

Task scheduling in cloud computing based on meta-heuristic techniques: A review paper

Rasha A. Al-Arasi^{1,*}, Anwar Saif²

¹Sana'a University, Department of Computer Science, Sana'a, Yemen

²Sana'a University, Department of Information Systems, Sana'a, Yemen

Abstract

Cloud computing delivers computing resources like software and hardware as a service to the users through a network. Due to the scale of the modern datacentres and their dynamic resources provisioning nature, we need efficient scheduling techniques to manage these resources. The main objective of scheduling is to assign tasks to adequate resources in order to achieve one or more optimization criteria. Scheduling is a challenging issue in the cloud environment, therefore many researchers have attempted to explore an optimal solution for task scheduling in the cloud environment. They have shown that traditional scheduling is not efficient in solving this problem and produce an optimal solution with polynomial time in the cloud environment. However, they introduced sub-optimal solutions within a short period of time. Meta-heuristic techniques have provided near-optimal or optimal solutions within an acceptable time for such problems. In this work, we have introduced the major concepts of resource scheduling and provided a comparative analysis of many task scheduling techniques based on different optimization criteria.

Keywords: Cloud computing, Resource scheduling, Optimization criteria, Scheduling, Meta-heuristic techniques, Task scheduling.

Received on 10 September 2019, accepted on 16 January 2020, published on 30 January 2020

Copyright © 2020 Rasha A. Al-Arasi *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eai.13-7-2018.162829

*Corresponding author. rasha.ali66@gmail.com

1. Introduction

Cloud Computing (CC) is the latest technology with a fast outgrowth in the field of distributed computing. It confers the users with high reliability, security, scalability, cost-effective mechanism, group collaboration and ease of access to various applications and resources [1]. It is a model for enabling appropriate, on-demand provisioning of computing resources such as software, hardware, applications, and services that can be fast provisioned and freed with least management overhead or interaction from service providers [2]. Cloud computing offers three primary types of service models namely Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) [3]. Cloud computing can be implemented as a layered architecture and comes in four

main development models namely public, private, community, and hybrid clouds [4].

The major concept used in cloud computing is virtualization. Virtualization is a technique by which the user can easily access the computing resources without considering the complexity and internal details of the system [5]. It enables the user to create Virtual Machines (VMs) on physical servers [6], which leads to reducing the required hardware equipment and improving physical resources utilization in cloud computing. There are several advantages provided by clouds to cloud users and the service providers, the major advantages of cloud computing are described in [5, 7-10] and listed below:

- Reducing the cost by providing computing resources on-demand based on a pay-as-you-go system.

- Allocating and deallocating of resources efficiently to keep the capital-expenditure from wasting the unused resources.
- Providing high flexibility, considerable speed in scaling the infrastructure up or down.
- Providing an effective recovery and back up by distributing the backups among data centers in various locations.
- Allowing easy accessibility from anywhere in the world and at any time.

In spite of the significant advantages of cloud computing environments, some significant issues have influenced the efficiency and reliability of this environment [11]. Cloud computing faces many issues that have attracted researchers' attention and concern. In general, the major issues in the environments of cloud have been categorized into seven main categories: resource management, load balancing, privacy and security, migration to clouds, availability and scalability, energy-efficiency, interoperability and compatibility [12-15]. Scheduling in the cloud computing environment means many tasks can be executed on the available pool of computing resources in an optimal way. This operation depends on many optimization criteria such as reliability, makespan, load balancing, execution cost, budget, utilization [16]. In the procedure of scheduling tasks, tasks are delivered from users to cloud scheduler, then the cloud scheduler explores the status of the resources from cloud information service. Later, mapping the tasks on various resources based on their requirements [17]. The efficient scheduler assigns the appropriate resources (e.g. VMs) to the tasks in an optimal way.

Generally, the operation of allocating tasks on apparently unbounded computing resources in the cloud computing environment is a nondeterministic polynomial time (NP)-hard problem. Many researchers attempted to explore an optimal solution with polynomial-time for the task scheduling in the cloud environment. There is no specific technique that has introduced an optimal solution with polynomial-time for this problem. Thus, the techniques based on meta-heuristics have been used to deal with these complex problems to obtain near-optimal or optimal solutions. In the past years, many meta-heuristics techniques have been introduced and gained considerable popularity such as genetic algorithms (GA), particle swarm optimization (PSO), ant colony optimization (ACO), tabu search (TS), simulated annealing (SA), bat algorithm (BA), memetic algorithm (MA) [18].

In order to develop an effective scheduling algorithm, we need to clearly understand resource management and various problems associated with different scheduling techniques. Thus, the objective of this paper is to present

the major concepts of resource scheduling and provide a comparative analysis of various task scheduling techniques. Systematic analysis of task scheduling in cloud computing is presented based on optimization criteria suitable for cloud computing environments. This paper will help researchers to identify the suitable approach for suggesting adequate technique for scheduling user's applications in cloud environment.

In this work, we consider only the scheduling problems with regards to cloud computing and not the whole distributed systems.

The remainder of the paper is organized in five sections. We present resource management in cloud computing in section 2. Scheduling is discussed in section 3. The discussion is presented in section 4. Finally, a conclusion and future work remarks are summarized in section 5.

2. Resource Management in Cloud Computing

Resource management is an important challenge in distributed computing such as cloud computing [19]. In cloud computing, various cloud users require different services depending on their changing needs. So, the task of cloud computing is to introduce all the required services. However, due to the limitation of available resources, it is difficult for cloud service providers to provide all the required services in a timely manner. Because cloud computing relies on virtualization technology with a distributed model, it becomes easy to introduce dynamically new resources which was difficult in the traditional resource management techniques [20]. In the next section, we present the type of resources and address the challenges of resource management in the cloud computing environment.

2.1. Type of Resources

In the following section we briefly introduce the classifications of the main types of resources based on their services such as storage, computation, network, security, and energy. Table 1 summarizes the types of resources in cloud computing.

1. Storage services: computer systems are likely to fail over time. So, the company or individuals need continuity to save and manage backups. Storage as a Service (StaaS) is basically a system that permits you to save data in the cloud. It includes thousands of database servers, and hard drives [21]. StaaS provides long-term retention for data, reduces cost in physical space and hardware, and reduces risks for disaster recovery. Thus, StaaS, enhances availability and work continuity [22].
2. Computation services: computation as a Service (CaaS) is rapid computational service in the cloud computing environment. It includes the

- power of processing, the capacity of memory, efficient algorithms, operating system [22].
3. Network services: Network as a Service (NaaS) consists of physical resources such as physical network links, sensors, workstations and intermediate devices, and logical resources such as protocols, throughput, bandwidth, delay, loads and virtual network links [23]. Storage services and computation services cannot be thought of without network services such as bandwidth and delay. These services are the most significant services from network point of view, because every service in cloud computing is provided through high speed Internet [24].
 4. Security services: security as a service (SECaaS) is one of the important challenges in cloud computing environment [25]. SECaaS introduces high protection for users from attacks and threats over the internet [26]. It introduces services such as authentication, trust, intrusion detection, penetration testing, anti-malware, anti-virus, and security event management [27].
 5. Energy services: Energy consumption in the cloud data centers is very high. Energy service consists of physical resources such as cooling devices and uninterruptable power supplies (UPS). Many energy saving techniques have been introduced to manage unused resources to reduce the cost. Significant energy can be saved in data centers by applying energy saving techniques on servers and networks [28].

2.2. Challenges of resource management in cloud

The important challenges that are commonly associated with resource management in cloud systems are resource allocation, resource provisioning, resource mapping, resource discovery and selection, resource adaptation, resource brokering, and resource scheduling. We will briefly introduce the basic concept of these challenges:

Resource allocation: is the economical distribution of cloud resources among different applications through the internet [29].

Resource provisioning: is the process of allocating the service provider's resources to the cloud users with the service quality assurance which is determined in the service level agreement (SLA). It can be classified into two types: dynamic and static resource provisioning [30].

Resource mapping: is a consistency between resources available with a service provider and resources required by cloud users [31].

Resource discovery and selection: is the process of discovering all resources presented in the system and collecting the current state of resources then making decisions which target resource should be selected based on the information obtained from the discovery [32].

Resource adaptation: is the capability of this system to dynamically adjust resources to meet user requirements [26].

Resource brokering: is the process of negotiation for the required resources through an agent to guarantee that necessary resources are available in time to complete the objectives [20].

Resource scheduling: defined by [26], as a timetable of events and resources that records when an activity should start or end, depending on its (1) duration, (2) predecessor activities, (3) predecessor relationships, and (4) resources allocated.

Table 1. Type of Resources

Storage	Hard drives Database
Computation	Memory. Processing. Algorithms. Operating System.
Network	Physical: Network link. Workstations. Sensors. Intermediate devices Logical: Virtual network link. Protocols. Throughput. Bandwidth. Delay. Loads.
Security	Authentication. Trust. Privacy. Anti-malware. Anti-virus. Intrusion detection. Penetration testing.
Energy	Cooling devices UPS. Energy saving technique.

3. Scheduling

The main objective of scheduling is the optimal allocation of resources to specific tasks in a limited time to achieve a high-performance computing and desirable quality of service.

