

## Hybrid Whale Optimization Algorithm for Resource Optimization in Cloud E-Healthcare Applications

Punit Gupta<sup>1</sup>, Sanjit Bhagat<sup>2</sup>, Dinesh Kumar Saini<sup>1,\*</sup>, Ashish Kumar<sup>2</sup>, Mohammad Alahmadi<sup>3</sup> and Prakash Chandra Sharma<sup>1</sup>

<sup>1</sup>School of Computing & Information Technology, Manipal University Jaipur, Jaipur, 303007, India

<sup>2</sup>School of Basic Sciences, Manipal University Jaipur, Jaipur, 303007, India

<sup>3</sup>Department of Software Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah, 23218, Saudi Arabia

\*Corresponding Author: Dinesh Kumar Saini. Email: dkssohar@gmail.com

Received: 26 August 2021; Accepted: 01 December 2021

**Abstract:** In the next generation of computing environment e-health care services depend on cloud services. The Cloud computing environment provides a real-time computing environment for e-health care applications. But these services generate a huge number of computational tasks, real-time computing and comes with a deadline, so conventional cloud optimization models cannot fulfil the task in the least time and within the deadline. To overcome this issue many resource optimization meta-heuristic models are been proposed but these models cannot find a global best solution to complete the task in the least time and manage utilization with the least simulation time. In order to overcome existing issues, an artificial neural-inspired whale optimization is proposed to provide a reliable solution for healthcare applications. In this work, two models are proposed one for reliability estimation and the other is based on whale optimization technique and neural network-based binary classifier. The predictive model enhances the quality of service using performance metrics, makespan, least average task completion time, resource usages cost and utilization of the system. From results as compared to existing algorithms the proposed ANN-WHO algorithms prove to improve the average start time by 29.3%, average finish time by 29.5% and utilization by 11%.

**Keywords:** Cloud computing; whale optimization; health care; resource optimization

### 1 Introduction

Internet-based computing puts the demand of the service-oriented computing paradigm. Among the developers, businesses and end-users cloud computing has become popular providing storage, computing and network resources. In the global business world, cloud computing provides an optimal assignment of cloud resources [1]. Resources are provided to the various category of users using pricing models pay-as-you-go model and subscription model [2,3]. The heterogeneous virtualized



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

resources shared pool handles the user requests using a set of virtual machines [4,5]. Cloud computing supports the features of the market-oriented computing paradigm which provides optimal resource utilization with minimum operational cost. The operational cost and time play a prominent role in the performance evaluation and analysis process [6,7]. The provisioning tasks on virtual machines perform using static, dynamic and meta-heuristic approaches. Broad classification of scheduling in a scalable cloud environment performs virtual machine level. First category include user request mapping on virtual machine and second category include virtual machine mapping on host machine [8,9]. The primary focus of the scheduling of tasks on a virtual machine and virtual machine on a host improves the performance evaluation parameters time and cos. The optimal assignment of the tasks-vm and vm-host mapping comes under the category of Np-hard problem. Hence when the size of the universal solution space increases the optimal solution may converge slowly and exponential time is required for the optimal solution findings. The meta-heuristic techniques provide better results than simple static, dynamic approaches which provide the solution using polynomial time for the completion of the tasks execution and virtual machine execution. The meta-heuristic techniques Genetic, ACO, and PSO and their variations are used for effective mapping of virtual machine level [10–13]. The optimal solution depends on the fitness values of the individual schedule of the evolutionary technique. Meta-heuristic techniques follow the features of the biological and nature-inspired methodologies [14]. The optimization technique whale optimization algorithm (WOA) is a meta-heuristic technique that supports the global search capability. It plays a prominent role in the domain of science and engineering.

The healthcare sector is one of the highest data-generated industries. Due to advanced medical facilities life expectancy of a human is rapidly increased. The increased population aging required additional resources and diverse medical facilities. A large volume of data on patients is generated daily that plays a vital role in decision making to propose treatment strategies for patients. To store and analyze it some innovative and cost-effective strategies are required to help the healthcare sector to meet our challenges at the global level. Cloud computing is one such cost-effective method that helps in real-time data collection, storage, and exchange of data. Cloud infrastructure is used for high-volume storage and efficient data analysis. It becomes necessary to transit paper-based services of the health sector on the cloud using information technology. Though, cloud infrastructure is very effective, but security, reliability and privacy are prominent concerns associated with the use of the cloud in the health sector. By ensuring the reliability and security of cloud infrastructure cloud computing can play a centerpiece role in the transformation of paper records to modern information technology tools. This transformation can establish a better relationship between doctors and patients, as well as helps in better decision making. So, presently scientists should focus on ensuring the reliability of cloud infrastructure in the health sector. Up to the best of our knowledge, reliability aspects of the cloud in the health sector are not so much explored. Some work on the reliability of cloud and computing networks is available in the literature. The main issue with cloud healthcare applications is real-time processing and the need for a highly reliable infrastructure to complete the task within a deadline with the least processing time. So to achieve there exist many solutions but the solution like genetic and interactive algorithms have high simulation time/searching time to find the best solution. So in order to remove this, an artificial neural-based optimization is proposed to find the best solution in constant time by training the machine learning model once with the best solution from the Whale algorithm. Here whale algorithm is proposed because it has a high probability to find the best solution in the least time as compare ACO, PSO, Bat and genetic algorithm as shown in results.

In this manuscript, our focus is to develop a reliability model for cloud infrastructure as well as proposing a hybrid approach ANN-WOA technique that provides an optimal solution using

performance metrics time and resource utilization. The performance evaluation process includes the training process and validation. The whale optimization technique provides the training data set of the ANN-WOA model. The major contribution of the proposed ANN-WOA model and reliability model summarizes as follows.

- 1) The tasks execution on virtual machine enhances using the proposed ANN-WOA, Our focus includes to introduce a neural inspired whale optimization model for task scheduling and apply the hybrid model to solve the virtual machine level scheduling problem.
- 2) The proposed ANN-WOA focuses on efficient task scheduling by incorporating advanced optimization strategies to improve both the convergence speed and accuracy of the whale optimization approach.
- 3) The detailed design and implementation of neural inspired whale optimization technique described and compare it against existing meta-heuristics technique Genetic, and BB-BC cost. Our results and discussions section demonstrates that hybrid technique ANN-WOA outperforms using resource utilization in a scalable cloud environment.
- 4) The proposed model focuses on the identification of the sensitive component of cloud infrastructure by estimating the reliability of the cloud and measuring the effect of variation in failure and repair rates. The proposed model improves the performance of task and cloud infrastructure both at the same time.

