for MOHIPSO method as follows:

$$\text{fit}(X) = \text{Min}\{ \alpha * TET + \beta * MS + \gamma * DI \} \tag{6}$$

Where parameter  $\alpha$  refers to the weight of total execution time,  $\beta$  refers to the weight of makespan and  $\gamma$  refers to the weight of degree imbalance.  $\text{fit}(X)$  function is considering three parameters specified in the equations (3),(4) &(5) and three control parameters such as  $\alpha, \beta$  and  $\gamma$  with in the range of [0,1] and sum of  $\alpha+\beta+\gamma=1$  and these parameters values are specified in Table I.

**IV. RESULTS AND COMPARISON**

MOHIPSO algorithm is compared with two other variants of PSO algorithm. The first variant of PSO is a multi-objective based PSO algorithm by considering MS, TET and DI objectives for finding the fitness value of a particle and the second variant of PSO is called as SJF-PSO, in which initialization of particles with SJFP algorithm and multi-objective based fitness value is calculated to map the tasks to VMs. Finally proposed MOHIPSO is a combination of Standard PSO with MET and SJF-PSO gives better performance compared with existing algorithms.

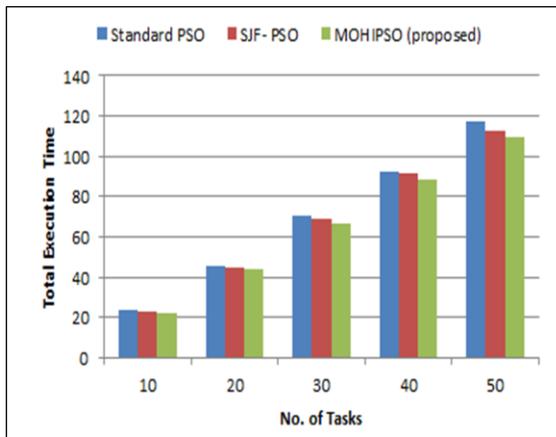


Fig. 3. Comparison of Total Execution time between PSO Variants.

Fig. 3 indicates the number of tasks vs. total execution time, in which the proposed algorithm MOHIPSO shortens the execution time compared other standard PSO and variant of PSO.

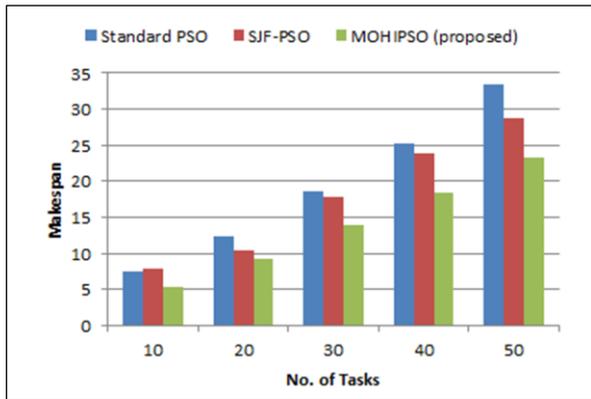


Fig. 4. Comparison of Makespan between PSO Variants.

In Fig. 4, the comparison between number of tasks and makespan, the proposed algorithm reduces the makespan compared to the standard PSO and SJF-PSO

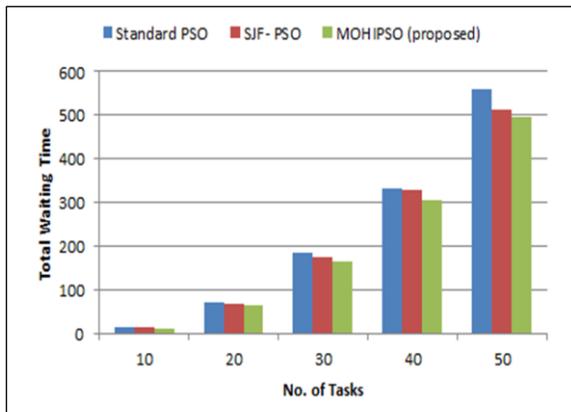


Fig. 5. Comparison of Total Waiting Time between PSO Variants.

Fig. 5 indicates the number of tasks vs. total waiting time of all tasks. The proposed algorithm reduces the waiting time compared to the standard PSO and SJF-PSO.

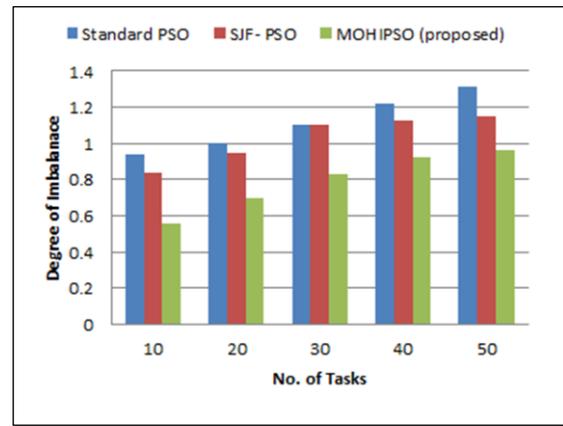


Fig. 6. Comparison of Degree of VM Imbalance between PSO Variants.