The scheduling must schedule the given tasks to available resources subject to certain constraints to improve one or more optimization criteria [33]. In distributed computing systems, scheduling is responsible for selecting the appropriate resources for task execution taking into consideration some dynamic and static task's parameters [34]. Scheduling algorithms differ by the nature of tasks in the application. When a task has a sequence, the task can be scheduled only after all of its

main tasks have been completed, this is called workflow scheduling. In another case, when tasks are independent of each other, they can be scheduled in any order and known as independent task scheduling [35].

3.1. Scheduling Procedure

Scheduling procedure in cloud computing can be classified into three phases: resource discovering and monitoring, resource selection, and task submission [36]. Figure 1 clearly depicts the three stages:

- Resource discovering and monitoring: In the first phase, Datacenter Broker (DB) detects all resources presented in the cloud system and gathers the current state of all available resources and all remaining resources that may be available in the cloud system. Indeed, these resources are generally the virtual resources.
- Resource selection: In the second phase, the cloud scheduler makes decision which target resource should be selected based on the information obtained from the discovery phase.
- Task submission: In the last phase, the task is assigned to the best available selected resource.

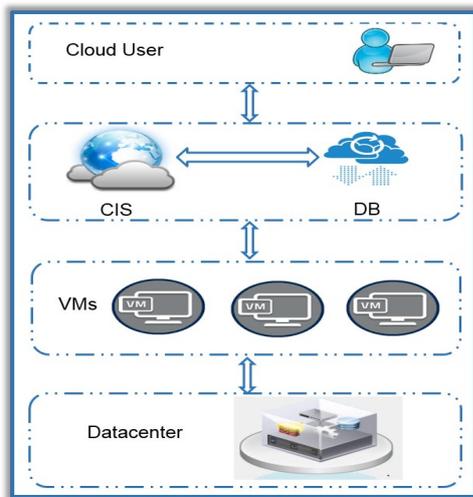


Figure 1. Procedure of scheduling in cloud

Datacenter Broker resides between the cloud service provider and cloud user. It collects all the information about all resources presented in the cloud system and their current status [37]. A datacenter contains a large number of hosts and related equipment. Each host can have multiple VMs based on its hardware specifications (CPU, RAM, bandwidth) [38]. This information needs to be stored in a depository for future use by DB. Thus, Cloud Information Services (CIS) is used as a depository to store the cloud datacenter entities. Once a data center is created, it has to be registered on a CIS. When a cloud user requests a service, he/she sends tasks to the datacenter

broker which gathers the information about the available resources from the CIS then it allocates the tasks to each virtual machine based on the scheduling policy which is defined in the datacenter broker.

3.2. Cloud Resource Scheduling layers

Cloud computing contains three primary types of service models namely IaaS, PaaS, and SaaS, which can be implemented as a layered cloud computing architecture. Based on this architecture, the cloud scheduling problems can be classified into three scheduling layers: Infrastructure layer, platform layer and software layer[39].

3.2.1. Scheduling in Software Layer

Scheduling at the software layer answers the question, how to allocate resources for tasks to satisfy user's objectives such as task completion time (makespan), reliability, costs and application performance. Also, to meet with the desire of cloud service provider to schedule available cloud resources effectively to save the energy consumed by the data center [40].

3.2.2. Scheduling in Platform Layer

The physical resources like network resources, storage resources, and computational resources[41] are better virtualized as uniform resources. Therefore, scheduling these resources in the platform layer concentrate on how to migrate and map virtual resource into a physical resource with an effective load balance and cost.

3.2.3. Scheduling in Infrastructure layer

Cloud service providers need to build a number of cloud data centers around the world to introduce services to the cloud users. So, service resources must be deployed efficiently at different locations of cloud data centers around the world. In addition, other cloud providers can be connected with each other to compose a cloud federation that provides more efficient cloud services. Thus, when scheduling in the infrastructure layer, issues such as scheduling data routing and cloud federation should be addressed and resolved [42]. Cloud federations interconnect the cloud computing environments of two or more service providers to participate together and deliver their services to the cloud users as a single service. In cloud computing environment, data routing is to find cloud resources fast in a multi-cloud environment [43].

3.3. Task-Resource Scheduling Problem Formulation

In cloud computing, task scheduling optimization should define the optimal number of required systems so that the total cost is minimized. Assuming that there are n tasks their execution time on each processing machine is known and they should be processed on m available computational resources. The goal is to maximize the

utilization of the available resources and minimize the total execution time. Assume that the number of tasks is more than the number of available resources ($n > m$), and tasks are not allowed to migrate between resources [44]. To formulate the problem, consider the set of tasks defined as $T_i = \{1, 2, \dots, n\}$ where n is the number of independent tasks and $R_j = \{1, 2, \dots, m\}$ where m is the number of computational resources. Therefore, cloud resource scheduling problem is to get an optimal mapping (OM) of tasks (T_i) to resources (R_j) OM: $T_i \rightarrow R_j$. The definition of this problem is depicted in Figure 2, where, two or more tasks may share one resource [45].

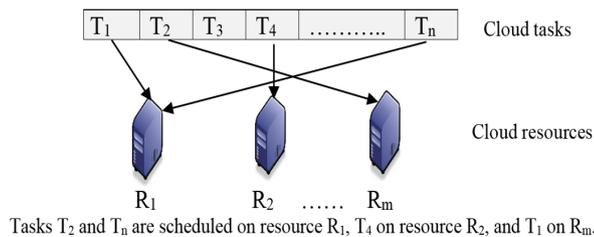


Figure 2. Cloud Resource Scheduling Problem

3.4. Optimization Criteria

This section explains the parameters used to measure the effectiveness of scheduling. The existing works have addressed different kinds of optimization criteria such as makespan, cost, budget, deadline, resource utilization, throughput, load balancing, and energy efficiency. Generally, these optimization criteria are categorized into two desires based on cloud service: cloud users desire and cloud service providers desire, figure 3 [47]. These optimization criteria are addressed from most of the reviewed works, thus this work tries to demonstrate the way these criteria are studied in a comparative method.

3.4.1. User Desire Criteria

- **Makespan (completion time):** Makespan is defined as the completion time of the last task that is required to complete and leave the cloud system [48].
- **Cost:** cost is the total amount the user pays to a service provider on the basis of their resource usage [49].
- **Budget:** it indicates the constraints on completing the tasks within the budget [50].
- **Deadline:** it represents the termination of running tasks at a certain time [51].

3.4.2. Provider Desire Criteria

- **Resource utilization:** making the most of the available resources and keep resources as busy as possible. It is useful for service providers to get gain by leasing the finite resources to the cloud user on-demand [52].
- **Throughput:** it measures the number of completed tasks per unit time [53].

- **Load balancing:** load balancing in cloud computing is the distributions of loads evenly between the VMs over physical resources. Many techniques have been introduced by the authors in [54-56].
- **Energy efficiency:** energy efficiency can be defined as a reduction of energy consumed by a task [57].

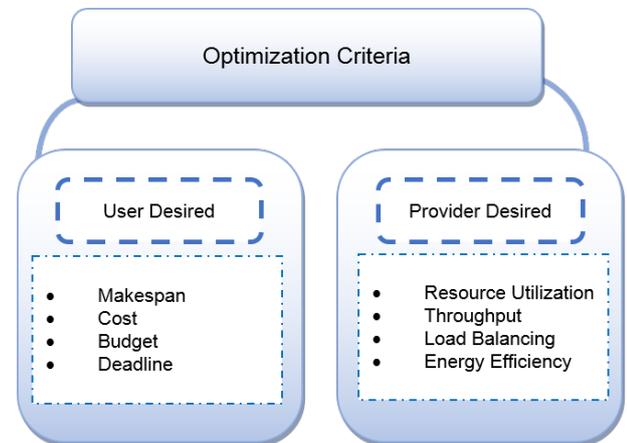


Figure 3. Optimization Criteria

Optimization problems can be divided into discrete and continuous problems. The decision variables for a combinatorial problem have discrete values; while the decision variables for a continuous optimization problem can take up values within the domain of real values (R_i) [58, 59]. According to the number of criteria involved in the optimization problem, this can be divided into single-criterion and multicriteria. The task of single-criterion optimization is to find the optimal solution according to only one criterion function. When the optimization problem involves more than one criteria function, the task is to find one or more optimal solutions regarding each criterion. Here, a solution which is good with respect to one criterion can be worse for another, and vice versa [60]. Therefore, the goal of multi-criteria optimization is to find a set of solutions that are optimal with respect to all other criteria. Noticeable, most real-world problems are multi-criteria. Nowadays, there exist optimization techniques that search for solutions by using Meta-heuristic and heuristic based search techniques. Stochastic and deterministic search principles are applied in these techniques. If an algorithm successfully solves all instances of problem (P), then we can say that it is capable of solving that problem. Usually, we are interested in which technique solves the problem more efficiently. Normally, the term efficiency is connected with the resources of the computer (space and time) that are occupied by running a technique [61, 62]. Generally, the most efficient technique is the one that finds the solution to the problem in the fastest way. In practice, the time complexity of an algorithm is not measured by the effective time necessary for solving the problem on a concrete computer because this measurement suffers from a lack of criteria. The same algorithm could be run on

different operating systems or even on different hardware configurations. Therefore, the algorithm's complexity is measured in an informal way that determines the complexity with regard to the amount of input data, necessary for the problem description. The time complexity of an algorithm determines the way in which the increase in the instance size influences the time complexity. This relation can be expressed with the so-called asymptotic time complexity function $O(f(n))$ that determines the upper bound of time complexity for problem P. For example, the function $O(n^2)$ denotes that the increase in the instance size n will cause an increase in the time complexity to almost n^2 . The algorithmic theory divides problems, with regard to the asymptotic time complexity function, into two classes: NP-hard and P-hard. In the first class, problems that demonstrate the exponential time complexity $O(2^n)$ and are, therefore, treated as "complicated." That is, the exponential time complexity may cause that some increase in the input data can increase solution time of the problem exponentially. In the worst case, we could be waiting for the solution over an infinite period of time. In other hands, problems of class P-hard have polynomial time complexity $O(n^k)$ and are treated as "simple." [62].