The rest of this manuscript is organized as follows. In Section 2 covers the related works. Section 3 includes the motivations of task scheduling optimization. Section 4 presents the proposed ANN-WOA approach and its implementation details. In Section 5, incorporates a reliability model of cloud infrastructure. The performance evaluation and analysis includes in Section 6, and finally, Section 7 concludes the work.

## 2 Related Work

Resource assignment to the tasks in a cloud computing environment is a challenging concern. It is complex problem of assignment of storage, computing and network resources. The datacenter provides heterogeneous resources to the users in a scalable cloud environment. Optimal assignment of tasks on virtual machines follows the various static, dynamic and meta-heuristic techniques. The researchers have been proposed the techniques using multiple performance evaluation criteria. In exiting work [15–19] author has showcased a study on the issue in cloud computing and existing work to solve those issues. Malawski et al. [20] focused on performance metrics deadline of the tasks and the cost of the resources used in a hybrid cloud scenario. Su et al. presented a task scheduling in a cloud environment using performance metrics cost of the storage, computing and network resources across the globe. The presented approach follows the features of the Pareto dominance [21]. Researchers focused on cluster base scheduling and list base scheduling approaches DSC [22] and list scheduling algorithm (e.g., DSL [23]). The meta-heuristic approaches have wide application in a cloud computing system. Still, there may be low quality of service due to an exponential increase in the number of user requests [24]. Various meta-heuristic techniques follow the features of nature-inspired and bio-inspired techniques. The meta-heuristic techniques include Genetic awareness, ACO, and PSO. Aziza et al. presented the bio-inspired Genetic time shared and space shared techniques. The proposed technique enhances the performance using evaluation metrics makespan and cost [25]. Li et al. demonstrated ACO base tasks scheduling using load balancing in a cloud computing environment [26]. Wang et al. presented an enhanced PSO technique for the optimal placement of the virtual machine. The performance is evaluated using performance metric power savings and quality of service parameters

of Global optimization [27]. The performance can be improved using a hybrid technique that follows the features of multiple nature-inspired techniques. Chen et al. [28] proposed swarm optimization and ant colony optimization-based hybrid tasks scheduling techniques which focused on performance metrics makespan of the tasks on a virtual machine. Liu et al. [29] demonstrated the bio-inspired technique which follows the features of the bio-inspired technique which provides a Global optimal solution. Tsai et al. [30] proposed a hyper-heuristic technique based on Genetic, PSO, and ACO techniques. Performance is evaluated using makespan. Further, the efficiency can be improved using whale optimization technique of task scheduling on a virtual machine. The multi-objective model provides better performance than simple meta-heuristic techniques. The multi-objective performance metrics in a scalable cloud computing environment include time, cost and power savings [31]. Ramezani et al. [32] multi-objective particle swarm optimization technique which provides the better quality of service using performance metrics execution time and cost paid in the duration of execution time. Mao et al. [33] focused on performance metrics stability of the cloud computing system and time. Liu et al. [34] presented a framework that focused on the performance metrics response time and resource utilization of cloud environments. The reliability modeling and estimation of communication networks is very vital and attracted the researcher [35]. Various redundancy strategies have been utilized in designing a cloud infrastructure [36]. As redundancy is very helpful in increasing the efficiency of cloud infrastructure. The performance of big data applications is also assessed using concurrent modeling [37]. Reliability estimation of cloud service-based applications is also done using SRGM and NMSPN methodologies [38]. Various subsystems of cloud infrastructure were also discussed as a reliability point of view [39]. The semi-Markovian approach is utilized for the availability assessment of the cloud having multiple pools of physical and virtual equipment [40].

The concepts of switching failure, common cause failure and vacations along with warm standby units in computer networks have been used extensively [41]. The technique for the development of a stochastic model of cloud infrastructure is discussed in the literature [42]. End-to-end performance analysis in a cloud using stochastic models has been done [43]. The 80/20 rule has been suggested to improve the reliability of cloud-based data centers [44]. Markovian approach applied in fuzzy reliability investigation of data center [45]. Hence meta-heuristic techniques are widely used in multi-objective resource optimization in a cloud environment as well as Markovian is used in reliability evaluation. In this work, our key focus will be to proposed ANN-WOA model which follows the features of multi-objective intelligent meta-heuristic technique and reliability investigation of cloud infrastructure. Our objective is to improve the performance of the state-of-the-art method of task scheduling in the cloud computing environment. This is best of our knowledge to propose a neural whale optimization model which will improve the performance of the cloud datacenter resources using the optimal placement of tasks on a virtual machine.

### 3 Motivation for Work

The key focuses are to improve the quality of service using enhanced meta-heuristic techniques and investigate the reliability of cloud infrastructure. The existing meta-heuristic technique and static approaches include the performance metrics time, cost and power savings. Our objective is to improve the research gaps in existing approaches. The major focus of the researchers includes time and resource utilization cost. The state of art methodologies provides the direction to improve the performance of the static, dynamic and meta-heuristic techniques using multi-objective performance evaluation metrics. The proposed model will integrate the whale optimization meta-heuristic technique with a neural network. The artificial neural network-inspired whale optimization (ANN-WOA) technique will focus on multi-objective criteria i.e., time, cost, power, and service level agreement.

The performance evaluation and analysis parameters are shown in the Tab. 1 are evaluated using real-time workload and fabricated workload. The proposed model ANN-WOA will be validated against the static, dynamic and meta-heuristic techniques, GA-Cost, GA-Exe, Max-Min, and PSO respectively. The state of art methods and proposed ANN-WOA will be implemented using the integrated environment of cloudsim 4.0 and NetBeans IDE 8.0.1. The results and discussions section will show the comparisons of the simulation results using different user requests and population sizes.

#### 4 Proposed Model

The proposed model is based on the hybrid technique ANN-WOA. It includes the whale optimizer to prepare the training data sets. The training data sets includes (ti, vmj) pair of tasks mapping on the virtual machine using optimal objective parameters. The output of the whale optimizer acts as input parameters to the neural classifier which predicts the virtual machine identity for the allocation of the tasks on virtual machines with an optimal solution. The data set is prepared using the whale optimization technique. The prepared data sets is divided into training and testing data sets for the training and testing of the neural network-based binary classifier which ensures the selection of an appropriate virtual machine for the assignment of the tasks. The preparation of the data set depends on the input parameters of the whale optimizer. The ANN-WOA model assures the high quality of service in a complex cloud computing environment having  $n \gg m$  i.e., number of tasks assigned to the limited number of virtual machines in a scalable cloud computing environment. The ANN whale optimization technique provides an optimal assignment of the tasks on virtual machines in a scalable cloud computing environment.