Fig. 6 shows the graph between tasks and degree of VM imbalance, it clearly shows that as the number of tasks increases then DI is increased. In order to reduce this, MOHIPSO algorithm considers the best way of scheduling tasks to the given resources on efficient way.

## V. CONCLUSION AND FUTURE WORK

Scheduling plays a crucial role in the cloud environment for effective distribution of tasks to enhance the quality of service. The MOHIPSO satisfies multiple objectives and provides obvious improvements in terms of makespan, execution time, waiting time, and degree of VM imbalance compared to traditional PSO and SJF-PSO. The simulation results show that the MOHIPSO has improved.

Task scheduling is not a multi-objective solution but it is to be a many-objective based solution by considering multiple objectives on both sides of the cloud provider and user. In future, the proposed algorithm can be extended to appraise other quality parameters like energy consumption and cost apart from makespan, execution time, waiting time, and degree of imbalance.

## REFERENCES

- [1] Bangyal, Waqas Haider, et al. "Comparative Analysis of Low Discrepancy Sequence-Based Initialization Approaches Using Population-Based Algorithms for Solving the Global Optimization Problems." *Applied Sciences* 11.16 (2021): 7591.
- [2] Alsaidy, Seema A., Amenah D. Abbood, and Mouayad A. Sahib. "Heuristic initialization of PSO task scheduling algorithm in cloud computing." *Journal of King Saud University Computer and Information Sciences* (2020).
- [3] Madni, Syed Hamid Hussain, et al. "Performance comparison of heuristic algorithms for task scheduling in IaaS cloud computing environment." *PloS one* 12.5 (2017): e0176321.
- [4] Al-Qerem, Ahmad, and Ala Hamarsheh. "Statistical-Based Heuristic for Tasks Scheduling in Cloud Computing Environment." *International Journal of Communication Networks and Information Security* 10.2 (2018): 358-365.
- [5] Mansouri, Najme, Behnam Mohammad Hasani Zade, and Mohammad Masoud Javidi. "Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory." *Computers & Industrial Engineering* 130 (2019): 597-633.
- [6] Bangyal, Waqas Haider, Jamil Ahmed, and Hafiz Tayyab Rauf. "A modified bat algorithm with torus walk for solving global optimisation problems." *International Journal of Bio-Inspired Computation* 15.1 (2020): 1-13.

- [7] Abdullahi, Mohammed, et al. "An efficient symbiotic organisms search algorithm with chaotic optimization strategy for multi-objective task scheduling problems in cloud computing environment." *Journal of Network and Computer Applications* 133 (2019): 60-74.
- [8] Zhou Zhou, Fangmin Li, Huaxi Zhu<sup>3</sup>, Houliang Xie, Jemal H. Abawajy, and Morshed U.Chowdhury "An improved genetic algorithm using greedy strategy toward task scheduling optimization in cloud environments." *Neural Computing and Applications* 32.6 (2020): 1531-1541.
- [9] Ngatman, Mohd Farhan, Johan Mohd Sharif, and Md Asri Ngadi. "A study on modified PSO algorithm in cloud computing." *2017 6th ICT international student project conference(ICT-ISPC)*. IEEE, 2017.
- [10] Zhang, Yudong, Shuihua Wang, and Genlin Ji. "A comprehensive survey on particle swarm optimization algorithm and its applications." *Mathematical problems in engineering* 2015 (2015).
- [11] Geng, Shaojin, et al. "Many-objective cloud task scheduling." *IEEE Access* 8 (2020): 79079-79088
- [12] Gobalakrishnan, N., and C. Arun. "A new multi-objective optimal programming model for task scheduling using genetic gray wolf optimization in cloud computing." *The Computer Journal* 61.10 (2018): 1523-1536.
- [13] Langhnoja, Himani K., and Hetal A. Joshiyara. "Multi-Objective Based Integrated Task Scheduling In Cloud Computing." *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*. IEEE, 2019
- [14] Alkayal, Entisar S., Nicholas R. Jennings, and Maysoon F. Abulkhair. "Efficient task scheduling multi-objective particle swarm optimization in cloud computing." *2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops)*. IEEE, 2016.
- [15] Alguliyev, Rasim M., Yadigar N. Imamverdiyev and Fargana Jabbar Abdullayeva."PSO-based load balancing method in cloud computing." *Automatic Control and Computer Sciences* 53.1 (2019): 45-55.
- [16] Almazok, Salem A., and Bülent Bilgehan. "A novel dynamic source routing (DSR) protocol based on minimum execution time scheduling and moth flame optimization (MET-MFO)." *EURASIP Journal on Wireless Communications and Networking* 2020.1 (2020): 1-26.