3.5. Task Scheduling Techniques

There are several types of scheduling techniques for distributed computing systems. Figure 4 depicts the three main types of scheduling techniques namely Meta-heuristic techniques, traditional techniques, and heuristic techniques. The Meta-heuristic techniques are classified into two categories: Swarm Intelligence(SI) and Bio-Inspired(BI) [62].

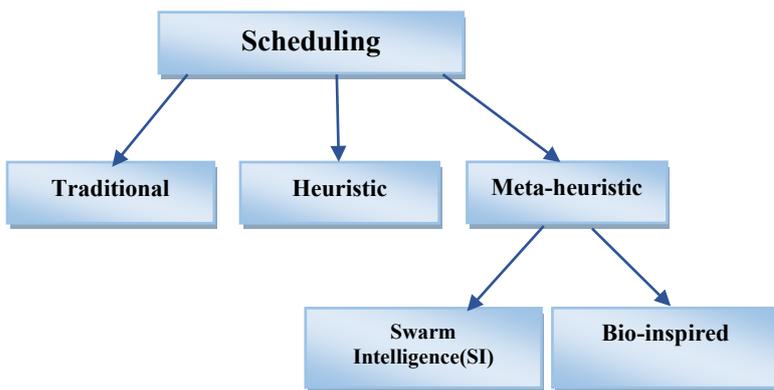


Figure 4. Task scheduling techniques

3.5.1. Traditional Techniques:

Traditional techniques are the essential techniques for scheduling different tasks such as: Shortest Job First (SJF), First Come First Serve (FCFS) and Round Robin (RR) [63]. These techniques are simple, fast, deterministic and obtain exact solutions[64]. But, they are not efficient to understand the optimality problem in many situations [65]. So, traditional techniques are not feasible in cloud

environment scheduling [66]. Many works have been carried out to improve the implementation of the traditional techniques [63, 67-70]. Round robin is one of these techniques that work using time slice or a quantum. The RR algorithm has a drawback that it utilizes static time quantum [67]. The proposed CPU scheduling in [68] relies on the round-robin scheduling, but changes the way of the scheduling calculations. It decreases the waiting time and turnaround time radically contrasted with the straightforward RR scheduling, rather than giving static time quantum in the CPU scheduling. FCFS algorithm means that, task that comes first will be executed first. Researchers in [69] proposed an algorithm for task scheduling based on fuzzy clustering algorithms to increase the resource utilization and minimize the task execution time. SJF is a scheduling technique that depends on the execution time of the task. The tasks are queued based on priority, the longest time is placed last with the lowest priority and the smallest time is placed first with the highest priority [71]. In this algorithm CPU is assigned to the task with least burst time. Elmougy et al. in[70] proposed a hybrid algorithm of RR and SJF called SRDQ algorithm. This algorithm considers a dynamic variable task quantum time.

3.5.2. Heuristic Techniques:

These techniques are using sample space of random solutions to find the optimal or near optimal solution [66]. Many heuristic techniques exists such as min-min, priority-based min-min, enhanced max-min, and max-min.[72]. These techniques give better results as compared to the traditional techniques, but do not guarantee to score high ranking in the cloud scheduling [73].

The solutions resulting from heuristic techniques often get stuck in the problem of local minima [66]. An improved Max-min technique using the expected execution time for selection basis instead of completion time is proposed in [74]. It allocates a task with average execution time. The algorithm increases the chance of synchronous assignment of tasks on resources. The basic Min-Min algorithm is a straightforward and effective algorithm that generates the best scheduling in terms of reducing task completion time. However, the biggest drawback is load balancing, which is considered to be one of the major challenges for cloud service providers. Authors in [75] have improved load balancing by proposing Load Balance Improved Min-Min(LBIMM) algorithm. LBIMM algorithm is designed based on the Min-Min algorithm in order to increase the resource utilization and decrease the completion time.

3.5.3. Meta-heuristic Techniques:

The problem of allocating tasks on resources in cloud computing environment is NP-Hard problem. Therefore, task scheduling is clarified by using meta-heuristic and heuristic to obtain near-optimal or optimal solutions. Heuristic techniques are subsets of meta-heuristic techniques. Meta-heuristic techniques are often nature-

inspired by social behavior of insects [76]. The meta-heuristic word is stated by Fred Glover in 1986, the prefix “meta” means higher level and “heuristic” means to discover by trial and error. We adopted a clear definition of the word metaheuristic from [77] “a metaheuristic is a high-level problem independent algorithmic framework that provides a set of guidelines or strategies to develop heuristic optimization algorithms. The term is also used to refer to a problem-specific implementation of a heuristic optimization algorithm according to the guidelines expressed in such framework”. All meta-heuristic techniques have two main components intensification and diversification. Diversification generates assorted solutions to explore the search space more thoroughly on the global scale, while intensification concentrates on search in local scale by using the local information in the search process to generate better solutions. Thus, the current local information can be derivative of the target. Because the heuristic techniques often get stuck in the problem of local optima, meta-heuristic techniques were demonstrating most effective to avert this situation as mentioned in [66, 78, 79]. Meta-heuristic techniques are classified into two categories: swarm intelligence (SI) and bio-inspired. Bio-inspired has penetrated into almost all areas of sciences, data mining, biomedical engineering, control systems, and parallel processing. There are many bio-inspired algorithms such as MA, GA, and imperative competitive algorithm (ICA). Swarm intelligence is a comparatively new technique to resolve the unconstrained optimization problems and is inspired from a social behavior of insect colonies and other animals such as PSO, ACO, artificial bee colony (ABC), glowworm swarm algorithm(GSA), BA, Firefly algorithm (FA), cuckoo search (CS), cat swarm optimization (CSO). Researchers always try to find better algorithms, especially for scheduling task in cloud computing. We present here a comparative analysis of these algorithms based on diverse optimization criteria that reinforce the intensification of Search space. Researchers in [80] presented an algorithm for independent task scheduling in grid computing by an amalgamation of PSO with the gravitational emulation local search (GELS) to avoid local minima problem. The amalgamation PSO–GELS algorithm shows a significant reduction in Makespan time. A novel PSO algorithm was presented in [81], it is based on a hyper-heuristic algorithm for secure tasks scheduling in the grid environment. The hyper-heuristic algorithm reduces both Makespan and cost.

A task scheduling algorithm based on double-fitness adaptive algorithm-job spanning time and load balancing genetic algorithm (JLGA) is introduced by authors in [82] for distribution of load between VMs, energy reduction and minimization of makespan time. This algorithm uses a greedy algorithm to initialize the population. It takes crossover and mutation for adaptive probabilities instead of fixed value. In [83], authors propose a hybrid PSO (HPSO) which is an amalgamation of PSO and TS algorithms. HPSO provides the local search technique by Tabu Search. HPSO enhances randomly generated population by separating it into two equal parts. Part one is improved using PSO, and another part with TS. Then, merging them again into one part to exchange global and

local best position of the particles. HPSO minimizes the makespan and optimizes utilization of the resources. Raghavan et al. [84] solved workflow scheduling problem in the cloud using bat algorithm which gives better results of cost processing compared with Best Resource Selection (BRS) algorithm. To enhance the intensification of search space, an amalgamation of PSO and CS algorithm is presented in [78]. The hybrid PSOC algorithm achieves good resource utilization and makespan reduction for independent task scheduling in cloud computing. The authors in [85] reduces the PSO precocious convergence and improves local search ability by using hill climbing algorithm after each iteration. The hybrid GHPSO algorithm works for discrete problems by using mutation and crossover strategies of a genetic algorithm. GHPSO is used in minimizing costs.

Researchers in [86] applied PSO to minimize the execution cost of running workflow application on the Cloud. PSO generates initial population randomly, while the proposed algorithm in [87] generates the initial population of particles based on shortest job to fastest processor (SJFP) algorithm. Researchers in [88] presented an amalgamation of PSO with the tabu search mechanism (PSOTBM) for independent task scheduling in cloud computing. The amalgamation PSOTBM shows a considerable reduction in energy consumption up to 67.5%. A Novel approach was presented in[89], it uses a family genetic algorithm (FGA) to increase resource utilization by effectively assigning VMs to the suitable physical machines. CSO-GA [90] is a combination of CSO and GA algorithms. This hybrid algorithm optimizes the makespan in comparison with other scheduling techniques. The researchers in [54] have proposed a novel ant colony based algorithm to reduce response time by balancing load via searching under loaded node. This algorithm uses FCFS for allocating the tasks to VMs. To produce optimal solutions for grid scheduling problem, the author in [91] used tree representation for GA solutions for mapping VMs and physical machines. Optimizing the energy saving and maximizing the revenues for service provider are described by authors [92], they presented multi-metrics genetic algorithm for independent tasks scheduling, such as makespan, cost, and energy efficiency. Researchers in [93] reduces the computational time of PSO and enhances the convergence rate by introducing an approach called MHPSO. MHPSO is a combination of mutation concept based PSO algorithm (MPSO) and stander hierarchical PSO algorithm (HPSO).