The following steps are used in the case of ANN-WOA model.

##### 4.1 Initialization of the Parameters

In this phase the parameters of the whale optimizer, neural network-based binary classifier initialize. Once the parameters initialize the whale optimizer provides an optimal set of the tasks and associated virtual machine decisions.

##### 4.2 Dataset Preparation for Training Purpose

This subsection covers the allocation of the tasks on virtual machines using the nature-inspired meta-heuristic whale optimizer technique. This whale optimizer technique provides data sets that are used for training and testing purposes. This subsection also covers all the steps used in this data set preparation for training purposes.

###### 4.2.1 Initialization

The initialization step generates the random population of the search agent. The search agents provide the solution space which includes multiple local optimal search agents and may include global optimal search agents.

###### 4.2.2 Calculation of Fitness Function

Step 2 calculates the fitness function values using the following equations.

$$f = 1/\text{distance} \quad (1)$$

$$distance = cloudletLength / MIPS \quad (2)$$

The fitness values evaluate for all the search agents and select the Global optimal agent with the fittest function value.

#### 4.2.3 Encircling Prey

The agent becomes aware of the prey and the agent considers itself as an optimal agent i.e., according to the position of the humpback whale (acts as an optimal agent) other solution agents update their position. The parameters associated with the agent acts as input parameters to the whale-inspired scheduler. The input parameters include tasks count, virtual machine count and the position vector of the best agent, position vector of the agents in the search space, coefficient vector, and search vector respectively. The encircling pray step of the whale optimizer includes search agent position according to the best search agent position as shown in Eq. (3).

$$\vec{s} = \vec{Q} \bullet \vec{p}^*(x) - \vec{p}(x) \quad (3)$$

where  $\vec{s}$  presents the position of the search agent. The variable  $x$  represents the current iteration.  $\vec{p}$  presents the position vector and  $\vec{p}^*$  presents the position vector of the best agent in the solution space, and  $\vec{Q}$  shows the coefficient vector. The new position of the search agent updates using Eq. (4).

$$\vec{p}(x+1) = \vec{p}^*(x) - \vec{M} \bullet \vec{s} \quad (4)$$

where  $\vec{M}$  represents the coefficient vector, and operator  $\bullet$  indicates the element-wise multiplication. The coefficient vector  $\vec{Q}$ ,  $\vec{M}$  calculated using Eqs. (5) and (6) respectively.

$$\vec{M} = 2\vec{l} \cdot \vec{k} - \vec{l} \quad (5)$$

$$\vec{Q} = 2 \cdot \vec{k} \quad (6)$$

where  $l$  varies from 2 to 0, and random vector  $k \rightarrow [0,1]$ .

#### 4.2.4 Exploitation and Exploration

The exploitation phase includes shrinking the encircling mechanism and agent position updating in a spiral manner. In the exploration phase the position of the search agent updates in a random fashion.

#### 4.2.5 Termination

The data set prepared used for the training process of the Binary classifier for the prediction of the appropriate virtual machine identity. It follows the iterations of the steps used to obtain an optimal solution.

*Algorithm of the Proposed Resource Provisioning Technique WOA*

**Input** data, max\_iteration, population

**Initialize** whales population  $X_i$  ( $i = 1, 2, 3 \dots \dots n$ )

**Initialize** A, C, l, a and p

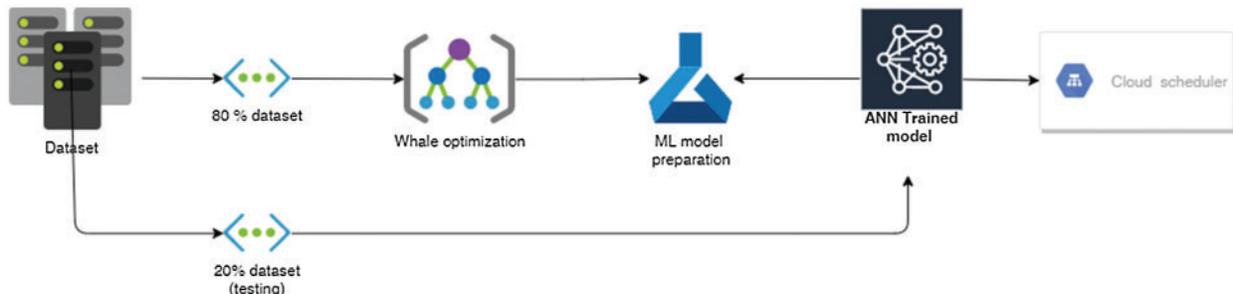
**Calculate** the fitness of each search agent

```

X* = best agent at the beginning
While (It < max_iteration)
    for each search_agent
        if (p < 0.5)
            if (|A| < 1)
                update search agent with  $\vec{p}(x + 1) = \vec{p}^*(x) - \vec{M} \bullet \vec{s}$ 
            else if (|A| >= 1)
                select random search agent (P_rand)
                update the current search agent
            end
        else if (p >= 0.5)
            update the position of current search agent
    end
Calculate fitness of each search agent  $distance = cloudletLength/MIPS$ 
Update X* if better agent found
It = It + 1
Update A, C, l, a and p
end while
return X*
    
```

**4.3 Design of Neural Computing Model**

The initialization parameters and the output of the whale optimizer act as an input to the multi-layer neural network classifier. The multi-layer classifier configuration contains the input layer, hidden layers and output layer respectively. The number of hidden layers, number input parameters, and the number of output parameters depends on the number of virtual machines and the number of submitted tasks on the virtual machine. Fig. 1 exhibits the block diagram of the presented ANN-WOA model which includes a detailed description of the multi-layer perceptron and whale optimization technique that generates the scheduling decision of the tasks on virtual machines. The output of the nature-inspired technique acts as an input (X1, X2) to the neural network classifier. The output of the ANN-WOA provides the virtual machine identity of allocation to tasks with optimal performance metrics time and cost. Fig. 1 shows the proposed model using whale optimization and ML model.



**Figure 1:** Proposed solution architecture

#### 4.4 Training of the Neural Model (Using Data Set Prepared Using WOA)

The scheduling decisions of the WOA provide the data sets which is used for the training of the neural network model. The training process trains the model using free parameters, hyperparameters and input parameters of the ANN-WOA model. The prepared data sets divide into two parts 80% data set used for the training of the model shown in the block diagram and 20% data set to use for the testing of the training model using variations of iterations.