To balance the load and maximize resource utilization over hosts on data centers, authors in [94] presented novel power aware load balancing method called imperialism competitive algorithm-minimum migration time (ICA-MMT). This method reduces energy consumption in cloud computing data centers. In [95], authors combined bee colony and PSO algorithms the proposed approach is called parallel bee colony optimization particle swarm optimization (PBCOPSO). PBCOPSO shows a significant improvement in minimizing the makespan and maximizing resource utilization. A novel load balancing algorithm is

introduced in [55] based on genetic algorithm in which independent tasks scheduling are addressed. This approach provides a load balancing and an efficient utilization of resources. Population of particles is initialized in [96] to provide an efficient utilization of resources and complete the tasks within a minimum period of time. A Hybrid approach is introduced by authors in [97] called FUGE that uses GA and fuzzy theory to perform optimal load balancing considering execution time and cost. Merging the imperialist competitive and local search optimization algorithms is the contribution of the authors in [98]. This algorithm addresses the reliability issue as well as makespan. This algorithm is compared with ant colony optimization and genetic algorithms and showed better performance. Load balancing approach [99], is introduced based on genetic algorithm in cloud computing environment for balancing load between VMs and reducing dynamic VM migration. Many other works have applied particle swarm optimization to resolve the problem of task scheduling such as [56, 85, 86, 100, 101]. Authors in [53] described and evaluated a cloud scheduler based on ACO, they handled the problem of response time and balancing throughput on a private cloud when various cloud users are executing their experiments.

Hybrid algorithm GA-PSO is presented in [102], the GA-PSO selects VMs based on speed and workflow of the task. This algorithm improves the load balancing and reduces makespan and cost. Kamaljit et al.[103] proposed a novel context and load-aware Family genetic algorithm methodology for efficient task scheduling using modified genetic algorithm known as family genetic algorithm.

Researchers in [104] proposed an algorithm based on bat algorithm (BA) for solving workflow scheduling problem in cloud computing with an objective of reducing the makespan. They implemented the BA in MATLAB and compared results with two popular existing algorithms, namely CSO and PSO. S. A. Hamad and F. A. Omara [105] have proposed task scheduling algorithm based on a modified GA. They overcome the limitation of the population size by using the tournament selection method to select the best chromosomes. Researchers in [106] presented a static task scheduling technique based on PSO algorithm.

They improved PSO using honeybee load balancing technique to increase resource utilization and decrease makespan. The hybrid task scheduling method in [107] uses PSO and hill climbing algorithms.

In this algorithm, the initialization of a population is randomly distributed using PSO. Then, selection of some particles to apply for hill climbing. This technique optimizes the makespan. Priority-based task scheduling in [108] called HGPSO algorithm combines the PSO and GA algorithms. In HGPSO, the tasks are arranged based on priority queue first, then the HGPSO algorithm is applied. The HGPSO performs good in terms of completion time, scalability, and availability compared to genetic and particle swarm optimization algorithms. Researchers in [109] presented a HTSCC Algorithm by

combining the strengths of GA and PSO algorithms to increase resource utilization and decrease makespan. The HTSCC algorithm is implemented and simulated using CloudSim simulator. The simulation results show that HTSCC algorithm outperforms the GA and PSO algorithms by decreasing the makespan and increasing the resource utilization.

In [110] the researchers presented MSDE algorithm depending on improving the performance of the Moth Search Algorithm (MSA) using the differential evolution (DE) to design a task scheduling model and global optimization problem.

To handle the starvation problem the researchers in [111] proposed a hybrid shortest-longest scheduling algorithm. They considered the capabilities of each VM and the length of the task to allocate the tasks to the most convenient VMs, so as to overcome the starvation problem and also satisfying and considering both the provider and user requirements. New hybrid QoS-based task scheduling algorithm is introduced in [112] to schedule dependent and independent tasks in a cloud environment. This work can be extended to implement the hybrid task scheduling algorithm in an effective way using cost involved in communication and energy efficiency.

The comparison of these algorithms is summarized in Table 2. The comparison considers the optimization metrics, nature of task, experimental scale and simulation environment.

4. Discussion

In this section, we provide an analysis and discussion of meta-heuristic techniques for scheduling tasks in cloud computing. The discussion is based on optimization criteria and classification of meta-heuristic techniques.

These techniques are analysed with regard to different optimization criteria as mentioned in section 3.4. Figure 5 depicts different criteria considered by different scheduling techniques. The mainly used criteria for scheduling is observed to be the makespan (33%), cost (18%), load balancing (16%), deadline (9%), and energy efficiency (9%). Less attention is given to budget (4%), throughput (4%), and resource utilization (7%). Some of these criteria are preferred by the user and others by the providers. Makespan, cost, budget, and deadline are preferred by the users, while resource utilization, throughput, load balancing, and energy efficiency are preferred by the providers. Minimization of makespan is an important factor for cloud users desire to speed up the execution of their applications.

Decisions in scheduling algorithms mostly based on the makespan metric of the applications. Thus, minimizing the makespan become the major concerns of the researchers to enhance the performane.

Moreover, Cost comes next in the researcher's interest. Various types of costs have been considered in the literature such as data storage, data transfer, data renting, communication and computation costs. Table 3 clearly

depicts some types of costs used in meta-heuristic scheduling techniques.

Table 2. Comparison on existing scheduling task algorithms

Approach/Algorithm	Nature of Tasks	Optimization Criteria	User/Provider desired Or Both	Experimental Scale	Simulation Environment
[80] PSO-GELS	INDEPENDENT	MAKESPAN	USER	VARIOUS RESOURCES VARYING WORKLOAD	JAVA SIMULATION ENVIRONMENT
[81] A NOVEL PSO - BASED HYPER-HEURISTIC	INDEPENDENT	MAKESPAN, COST	USER	150-250 RESOURCES 700 TASKS	GRIDSIM
[82] JLGA	INDEPENDENT	MAKESPAN, LOAD BALANCING, COST, ENERGY EFFICIENCY	BOTH	50 RESOURCES 1-90 TASKS	MATLAB
[83] HPSO	DEPENDENT	BUDGET	USER	TWO RESOURCES 200, 400, 600, 800 AND 1000 TASKS	CLOUDSIM
[84] WORKFLOW SCHEDULING BASED ON A BAT ALGORITHM	DEPENDENT	MAKESPAN, COST	USER	THREE RESOURCES FOUR TASKS	NOT MENTIONED
[78] PSOCS	INDEPENDENT	MAKESPAN, RESOURCE UTILIZATION	BOTH	10- 40 TASKS	CLOUDSIM
[85] GHPSO	DEPENDENT	COST, DEADLINE	USER	RANDOMLY GENERATED	CLOUD SIMULATION ENVIRONMENT (JAVA)
[86] PSO BASED HEURISTIC	DEPENDENT	MAKESPAN, COST	USER	AMAZON CLOUDFRONT	JSWARM PACKAGE
[87] MODIFIED PSO ALGORITHM	INDEPENDENT	MAKESPAN	USER	4 RESOURCES 100 - 800 TASKS	EUCALYPTUS CLOUD
[88] PSOTBM	INDEPENDENT	ENERGY EFFICIENCY	PROVIDER	100 - 5000 RESOURCES	CLOUDSIM
[89] A NOVEL FAMILY GENETIC APPROACH	VM PLACEMENT	RESOURCE UTILIZATION	PROVIDER	VARIOUS RESOURCES VARYING WORKLOAD	CLOUDSIM
[90] CSO-GA	DEPENDENT	MAKESPAN	USER	7-25 RESOURCES 50-100 TASKS	JAVA
[54] A NOVEL ANT COLONY BASED ALGORITHM	INDEPENDENT	LOAD BALANCING	PROVIDER	25, 50 AND 75 RESOURCES ONLINE TASKS	CLOUD ANALYST

Table 2. (continued)

Approach/Algorithm	Nature of Tasks	Optimization Criteria	User/Provider desired Or Both	Experimental Scale	Simulation Environment
[91] A GENETIC ALGORITHM SCHEDULING APPROACH	DEPENDENT	BUDGET AND DEADLINE CONSTRAINTS	USER	NA	GRID SIMULATION ENVIRONMENT
[92] MULTI-OBJECTIVE GENETIC ALGORITHM	INDEPENDENT	MAKESPAN, COST, ENERGY EFFICIENCY	BOTH	NA	CLOUDSIM
[94] ICA-MMT	INDEPENDENT	ENERGY EFFICIENCY, LOAD BALANCING	PROVIDER	800 RESOURCES	CLOUDSIM
[95] PBCOPSO	INDEPENDENT	MAKESPAN, RESOURCE UTILIZATION	BOTH	FOUR RESOURCES 40 - 640 TASKS	CLOUDSIM
[55] A NOVEL LOAD BALANCING STRATEGY USING GENETIC ALGORITHM	INDEPENDENT	LOAD BALANCING	PROVIDER	25, 50 AND 75 RESOURCES	CLOUD ANALYST
[96] FUZZY PARTICLE SWARM OPTIMIZATION ALGORITHM	INDEPENDENT	MAKESPAN	USER	THREE RESOURCES 40 - 640 TASKS	GRID SIMULATION ENVIRONMENT
[97] FUGE	INDEPENDENT	MAKESPAN, COST, LOAD BALANCING	BOTH	50 RESOURCES 100 - 1000 TASKS	CLOUDSIM, MATLAB
[98] RESOURCE PLANNING BASED ON ICA	INDEPENDENT	MAKESPAN	USER	NA	NOT MENTIONED
[99] SCHEDULING STRATEGY BASED ON GA	INDEPENDENT	LOAD BALANCING	PROVIDER	5 RESOURCES 15 TASKS	CLOUD EXPERIMENTAL ENVIRONMENT
[100] SCHEDULING STRATEGY FOR SCIENTIFIC WORKFLOWS ON IAAS.	DEPENDENT	MAKESPAN, DEADLINE	USER	6 RESOURCES 100 TASKS	CLOUDSIM
[56] AN OPTIMIZED SCHEDULING USING A DISCRETE PARTICLE SWARM	DEPENDENT	MAKESPAN, LOAD BALANCING, COST	BOTH	INTERVAL OF THE TASKS NUMBER [5, 500]	CLOUDSIM
[101] A REVISED DISCRETE PARTICLE SWARM OPTIMIZATION	DEPENDENT	DEADLINE, COST	USER	AMAZON CLOUD FRONT	AMAZON EC2
[53] BALANCING THROUGHPUT AND RESPONSE TIME IN ONLINE SCIENTIFIC CLOUDS VIA ACO	VM PLACEMENT	THROUGHPUT	PROVIDER	VARIOUS RESOURCES VARYING WORKLOAD	CLOUDSIM