The training process includes the following steps

1. Initialization of the bias parameter and connection weights.
2. Training the model using data sets generated using whale optimization base scheduling in cloud computing
3. Validation must be performed using 20% data sets for the accuracy test and efficiency improvement of the proposed model.
4. Once the model is validated then the network runs to predict the values of the expected outputs. The linear and non-linear activation functions use for the activation of the neurons in a scalable cloud computing environment. The complexity of the neural network classifier i.e., number of hidden layers and types of activation functions depends on the complexity of the datacenter node. Hence training step plays a prominent role in a scalable cloud computing environment.

#### 4.5 Accuracy Improvement Using Error Backpropagation and Correction

In this phase, our objective is to mitigate the difference between the desired output and target output. Hence in this supervised learning process, the error difference must be minimum. Smaller the difference between target and desired output better the accuracy of the model in a complex, scalable distributed data center network environment.

$$e_k(n) = d_k(n) - y_k(n) \quad (7)$$

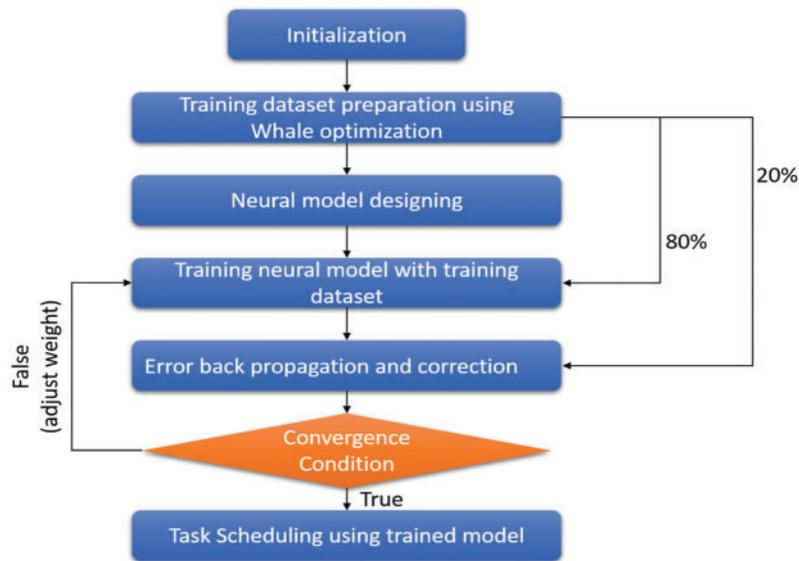
$$\delta_k = \frac{1}{2} * e_k^2(n) \quad (8)$$

In Eqs. (7) and (8), the parameter  $d_k(n)$  denotes the desired output,  $y_k(n)$  denotes the calculated output, and  $e_k(n)$  is the error, and  $\delta_k$  is the cost function used in the learning process of the ANN model respectively.

#### 4.6 Tasks Assignment on Virtual Machine Decision

This phase covers the assignment of the tasks on virtual machines using ANN-WOA model. The trained model provides an accuracy, correctness of the predictive outcomes in a complex scalable cloud environment.

Hence using the WOA-ANN model predict an output which includes the solution array of the scheduling assignment of the tasks on an appropriate virtual machine using multiple epochs and test the accuracy of the model using independent variable test values. The target outcomes and desired outcomes measure the level of accuracy. The quality of service of the model ensures using makespan and resource utilization cost which depends on utilization duration of the IaaS, PaaS and SaaS service delivery architecture. Fig. 2 shows the flow diagram of the proposed algorithm using the whale algorithm and artificial neural network.



**Figure 2:** Flow diagram

## 5 Reliability Modeling of Cloud Infrastructure

In the last few years cloud infrastructure becomes very popular in MNC's. In the cloud customers have the freedom to manage applications, data, and operating systems. Many services like networking, security, billing, recovery, and load balancing are managed by cloud services. But all these tasks can be effectively performed if cloud infrastructure work with high reliability. As cloud infrastructure is a very complex structure involving various physical and virtual components and it is very difficult to attain desired performance. It can be only achieved if it is operated with high reliability. For this purpose, here a stochastic model is proposed to attain the highest reliability of cloud infrastructure.

### 5.1 Assumptions

In the development of the model, the following assumptions have been utilized:

- All the subsystems arranged in a series configuration, i.e., failure of one cause complete cloud failure
- No provision of simultaneous failure
- Perfect repair and switch devices
- All failure and repair rates are exponentially distributed and independent of each other
- No correlated failure

### 5.2 Notations

The following assumptions shown in [Tab. 1.1](#) has been made to develop the model for cloud infrastructure: for cloud infrastructure:

**Table 1.1:** Assumptions for cloud model

S. No.	Sub-system	Operative model	Failure mode		Failure rate		Repair rate	
			H/w	S/w	H/w	S/w	H/w	S/w
1	Client	A	a <sub>h</sub>	a <sub>s</sub>	τ <sub>2</sub>	τ <sub>1</sub>	μ <sub>2</sub>	μ <sub>1</sub>
2	Database	B	b <sub>h</sub>	b <sub>s</sub>	τ <sub>9</sub>	τ <sub>8</sub>	μ <sub>9</sub>	μ <sub>8</sub>
3	Standby database	B1	b <sub>h</sub>	b <sub>s</sub>	τ <sub>9</sub>	τ <sub>8</sub>	μ <sub>9</sub>	μ <sub>8</sub>
4	Cloud	C	c <sub>h</sub>	c <sub>s</sub>	τ <sub>4</sub>	τ <sub>3</sub>	μ <sub>4</sub>	μ <sub>3</sub>
5	Service	D	—	d <sub>s</sub>	...	τ <sub>5</sub>	...	μ <sub>5</sub>
6	Network	E	e <sub>h</sub>	e <sub>s</sub>	τ <sub>7</sub>	τ <sub>6</sub>	μ <sub>7</sub>	μ <sub>6</sub>
	Load failure		CF			τ <sub>10</sub>	μ <sub>10</sub>	
$P_i(t)$	Probability that at time t system is in i <sup>th</sup> state							
$P'_i(t)$	Derivate of $P_i(t)$							

**5.3 System Description**

The cloud infrastructure is very complex having several physical and virtual components. Broadly, it has five subsystems namely clients, databases, cloud, service, and network. The clients, databases, cloud, and network may suffer due to hardware and software failures while the service facility suffers only due to snags in software. The complete system may collapse due to overload on the cloud and the whole cloud infrastructure causes failure. The hardware failures are rectified by the repairman and software undergoes rejuvenation upon failure. After all types of repair activities system work as good as new. All failure and repair rates are exponentially distributed.