Table 2. (continued)

Approach\Algorithm	Nature of Tasks	Optimization Criteria	User/Provider desired Or Both	Experimental Scale	Simulation Environment
[102]HYBRID GA-PSO ALGORITHM	DEPENDENT	MAKESPAN, COST, LOAD BALANCE	BOTH	16 RESOURCES 25 – 1000 TASKS	WORKFLOWSIM SIMULATOR
[103]A NOVEL CONTEXT AND LOAD-AWARE FAMILY GENETIC ALGORITHM	DEPENDENT	LOAD BALANCE	PROVIDER	DIFFERENT TYPES OF WORKLOADS FOR VARIOUS CONTEXT-AWARE WORKLOAD	CLOUDSIM
[104]BAT ALGORITHM (BA)	DEPENDENT	MAKESPAN, COST	USER	5 RESOURCES 17 TASKS	MATLAB
[105] TS-GA	INDEPENDENT	MAKESPAN, COST, RESOURCE UTILIZATION	BOTH	8 RESOURCES 25–100 TASKS	CLOUDSIM
[106] PSO METHOD USING A LOAD-BALANCING TECHNIQUE	INDEPENDENT	MAKESPAN, RESOURCE UTILIZATION	BOTH	3 RESOURCES 80 TASKS	NOT MENTIONED
[107] HYBRID PSO-HILL CLIMBING ALGORITHM	DEPENDENT	MAKESPAN	USER	8 RESOURCES 100 TASKS	C# IN AN AZURE CLOUD ENVIRONMENT
[108] HGPSO	INDEPENDENT	MAKESPAN	USER	0 - 25 RESOURCES 50 TASKS	NOT MENTIONED
[109] HTSCC	INDEPENDENT	MAKESPAN, RESOURCE UTILIZATION	BOTH	5 - 10 RESOURCES 10 - 40 TASKS	CLOUDSIM
[110] MSDE ALGORITHM	INDEPENDENT	MAKESPAN	USER	2 RESOURCES 100 - 600 TASKS	CLOUDSIM
[111] HSLJF ALGORITHM	INDEPENDENT	MAKESPAN, RESOURCE UTILIZATION	BOTH	2 RESOURCES WORKLOAD GENERATED FROM A STANDARD FORMATTED WORKLOAD OF A HIGH-PERFORMANCE COMPUTING CENTER CALLED HPC2N IN SWEDEN AS BENCHMARK	CLOUDSIM
[112] NEW HYBRID QOS-BASED TASK SCHEDULING ALGORITHM	DEPENDENT INDEPENDENT	MAKESPAN, COST, RESOURCE UTILIZATION	BOTH	2 RESOURCES 100 - 5000 TASKS	CLOUDSIM

Table 3. Types of cost used in meta-heuristic techniques

Approach	Types of cost
[113]	DATA RENTING COST.
[114]	DATA STORAGE, DATA TRANSFER COST.
[86, 115, 116]	COMPUTATION, TRANSFER COST.
[117]	DATA STORAGE, COMPUTATION, DATA TRANSFER COST.
[118]	COMMUNICATION, COMPUTATION, TRANSFER COST.
[51, 119-121]	COMPUTATION COST.
[122]	COMMUNICATION, COMPUTATION COST.

Since scheduling tasks to resources spreads over different data centers from many cloud service providers, this increases data transfer cost and complicates data management. Also, the cost is increasing, if data are stored on a permanent storage. On the other hand, the cost is negligible if data storage is in a single data center on the same cloud provider or shared between multiple applications.

Resource utilization and load balancing of virtual machines are of the most significant factors considered by service providers.

Switching off the VMs results in decreasing the consumption of energy in data centers which is rather an important factor for service providers. Only few researchers have considered the budget and throughput in tasks scheduling. Both of them scores only 4% form the total number of approaches as depicted in Figure 5. Meta-heuristic techniques such as PSO, GA, ICA and ACO are the mostly addressed techniques. Meta-heuristic techniques are classified into swarm intelligence and bio-inspired approaches, see Figure 6.

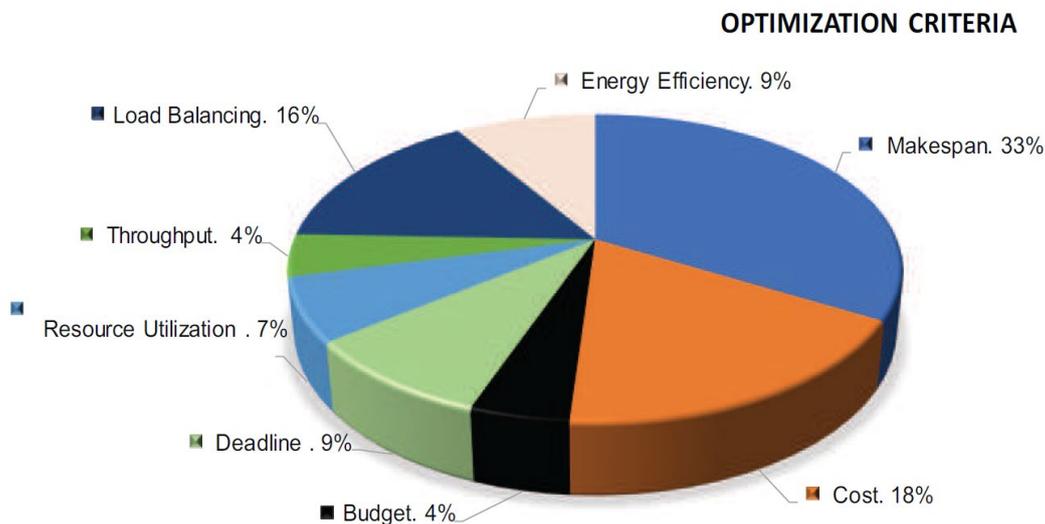


Figure 5. The Frequency of optimization criteria considered by meta-heuristic techniques

These factors influence providers' outcome and profit. Merging the load on virtual machines increases resource utilization. Thus, virtual machines could be either used for new tasks or switched off.

In the state of swarm intelligence approaches, PSO is the most widely used approach by the researchers in the cloud scheduling problems domain. Correspondingly in bio-inspired approaches, GA is the most widely used approach.

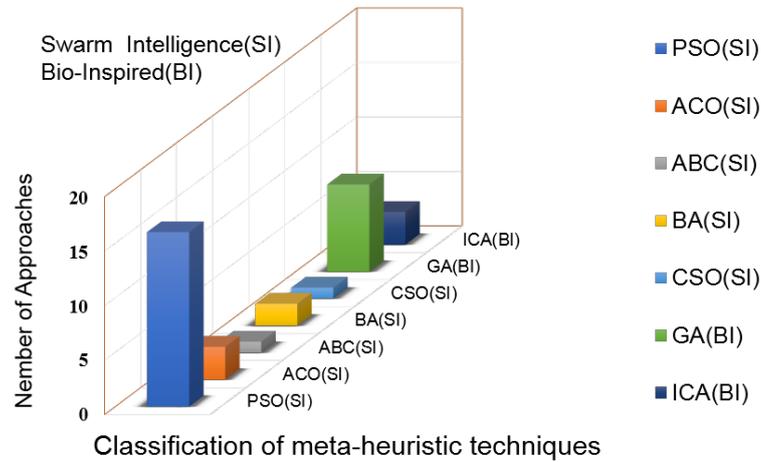


Figure 6. The approaches used for task scheduling in CC

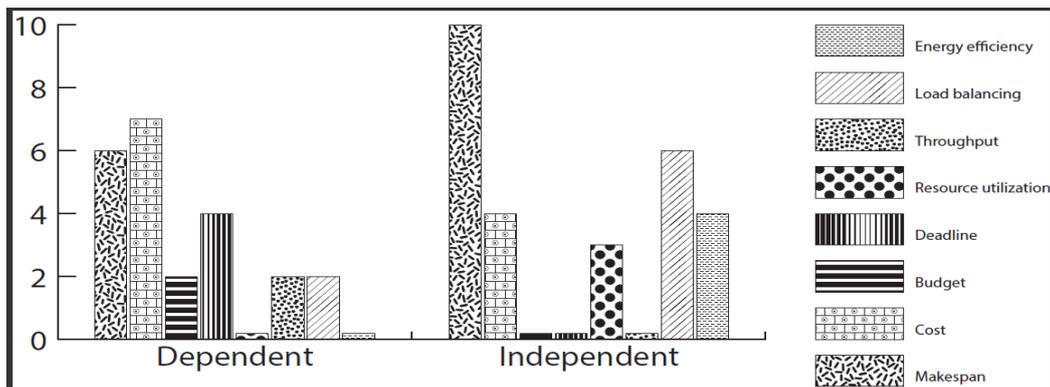


Figure 7. Optimization criteria considered in meta-heuristic techniques

The frequency of optimization criteria based on nature of the task is depicted in Figure 7.