**5.4 Stochastic Model Development and Analysis**

In this section, a stochastic model is developed and analyzed using Runge-Kutta method of 4th order. The Markovian birth-death process is applied to develop C-K differential-difference equations. The simulated results are obtained using MATLAB R2019a. The expression of the stochastic model is as follows:

$$P_1'(t) = -\sum_{i=1}^{10} \tau_i P_1(t) + \mu_1 P_4(t) + \mu_2 P_5(t) + \mu_3 P_6(t) + \mu_4 P_7(t) + \mu_5 P_9(t) + \mu_6 P_{10}(t) + \mu_7 P_{11}(t) + \mu_8 P_2(t) + \mu_9 P_3(t) + \mu_{10} P_8(t) \tag{9}$$

$$P_2'(t) = -[\mu_8 + \sum_{i=1}^{10} \tau_i] P_2(t) + \tau_8 P_1(t) + \mu_5 P_{12}(t) + \mu_2 P_{13}(t) + \mu_1 P_{14}(t) + \mu_3 P_{15}(t) + \mu_4 P_{16}(t) + \mu_{10} P_{17}(t) + \mu_8 P_{18}(t) + \mu_7 P_{19}(t) + \mu_6 P_{20}(t) + \mu_9 P_{21}(t) \tag{10}$$

$$P_3'(t) = -[\mu_9 + \sum_{i=1}^{10} \tau_i] P_3(t) + \mu_1 P_{31}(t) + \mu_2 P_{30}(t) + \mu_{10} P_{29}(t) + \mu_6 P_{28}(t) + \mu_8 P_{27}(t) + \mu_7 P_{25}(t) + \mu_9 P_{26}(t) + \mu_5 P_{24}(t) + \mu_4 P_{23}(t) + \mu_3 P_{22}(t) + \tau_9 P_1(t) \tag{11}$$

$$P_4'(t) = -\mu_1 P_4(t) + \tau_1 P_1(t) \quad (12)$$

$$P_5'(t) = -\mu_2 P_5(t) + \tau_2 P_1(t) \quad (13)$$

$$P_6'(t) = -\mu_3 P_6(t) + \tau_3 P_1(t) \quad (14)$$

$$P_7'(t) = -\mu_4 P_7(t) + \tau_4 P_1(t) \quad (15)$$

$$P_8'(t) = -\mu_{10} P_8(t) + \tau_{10} P_1(t) \quad (16)$$

$$P_9'(t) = -\mu_5 P_9(t) + \tau_5 P_1(t) \quad (17)$$

$$P_{10}'(t) = -\mu_6 P_8(t) + \tau_6 P_1(t) \quad (18)$$

$$P_{11}'(t) = -\mu_7 P_{11}(t) + \tau_7 P_1(t) \quad (19)$$

$$P_{12}'(t) = -\mu_2 P_{13}(t) + \tau_2 P_2(t) \quad (20)$$

$$P_{13}'(t) = -\mu_2 P_{13}(t) + \tau_2 P_2(t) \quad (21)$$

$$P_{14}'(t) = -\mu_1 P_{14}(t) + \tau_1 P_2(t) \quad (22)$$

$$P_{15}'(t) = -\mu_1 P_{15}(t) + \tau_3 P_2(t) \quad (23)$$

$$P_{16}'(t) = -\mu_4 P_{16}(t) + \tau_4 P_2(t) \quad (24)$$

$$P_{17}'(t) = -\mu_1 0 P_{17}(t) + \tau_1 0 P_2(t) \quad (25)$$

$$P_{18}'(t) = -\mu_8 P_{18}(t) + \tau_8 P_2(t) \quad (26)$$

$$P_{19}'(t) = -\mu_7 P_{19}(t) + \tau_7 P_2(t) \quad (27)$$

$$P_{20}'(t) = -\mu_6 P_{20}(t) + \tau_6 P_2(t) \quad (28)$$

$$P_{21}'(t) = -\mu_9 P_{21}(t) + \tau_9 P_2(t) \quad (29)$$

$$P_{22}'(t) = -\mu_3 P_{22}(t) + \tau_3 P_3(t) \quad (30)$$

$$P_{23}'(t) = -\mu_4 P_{23}(t) + \tau_4 P_3(t) \quad (31)$$

$$P_{24}'(t) = -\mu_5 P_{24}(t) + \tau_5 P_3(t) \quad (32)$$

$$P_{25}'(t) = -\mu_7 P_{25}(t) + \tau_7 P_3(t) \quad (33)$$

$$P_{26}'(t) = -\mu_9 P_{26}(t) + \tau_9 P_3(t) \quad (34)$$

$$P_{27}'(t) = -\mu_8 P_{27}(t) + \tau_8 P_3(t) \quad (35)$$

$$P_{28}'(t) = -\mu_6 P_{28}(t) + \tau_6 P_3(t) \quad (36)$$

$$P_{29}'(t) = -\mu_{10} P_{29}(t) + \tau_{10} P_3(t) \quad (37)$$

$$P_{30}'(t) = -\mu_2 P_{30}(t) + \tau_2 P_3(t) \quad (38)$$

$$P_{31}'(t) = -\mu_1 P_{31}(t) + \tau_1 P_3(t) \quad (39)$$

initial conditions

$$P_i(0) = \begin{cases} 1 & \text{if } i = 1 \\ 0 & \text{if } i \neq 1 \end{cases} \quad (40)$$

$$\text{Steady state Availability (SSA)} = P_1(t) + P_2(t) + P_3(t) \quad (41)$$

As the mathematical model (9)–(39) along with the initial condition (40) is very complex and in this situation, numerical solution is recommended instead of an analytical approach. The availability of the cloud infrastructure is given by.