The figure shows the deadline, throughput, and budget for dependent tasks and makespan, load balancing, energy efficiency and resource utilization, for independent tasks. The Meta-heuristic techniques are intended for global optimization, though they are not always successful or efficient. So, the researchers combine them with other techniques to improve the efficiency.

The intensification and diversification are the main components of meta-heuristic techniques. They enable researchers to explore the search space in the global and local region. The perfect amalgamation of these components helps to obtain global optimality. But this amalgamation needs a lot of techniques to be explored. Most of the researchers implement the hybrid meta-

heuristic techniques using CloudSim simulator [37] to simulate the performance of the optimization criteria.

5. Conclusion and future work

In this paper, we introduced the main concepts of scheduling and resource management in cloud computing. Moreover, we presented a comparative analysis of meta-heuristic scheduling techniques in cloud computing by considering the optimization criteria, natures of tasks, user and provider desire, and simulation environment. According to the reviewed literature, we concluded that most of the works are done based on the popular meta-heuristic techniques in cloud computing such as PSO, GA, ACO algorithms. The most used techniques for scheduling are observed to be the PSO in swarm intelligence and GA in bio-inspired. Moreover, there are

other algorithms like ICA, CSO, BA, ABC have been used in task scheduling but less so than other algorithms.

Finally, we conclude that the makespan is the mostly studied criterion in the literature. For future work we are planning to come up with a new model for failure handling using hybrid meta-heuristic techniques.

References

- [1] F. F. Moghaddam, M. Ahmadi, S. Sarvari, M. Eslami, and A. Golkar, "Cloud computing challenges and opportunities: A survey," in *Telematics and Future Generation Networks (TAFGEN), 2015 1st International Conference on*, 2015, pp. 34-38, 2015.
- [2] M. Oppitz and P. Tomsu, "Future Technologies of the Cloud Century," in *Inventing the Cloud Century*, ed: Springer, pp. 511-545, 2018.
- [3] A. A. Laghari, H. He, I. A. Halepoto, M. S. Memon, and S. Parveen, "Analysis of quality of experience frameworks for cloud computing," *IJCSNS*, vol. 17, p. 228, 2017.
- [4] V. Kumar, A. A. Laghari, S. Karim, M. Shakir, and A. A. Brohi, "Comparison of Fog Computing & Cloud Computing," *International Journal of Mathematical Sciences and Computing (IJMSC)*, vol. 5, pp. 31-41, 2019.
- [5] R. Buyya, J. Broberg, and A. M. Goscinski, "Cloud Computing Principles and Paradigms," 2011.
- [6] S. Mittal and A. Katal, "An Optimized Task Scheduling Algorithm in Cloud Computing," in *Advanced Computing (IACC), 2016 IEEE 6th International Conference on*, 2016, pp. 197-202, 2016.
- [7] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, pp. 7-18, 2010.
- [8] A. A. Laghari, H. He, M. Shafiq, and A. Khan, "Assessing effect of Cloud distance on end user's Quality of Experience (QoE)," in *2016 2nd IEEE international conference on computer and communications (ICCC)*, pp. 500-505, 2016.
- [9] A. A. Laghari, H. He, A. Khan, N. Kumar, and R. Kharel, "Quality of experience framework for cloud computing (QoC)," *IEEE Access*, vol. 6, pp. 64876-64890, 2018.
- [10] A. A. Laghari, H. He, M. Shafiq, and A. Khan, "Assessment of quality of experience (QoE) of image compression in social cloud computing," *Multiaagent and Grid Systems*, vol. 14, pp. 125-143, 2018.
- [11] X. Tan and B. Ai, "The issues of cloud computing security in high-speed railway," in *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, 2011, pp. 4358-4363, 2011.
- [12] M. A. Sharkh, M. Jammal, A. Shami, and A. Ouda, "Resource allocation in a network-based cloud computing environment: design challenges," *IEEE Communications Magazine*, vol. 51, pp. 46-52, 2013.
- [13] S. A. de Chaves, B. Carlos, C. M. Westphall, and G. A. Gerônimo, "Customer security concerns in cloud computing," in *Proc. The Tenth International Conf. on Networks (ICN)*, pp. 7-11, 2011.
- [14] R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente, "Key challenges in cloud computing: Enabling the future internet of services," *IEEE Internet Computing*, vol. 17, pp. 18-25, 2013.
- [15] D. Petcu, "Portability and interoperability between clouds: challenges and case study," in *European Conference on a Service-Based Internet*, pp. 62-74, 2011.
- [16] N. Rinaldo and E. Zimeo, "Time and cost-driven scheduling of data parallel tasks in grid workflows," *IEEE Systems Journal*, vol. 3, pp. 104-120, 2009.
- [17] T. Mathew, K. C. Sekaran, and J. Jose, "Study and analysis of various task scheduling algorithms in the cloud computing environment," in *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on*, 2014, pp. 658-664, 2014.
- [18] E.-G. Talbi, *Metaheuristics: from design to implementation* vol. 74: John Wiley & Sons, 2009.
- [19] L. Wang, Y. Ma, J. Yan, V. Chang, and A. Y. Zomaya, "pipsCloud: High performance cloud computing for remote sensing big data management and processing," *Future Generation Computer Systems*, vol. 78, pp. 353-368, 2018.
- [20] S. M. Parikh, N. M. Patel, and H. B. Prajapati, "Resource Management in Cloud Computing: Classification and Taxonomy," *arXiv preprint arXiv:1703.00374*, 2017.
- [21] M. Sookhak, A. Gani, M. K. Khan, and R. Buyya, "Dynamic remote data auditing for securing big data storage in cloud computing," *Information Sciences*, vol. 380, pp. 101-116, 2017.
- [22] B. Jennings and R. Stadler, "Resource management in clouds: Survey and research challenges," *Journal of Network and Systems Management*, vol. 23, pp. 567-619, 2015.
- [23] A. Fischer, J. F. Botero, M. T. Beck, H. De Meer, and X. Hesselbach, "Virtual network embedding: A survey," *IEEE Communications Surveys & Tutorials*, vol. 15, pp. 1888-1906, 2013.
- [24] Q. Duan, Y. Yan, and A. V. Vasilakos, "A survey on service-oriented network virtualization toward convergence of networking and cloud computing," *IEEE Transactions on Network and Service Management*, vol. 9, pp. 373-392, 2012.
- [25] C. Stergiou, K. E. Psannis, B.-G. Kim, and B. Gupta, "Secure integration of IoT and cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 964-975, 2018.
- [26] S. S. Manvi and G. K. Shyam, "Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey," *Journal of Network and Computer Applications*, vol. 41, pp. 424-440, 2014.
- [27] M. Hussain and H. Abdulsalam, "SECaaS: security as a service for cloud-based applications," in *Proceedings of the Second Kuwait Conference on e-Services and e-Systems*, p. 8, 2011.
- [28] X. Ye, Y. Yin, and L. Lan, "Energy-Efficient Many-objective Virtual Machine Placement Optimization in a Cloud Computing Environment," *IEEE Access*, 2017.
- [29] V. Anuradha and D. Sumathi, "A survey on resource allocation strategies in cloud computing," in *Information Communication and Embedded Systems (ICICES), 2014 International Conference on*, 2014, pp. 1-7, 2014.

- [30] B. B. Nagesh, "Resource Provisioning Techniques in Cloud Computing Environment-A Survey," *IJRCCCT*, vol. 3, pp. 395-401, 2014.
- [31] A. Leivadreas, C. Papagianni, and S. Papavassiliou, "Efficient resource mapping framework over networked clouds via iterated local search-based request partitioning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, pp. 1077-1086, 2013.
- [32] P. T. Endo, A. V. de Almeida Palhares, N. N. Pereira, G. E. Goncalves, D. Sadok, J. Kelner, *et al.*, "Resource allocation for distributed cloud: concepts and research challenges," *IEEE network*, vol. 25, 2011.
- [33] A. Reuther, C. Byun, W. Arcand, D. Bestor, B. Bergeron, M. Hubbell, *et al.*, "Scalable System Scheduling for HPC and Big Data," *arXiv preprint arXiv:1705.03102*, 2017.
- [34] R. Nallakumar, N. Sengottaiyan, and K. S. Priya, "A Survey on Scheduling and the Attributes of Task Scheduling in the Cloud," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 3, pp. 8167-8171, 2014.
- [35] J. Yu, R. Buyya, and K. Ramamohanarao, "Workflow scheduling algorithms for grid computing," *Metaheuristics for scheduling in distributed computing environments*, pp. 173-214, 2008.
- [36] R. D. Lakshmi and N. Srinivasu, "A Review and Analysis of Task Scheduling Algorithms in Different Cloud Computing Environments," *International Journal of Computer Science and Mobile Computing*, vol. 4, 2015.
- [37] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and experience*, vol. 41, pp. 23-50, 2011.
- [38] R. Kumar and G. Sahoo, "Cloud computing simulation using CloudSim," *arXiv preprint arXiv:1403.3253*, 2014.
- [39] A. N. Toosi, R. N. Calheiros, and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys (CSUR)*, vol. 47, p. 7, 2014.
- [40] J. Baliga, R. W. Ayre, K. Hinton, and R. S. Tucker, "Green cloud computing: Balancing energy in processing, storage, and transport," *Proceedings of the IEEE*, vol. 99, pp. 149-167, 2011.
- [41] F. Xu, F. Liu, H. Jin, and A. V. Vasilakos, "Managing performance overhead of virtual machines in cloud computing: A survey, state of the art, and future directions," *Proceedings of the IEEE*, vol. 102, pp. 11-31, 2014.
- [42] B. Song, M. Hassan, E.-N. Huh, C.-W. Yoon, and H.-W. Lee, "A hybrid algorithm for partner selection in market oriented cloud computing," in *Management and Service Science, 2009. MASS'09. International Conference on*, 2009, pp. 1-4, 2009.
- [43] F. du Pin Calmon, J. M. Cloud, M. Medard, and W. Zeng, "Multi-path data transfer using Network Coding," ed: Google Patents, 2017.
- [44] Z.-H. Zhan, X.-F. Liu, Y.-J. Gong, J. Zhang, H. S.-H. Chung, and Y. Li, "Cloud computing resource scheduling and a survey of its evolutionary approaches," *ACM Computing Surveys (CSUR)*, vol. 47, p. 63, 2015.
- [45] A. Al-maamari and F. A. Omara, "Task scheduling using PSO algorithm in cloud computing environments," *International Journal of Grid and Distributed Computing*, vol. 8, pp. 245-256, 2015.
- [46] A. Abro, Z. Deng, K. A. Memon, A. A. Laghari, and K. H. Mohammadani, "A Dynamic Application-Partitioning Algorithm with Improved Offloading Mechanism for Fog Cloud Networks," *Future Internet*, vol. 11, p. 141, 2019.
- [47] A. A. Laghari, H. He, S. Karim, H. A. Shah, and N. K. Karn, "Quality of experience assessment of video quality in social clouds," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.
- [48] M. S. A. Latiff, S. H. H. Madni, and M. Abdullahi, "Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm," *Neural Computing and Applications*, vol. 29, pp. 279-293, 2018.
- [49] S. Tang, B. He, H. Liu, and B.-S. Lee, "9 Resource Management in Big Data Processing Systems," 2016.
- [50] J. Yu and R. Buyya, "Scheduling scientific workflow applications with deadline and budget constraints using genetic algorithms," *Scientific Programming*, vol. 14, pp. 217-230, 2006.
- [51] D. Poola, S. K. Garg, R. Buyya, Y. Yang, and K. Ramamohanarao, "Robust scheduling of scientific workflows with deadline and budget constraints in clouds," in *Advanced Information Networking and Applications (AINA), 2014 IEEE 28th International Conference on*, 2014, pp. 858-865, 2014.
- [52] J. Lin, Y. Zhong, X. Lin, H. Lin, and Q. Zeng, "Hybrid ant colony algorithm clonal selection in the application of the cloud's resource scheduling," *arXiv preprint arXiv:1411.2528*, 2014.
- [53] E. Pacini, C. Mateos, and C. G. Garino, "Balancing throughput and response time in online scientific Clouds via Ant Colony Optimization (SP2013/2013/00006)," *Advances in Engineering Software*, vol. 84, pp. 31-47, 2015.
- [54] S. Dam, G. Mandal, K. Dasgupta, and P. Dutta, "An ant colony based load balancing strategy in cloud computing," in *Advanced Computing, Networking and Informatics-Volume 2*, ed: Springer, pp. 403-413, 2014.
- [55] K. Dasgupta, B. Mandal, P. Dutta, J. K. Mandal, and S. Dam, "A genetic algorithm (ga) based load balancing strategy for cloud computing," *Procedia Technology*, vol. 10, pp. 340-347, 2013.
- [56] C. Jianfang, C. Junjie, and Z. Qingshan, "An optimized scheduling algorithm on a cloud workflow using a discrete particle swarm," *Cybernetics and Information Technologies*, vol. 14, pp. 25-39, 2014.
- [57] S. Bajaj, "Current Drift in Energy Efficiency Cloud Computing: New Provocations, Workload Prediction, Consolidation, and Resource Over Commitment," in *Critical Research on Scalability and Security Issues in Virtual Cloud Environments*, ed: IGI Global, pp. 283-303, 2018.
- [58] A. A. Laghari, H. He, M. Shafiq, and A. Khan, "Impact of storage of mobile on quality of experience (QoE) at user level accessing cloud," in *2017 IEEE 9th international conference on communication software and networks (ICCSN)*, pp. 1402-1409, 2017.
- [59] P. Brucker, "Scheduling algorithms 5th edition," ed: Berlin: Springer, 2006.

- [60] J. Blazewicz, K. H. Ecker, E. Pesch, G. Schmidt, M. Sterna, and J. Weglarz, "Scheduling under resource constraints," in *Handbook on Scheduling*, ed: Springer, pp. 475-525, 2019.
- [61] J. W. Herrmann, *Handbook of production scheduling* vol. 89: Springer Science & Business Media, 2006.
- [62] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, and M. Karamanoglu, *Swarm intelligence and bio-inspired computation: theory and applications*: Newnes, 2013.
- [63] A. Karthick, E. Ramaraj, and R. G. Subramanian, "An efficient multi queue job scheduling for cloud computing," in *Computing and Communication Technologies (WCCCT), 2014 World Congress on*, pp. 164-166, 2014.
- [64] Q. Li and Y. Guo, "Optimization of resource scheduling in cloud computing," in *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2010 12th International Symposium on*, pp. 315-320, 2010.
- [65] P. Singh, M. Dutta, and N. Aggarwal, "A review of task scheduling based on meta-heuristics approach in cloud computing," *Knowledge and Information Systems*, pp. 1-51, 2017.
- [66] M. Kalra and S. Singh, "A review of metaheuristic scheduling techniques in cloud computing," *Egyptian informatics journal*, vol. 16, pp. 275-295, 2015.
- [67] D. Nayak, S. K. Malla, and D. Debadarshini, "Improved round robin scheduling using dynamic time quantum," *International Journal of Computer Applications (0975-8887) Volume*, 2012.
- [68] P. Sangwan, M. Sharma, and A. Kumar, "Improved Round Robin Scheduling in Cloud Computing," *Advances in Computational Sciences and Technology*, vol. 10, pp. 639-644, 2017.
- [69] J. Li, T. Ma, M. Tang, W. Shen, and Y. Jin, "Improved FIFO Scheduling Algorithm Based on Fuzzy Clustering in Cloud Computing," *Information*, vol. 8, p. 25, 2017.
- [70] S. Elmougy, S. Sarhan, and M. Joundy, "A novel hybrid of Shortest job first and round Robin with dynamic variable quantum time task scheduling technique," *Journal of Cloud Computing*, vol. 6, p. 12, 2017.
- [71] P. Thakur and M. Mahajan, "Different Scheduling Algorithm in Cloud Computing: A Survey," *International Journal of modern computer science*, vol. 5, 2017.
- [72] N. Sharma, D. S. Tyagi, and S. Atri, "HYMM: A New Heuristic in Cloud Computing," *International Research Journal of Engineering and Technology (IRJET)*, vol. 4, pp. 3520-3526, 2017.
- [73] R. J. S. Raj and S. M. Prasad, "Survey on variants of heuristic algorithms for scheduling workflow of tasks," in *Circuit, Power and Computing Technologies (ICCPCT), 2016 International Conference on*, 2016, pp. 1-4, 2016.
- [74] U. Bhoi and P. N. Ramanuj, "Enhanced max-min task scheduling algorithm in cloud computing," *International Journal of Application or Innovation in Engineering and Management (IJAIEEM)*, vol. 2, pp. 259-264, 2013.
- [75] H. Chen, F. Wang, N. Helian, and G. Akanmu, "User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing," in *Parallel Computing Technologies (PARCOMPTECH), 2013 National Conference on*, 2013, pp. 1-8, 2013.
- [76] X.-S. Yang, *Nature-inspired metaheuristic algorithms*: Luniver press, 2010.
- [77] K. Sörensen and F. W. Glover, "Metaheuristics," in *Encyclopedia of operations research and management science*, ed: Springer, pp. 960-970, 2013.
- [78] A. Al-maamari and F. A. Omara, "Task scheduling using hybrid algorithm in cloud computing environments," *Journal of Computer Engineering (IOSR-JCE)*, vol. 17, pp. 96-106, 2015.
- [79] C.-W. Tsai and J. J. Rodrigues, "Metaheuristic scheduling for cloud: A survey," *IEEE Systems Journal*, vol. 8, pp. 279-291, 2014.
- [80] Z. Pooranian, M. Shojafar, J. H. Abawajy, and A. Abraham, "An efficient meta-heuristic algorithm for grid computing," *Journal of Combinatorial Optimization*, vol. 30, pp. 413-434, 2015.
- [81] R. Aron, I. Chana, and A. Abraham, "A hyper-heuristic approach for resource provisioning-based scheduling in grid environment," *The Journal of Supercomputing*, vol. 71, pp. 1427-1450, 2015.
- [82] T. Wang, Z. Liu, Y. Chen, Y. Xu, and X. Dai, "Load balancing task scheduling based on genetic algorithm in cloud computing," in *Dependable, Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on*, 2014, pp. 146-152, 2014.
- [83] M. Sridhar and G. R. M. Babu, "Hybrid particle swarm optimization scheduling for cloud computing," in *Advance Computing Conference (IACC), 2015 IEEE International*, pp. 1196-1200, 2015.
- [84] S. Raghavan, P. Sarwesh, C. Marimuthu, and K. Chandrasekaran, "Bat algorithm for scheduling workflow applications in cloud," in *Electronic Design, Computer Networks & Automated Verification (EDCAV), 2015 International Conference on*, 2015, pp. 139-144, 2015.
- [85] S.-J. Xue and W. Wu, "Scheduling workflow in cloud computing based on hybrid particle swarm algorithm," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, pp. 1560-1566, 2012.
- [86] S. Pandey, L. Wu, S. M. Guru, and R. Buyya, "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments," in *Advanced information networking and applications (AINA), 2010 24th IEEE international conference on*, 2010, pp. 400-407, 2010.
- [87] S. Abdi, S. A. Motamedi, and S. Sharifian, "Task scheduling using Modified PSO Algorithm in cloud computing environment," in *International conference on machine learning, electrical and mechanical engineering*, pp. 8-9, 2014.
- [88] Z. Wang, K. Shuang, L. Yang, and F. Yang, "Energy-aware and revenue-enhancing Combinatorial Scheduling in Virtualized of Cloud Datacenter," *Journal of Convergence Information Technology*, vol. 7, pp. 62-70, 2012.
- [89] C. T. Joseph, K. Chandrasekaran, and R. Cyriac, "A novel family genetic approach for virtual machine allocation," *Procedia Computer Science*, vol. 46, pp. 558-565, 2015.
- [90] S. ROUHI and E. B. NEJAD, "CSO-GA: a new scheduling technique for cloud computing systems based on cat swarm optimization and genetic algorithm," *Cumhuriyet Science Journal*, vol. 36, pp. 1672-1685, 2015.