## 6 Result and Discussions

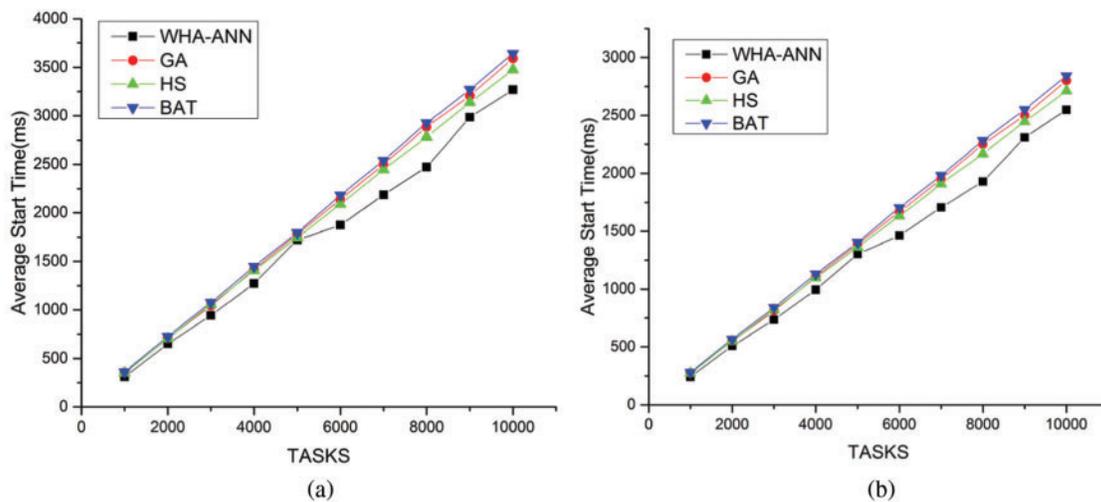
In this section, the simulation setup and result using various simulation configurations are discussed. Simulation is performed using Cloudsim 3.0, The simulation uses workload traces for real-time task simulation which is an SWF format workload file from parallel workload, a free open-source dataset from parallel workload repository. Tab. 1.2 shows the configuration of tasks with its size varying from 200 instructions to 40000 Instruction. Where the instructions are of small, moderate and large size.

The simulation is carried out with ANN model with 3 hidden layers and leaky Relu function as smoothening function at output layer and hidden layers. The number of neurons in each hidden layer is 50. In Figs. 3a and 3b shows a comparison of the average start time of task with 5 and 10 virtual machines in 4 datacenters. The results show the proposed algorithm provides better performance and the least average start time with increasing task load as compared to existing approaches. This shows that the waiting time in queue for the task in the least for most of the tasks. Figs. 4a and 4b discusses a comparison of the average finish time of task with 5 and 10 virtual machines in 4 datacenters. The

experiment shows that the proposed algorithm completes tasks with the least average finish time with increasing tasks in different scenarios. This shows that most of the tasks are allocated to resources with high computation performance which can complete the task in the least time within the deadline.

**Table 1.2:** Configuration parameters of user tasks

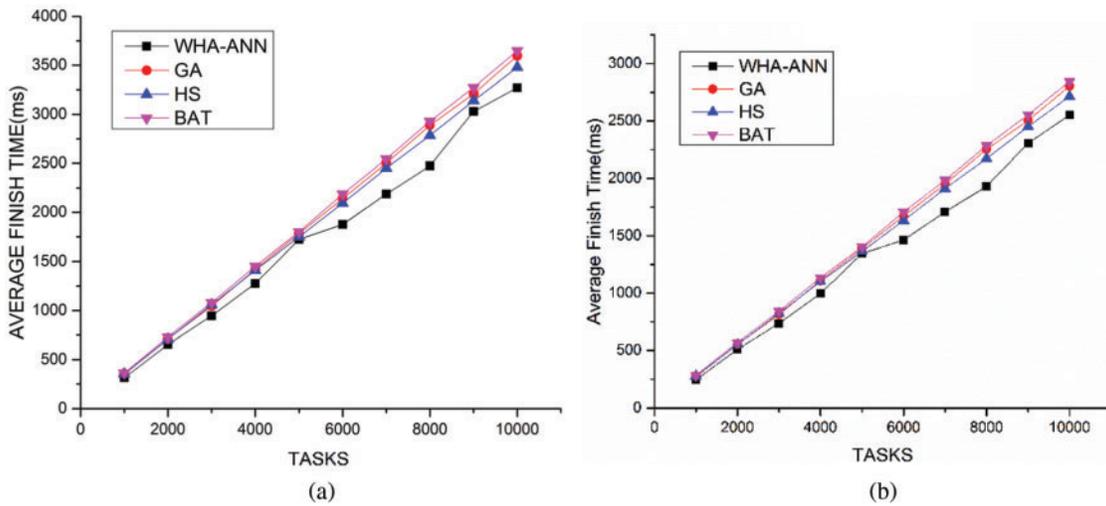
Task parameters	Values
Task length	200–40000
Input size	300 Byte
Output size	300 Byte
PE	1/2



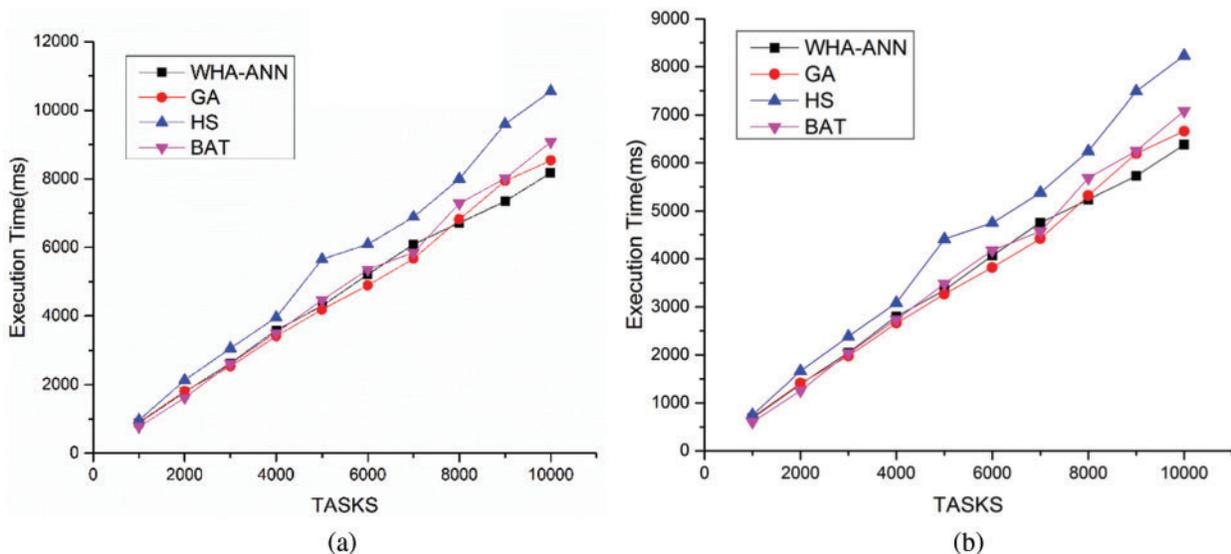
**Figure 3:** (a) Comparison average start time vs. number of tasks for 5 VM (b) Comparison average start time vs. number of tasks for 5 VM

Figs. 5a and 5b discusses a comparison of the execution time of tasks with 4 datacenters. The results show the proposed algorithm provides the least execution time with scaling tasks and virtual machines. For less number of tasks, the proposed algorithm proves to provide similar to existing best meta-heuristic algorithms like genetic algorithm, harmony search and bat-inspired algorithms.