- [91] S. Sawant, "A genetic algorithm scheduling approach for virtual machine resources in a cloud computing environment," 2011.
- [92] J. Liu, X.-G. Luo, X.-M. Zhang, F. Zhang, and B.-N. Li, "Job scheduling model for cloud computing based on multi-objective genetic algorithm," *IJCSI International Journal of Computer Science Issues*, vol. 10, pp. 134-139, 2013.
- [93] B. Zarei, R. Ghanbarzadeh, P. Khodabande, and H. Toofani, "MHPSO: A new method to enhance the particle swarm optimizer," in *Digital Information Management (ICDIM), 2011 Sixth International Conference on*, 2011, pp. 305-309, 2011.
- [94] S. Yakhchi, S. Ghafari, M. Yakhchi, M. Fazeli, and A. Patooghy, "ICA-MMT: A load balancing method in cloud computing environment," in *Web Applications and Networking (WSWAN), 2015 2nd World Symposium on*, 2015, pp. 1-7, 2015.
- [95] R. J. Priyadarsini and L. Arockiam, "PBCOPSO: A parallel optimization algorithm for task scheduling in cloud environment," *Indian Journal of Science and Technology*, vol. 8, 2015.
- [96] H. Liu, A. Abraham, and A. E. Hassanien, "Scheduling jobs on computational grids using a fuzzy particle swarm optimization algorithm," *Future Generation Computer Systems*, vol. 26, pp. 1336-1343, 2010.
- [97] M. Shojafar, S. Javanmardi, S. Abolfazli, and N. Cordeschi, "FUGE: A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method," *Cluster Computing*, vol. 18, pp. 829-844, 2015.
- [98] A. ABARGHOEI, E. MAHDIPOUR, and M. ASKARZADEH, "Cloud computing resource planning based on imperialist competitive algorithm," *Cumhuriyet Science Journal*, vol. 36, pp. 1311-1324, 2015.
- [99] J. Gu, J. Hu, T. Zhao, and G. Sun, "A new resource scheduling strategy based on genetic algorithm in cloud computing environment," *Journal of Computers*, vol. 7, pp. 42-52, 2012.
- [100] M. A. Rodriguez and R. Buyya, "Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds," *IEEE Transactions on Cloud Computing*, vol. 2, pp. 222-235, 2014.
- [101] Z. Wu, Z. Ni, L. Gu, and X. Liu, "A revised discrete particle swarm optimization for cloud workflow scheduling," in *Computational Intelligence and Security (CIS), 2010 International Conference on*, 2010, pp. 184-188, 2010.
- [102] A. M. Manasrah and H. Ba Ali, "Workflow Scheduling Using Hybrid GA-PSO Algorithm in Cloud Computing," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.
- [103] K. Kaur, N. Kaur, and K. Kaur, "A Novel Context and Load-Aware Family Genetic Algorithm Based Task Scheduling in Cloud Computing," in *Data Engineering and Intelligent Computing*, ed: Springer, pp. 521-531, 2018.
- [104] S. Sagnika, S. Bilgaiyan, and B. S. P. Mishra, "Workflow Scheduling in Cloud Computing Environment Using Bat Algorithm," in *Proceedings of First International Conference on Smart System, Innovations and Computing*, pp. 149-163, 2018.
- [105] S. A. Hamad and F. A. Omara, "Genetic-based task scheduling algorithm in cloud computing environment," *International Journal of Advanced Computer Science and Applications*, vol. 7, pp. 550-556, 2016.
- [106] F. Ebadifard and S. M. Babamir, "A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment," *Concurrency and Computation: Practice and Experience*, vol. 30, p. e4368, 2018.
- [107] N. Dordaie and N. J. Navimipour, "A hybrid particle swarm optimization and hill climbing algorithm for task scheduling in the cloud environments," *ICT Express*, 2017.
- [108] A. S. Kumar and M. Venkatesan, "Task scheduling in a cloud computing environment using HGPSO algorithm," *Cluster Computing*, pp. 1-7, 2018.
- [109] R. A. Al-Arasi and A. Saif, "HTSCC: A Hybrid Task Scheduling Algorithm in Cloud Computing Environment," *International Journal Of Computers & Technology*, vol. 17, pp. 7236-7246, 2018.
- [110] M. A. Elaziz, S. Xiong, K. Jayasena, and L. Li, "Task scheduling in cloud computing based on hybrid moth search algorithm and differential evolution," *Knowledge-Based Systems*, vol. 169, pp. 39-52, 2019.
- [111] M. A. Alworafi, A. Dhari, S. A. El-Booz, A. A. Nasr, A. Arpitha, and S. Mallappa, "An Enhanced Task Scheduling in Cloud Computing Based on Hybrid Approach," in *Data Analytics and Learning*, ed: Springer, pp. 11-25, 2019.
- [112] S. Potluri and K. S. Rao, "Simulation of QoS-Based Task Scheduling Policy for Dependent and Independent Tasks in a Cloud Environment," in *Smart Intelligent Computing and Applications*, ed: Springer, pp. 515-525, 2020.
- [113] H. R. Faragardi, A. Rajabi, R. Shojaei, and T. Nolte, "Towards energy-aware resource scheduling to maximize reliability in cloud computing systems," in *High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC), 2013 IEEE 10th International Conference on*, 2013, pp. 1469-1479, 2013.
- [114] P. Bryk, M. Malawski, G. Juve, and E. Deelman, "Storage-aware algorithms for scheduling of workflow ensembles in clouds," *Journal of Grid Computing*, vol. 14, pp. 359-378, 2016.
- [115] A. Verma and S. Kaushal, "Budget constrained priority based genetic algorithm for workflow scheduling in cloud," 2013.
- [116] R. Pragaladan and R. Maheswari, "Improve workflow scheduling technique for novel particle swarm optimization in cloud environment," *Int. J. Eng. Res. Gen. Sci*, vol. 2, 2014.
- [117] H. M. Fard, R. Prodan, J. J. D. Barrionuevo, and T. Fahringer, "A multi-objective approach for workflow scheduling in heterogeneous environments," in *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*, 2012, pp. 300-309, 2012.
- [118] E. Barrett, E. Howley, and J. Duggan, "A learning architecture for scheduling workflow applications in the cloud," in *Web Services (ECOWS), 2011 Ninth IEEE European Conference on*, 2011, pp. 83-90, 2011.
- [119] A. Bala and I. Chana, "Autonomic fault tolerant scheduling approach for scientific workflows in Cloud

- computing," *Concurrent Engineering*, vol. 23, pp. 27-39, 2015.
- [120] X. Tang, K. Li, and G. Liao, "An effective reliability-driven technique of allocating tasks on heterogeneous cluster systems," *Cluster computing*, vol. 17, pp. 1413-1425, 2014.
- [121] L. Singh and S. Singh, "Deadline and cost based ant colony optimization algorithm for scheduling workflow applications in hybrid cloud," *Journal of Scientific & Engineering Research*, vol. 5, pp. 1417-1420, 2014.
- [122] Q.-M. Kang, H. He, H.-M. Song, and R. Deng, "Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization," *Journal of Systems and Software*, vol. 83, pp. 2165-2174, 2010.