Figs. 6a and 6b discusses a comparison of the change in utilization of with 5 and 10 virtual machines in 4 datacenters. The results show the proposed algorithm provides better performance and high utilization with increasing task load. This shows the proposed algorithm provides better resource management by utilizing the resources and high performance. This also shows that there is a low probability of hot spots in the cloud infrastructure. Where power is directly promotional to the utilization of the server. Better the utilization least the power consumption of the system, in existing systems there exist host spots which are highly loaded which consume more power lever other consuming power without been utilized.



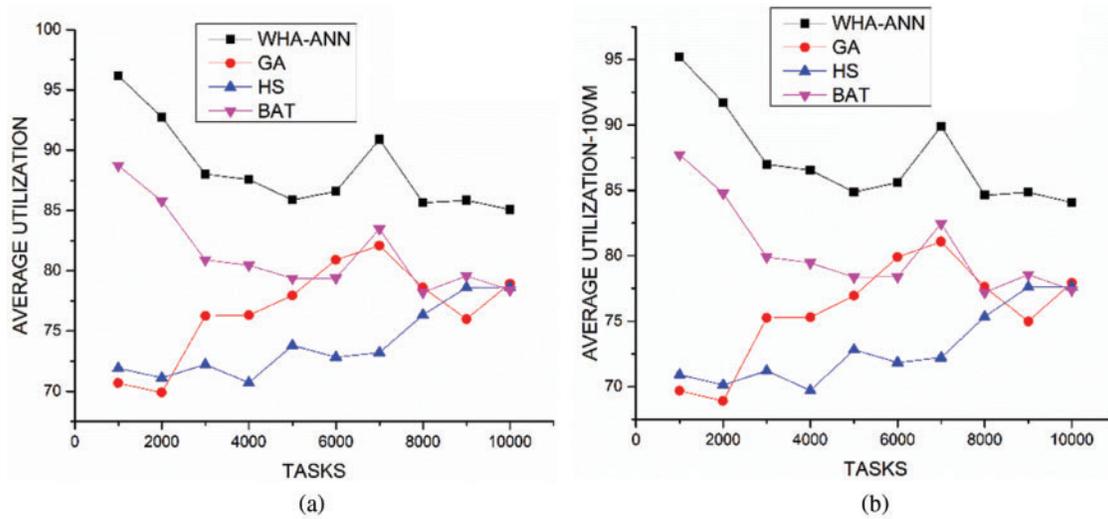
**Figure 4:** (a) Comparison average finish time vs. number of tasks for 5 VM (b) Comparison average finish time vs. number of tasks for 5 VM



**Figure 5:** (a) Comparison execution time vs. number of tasks for 5 VM (b) Comparison execution time vs. number of tasks for 5 VM

The simulated results of steady state availability were obtained for a particular set of parametric values with respect to time. The base values are as follows:  $\tau_1 = 0.0002$ ,  $\tau_2 = 0.0006$ ,  $\tau_3 = 0.0003$ ,  $\tau_4 = 0.0001$ ,  $\tau_5 = 0.002$ ,  $\tau_6 = 0.00025$ ,  $\tau_7 = 0.017$ ,  $\tau_8 = 0.0018$ ,  $\tau_9 = 0.021$ ,  $\tau_{10} = 0.0038$ ,  $\mu_1 = 0.221$ ,  $\mu_2 = 0.181$ ,  $\mu_3 = 0.451$ ,  $\mu_4 = 0.204$ ,  $\mu_5 = 0.218$ ,  $\mu_6 = 0.458$ ,  $\mu_7 = 0.259$ ,  $\mu_8 = 0.456$ ,  $\mu_9 = 0.318$ ,  $\mu_{10} = 0.318$ . From Tab. 2, it is revealed that SSA decreases with time, h/w failure rate and software failure rate. At time  $t = 20$ , initially cloud infrastructure availability is 0.911775 while after making 100% variation in the failure rate of hardware and software of client, it is observed that availability more rapidly declined in case of software failure. The 100% variation in h/w repair and software rejuvenation shows that

software rejuvenation is more sensitive than hardware repair. Tab. 2 shows that hardware failure and repair rates of the cloud are more sensitive instead of software failure and rejuvenation. Tab. 2 depicted that software failures of services and load failure of cloud infrastructure highly influence the SSA of the cloud infrastructure. Tab. 2 revealed that software failure in network and software rejuvenation sharply decline of the incline the availability of the system. Tab. 2 revealed that software failure in databases decreases the SSA of cloud infrastructure.



**Figure 6:** (a) Comparison average utilization vs. number of tasks for 5 VM (b) Comparison average utilization vs. number of tasks for 5 VM

**Table 2:** Analysis of impact on failure and repair rates

	Average base values	Average H/w failure	Average S/w failure	Average H/w repair	Average S/w rejuvenation
Impact of client's failure and repair rates	0.9114029	0.910652	0.908663	0.911778	0.9127762
Impact of cloud's failure and repair rates	0.9114029	0.910851	0.910996	0.911679	0.9116063
Impact of service & load failure and repair rates	0.9114029	0.903851	0.901584	0.915223	0.9163935
Impact of network's failure and repair rates	0.9114029	0.91095	0.859967	0.91163	0.9394854

(Continued)

**Table 2:** Continued

	Average base values	Average H/w failure	Average S/w failure	Average H/w repair	Average S/w rejuvenation
Impact of database's failure and repair rates	0.9114029	0.910978	0.901793	0.911608	0.9141128

## 7 Conclusions and Future Directions

This work presented a whale optimization and neural-inspired virtual machine assignment to the user requests approach based on the features of the human brain and optimization method for healthcare cloud applications. The proposed algorithm tries to improve the fault tolerance of the cloud environment and managing to improve the performance of the system at the same time using machine learning techniques. The results and discussions section illustrates that WHA-ANN performs better than existing meta-heuristic techniques. The study of the proposed algorithm is performed with scaling task load and resources. The proposed stochastic model revealed that software failure of network subsystem is most sensitive and needs additional care for operation. In the future work may focus on virtual machine allocation in a data center network. The drawback of the proposed algorithm is that the neural network needs to be trained after some time with updated upcoming tasks to overcome the decay of the trained model. This training time will although be much lesser than the total sum of the simulation time of the existing algorithm. In the future, the algorithm may be tried to improve the power consumption and scalability in the cloud environment. In future work, the idea of arbitrary failure and repair rates and multiple failures can be incorporated to enhance the availability of the cloud infrastructure.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] A. N. Toosi, R. N. Calheiros and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, pp. 1–47, 2014.
- [2] M. Cusumano, "Cloud computing and SaaS as new computing platforms," *Communications of the ACM*, vol. 53, no. 4, pp. 27–29, 2010.
- [3] S. K. Gavvala, C. Jatoth, G. Gangadharan and R. Buyya, "QoS-Aware cloud service composition using eagle strategy," *Future Generation Computer Systems*, vol. 90, pp. 273–290, 2019.
- [4] P. Kaur and S. Mehta, "Resource provisioning and work flow scheduling in clouds using augmented shuffled frog leaping algorithm," *Journal of Parallel and Distributed Computing*, vol. 101, pp. 41–50, 2017.
- [5] Z. Li, H. Shen and C. Miles, "PageRankVM: A pagerank based algorithm with anti-collocation constraints for virtual machine placement in cloud datacenters," in *Proc. of the IEEE 38th Int. Conf. on Distributed Computing Systems (ICDCS)*, Vienna, Austria, pp. 634–644, 2018.
- [6] A. Michael, A. Fox, R. Griffith, A. D. Joseph, R. Katz *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 4, pp. 50–58, 2010.

- [7] H. Morshedlou and M. R. Meybodi, "Decreasing impact of SLA violations: A proactive resource allocation approach for cloud computing environments," *IEEE Transactions on Cloud Computing*, vol. 2, no. 2, pp. 156–167, 2014.
- [8] B. Xu, C. Zhao, E. Hu and B. Hu, "Job scheduling algorithm based on berger model in cloud environment," *Advances in Engineering Software*, vol. 42, no. 7, pp. 419–425, 2011.
- [9] S. Kayalvili and M. Selvam, "Hybrid SFLA-GA algorithm for an optimal resource allocation in cloud," *Cluster Computing*, pp. 3165–3173, 2018.
- [10] A. Arunarani, D. Manjula and V. Sugumaran, "Task scheduling techniques in cloud computing: A literature survey," *Future Generation Computer Systems*, vol. 91, pp. 407–415, 2019.
- [11] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*, vol. 7, Hoboken, NJ, USA: Wiley, 2000.
- [12] M. Dorigo and G. D. Caro, "Ant colony optimization: A new metaheuristic," *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99*, vol. 2, pp. 1470–1477, 1999.
- [13] J. Kennedy, "Particle swarm optimization," in *Proc. of the the IEEE Int. Conf. on Neural Networks*, Springer, pp. 760–766, 2011.
- [14] W. Z. Sun, J. S. Wang and X. Wei, "An improved whale optimization algorithm based on different searching paths and perceptual disturbance," *Symmetry*, vol. 10, no. 6, 2018.
- [15] N. Anand. "Handbook of cloud computing: Basic to advance research on the concepts and design of cloud computing," BPB Publications, 2019.
- [16] A. Kaur, P. Gupta, M. Singh and A. Nayyar, "Data placement in era of cloud computing: A survey, taxonomy and open research issues," *Scalable Computing: Practice and Experience*, vol. 20, pp. 377–398, 2019.
- [17] S. Ali, Y. Hafeez, N. Z. Jhanjhi, M. Humayun, M. Imran *et al.*, "Towards pattern-based change verification framework for cloud-enabled healthcare component-based," *IEEE Access*, vol. 8, pp. 148007–148020, 2020.
- [18] A. Nayyar, D. N. Le and N. G. Nguyen, "Advances in swarm intelligence for optimizing problems in computer science," CRC Press, 2018.
- [19] P. S. Rawat, P. Dimri, P. Gupta and G. P. Saroha, "Resource provisioning in scalable cloud using bio-inspired artificial neural network model," *Applied Soft Computing*, vol. 99, pp. 106876, 2021.
- [20] M. Malawski, K. Figiela and J. Nabrzyski, "Cost minimization for computational applications on hybrid cloud infrastructures," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1786–1794, 2013.
- [21] S. Su, J. Li, Q. Huang, X. Huang, K. Shuang *et al.*, "Cost-efficient task scheduling for executing large programs in the cloud," *Parallel Computing*, vol. 39, no. 4/5, pp. 177–188, 2013.
- [22] T. Yang and A. Gerasoulis, "DSC: Scheduling parallel tasks on an unbounded number of processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 9, pp. 951–967, 1994.
- [23] G. C. Sih and E. A. Lee, "A Compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures," *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, no. 2, pp. 175–187, 1993.
- [24] C. W. Tsai and J. J. Rodrigues, "Metaheuristic scheduling for cloud: A survey," *IEEE Systems Journal*, vol. 8, no. 1, pp. 279–291, 2014.
- [25] H. Aziza and S. Krichen, "Bi-objective decision support system for task scheduling based on genetic algorithm in cloud computing," *Computin*, vol. 100, no. 2, pp. 65–91, 2018.
- [26] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [27] S. Wang, A. Zhou, C. H. Hsu, X. Xiao and F. Yang, "Provision of data-intensive services through energy- and QoS-aware virtual machine placement in national cloud data centers," *IEEE Transactions on Emerging Topics in Computing*, vol. 4, no. 2, pp. 290–300, 2016.
- [28] X. Chen and D. Long, "Task scheduling of cloud computing using integrated particle swarm algorithm and ant colony algorithm," *Cluster Computing*, vol. 22, no. 4, pp. 2761–2769, 2017.

- [29] C. Y. Liu, C. M. Zou and P. Wu, "A task scheduling algorithm based on genetic algorithm and ant colony optimization in cloud computing," in *Proc. of the 13th Int. Symp. on Distributed Computing and Applications to Business, Engineering and Science*, Hubei, China, pp. 68–72, 2014.
- [30] C. W. Tsai, W. C. Huang, M. H. Chiang, M. -C. Chiang and C. -S. Yang, "A Hyper-heuristic scheduling algorithm for cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 2, pp. 236–250, 2014.
- [31] L. Zuo, L. Shu, S. Dong, C. Zhu and T. Hara, "A Multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing," *IEEE Access*, vol. 3, pp. 2687–2699, 2015.
- [32] F. Ramezani, J. Lu and F. Hussain, "Task scheduling optimization in cloud computing applying multi-objective particle swarm optimization," in *Proc. of the Int. Conf. on Service-Oriented Computing*, Berlin, Germany, pp. 237–251, 2013.
- [33] Y. Mao, V. Green, J. Wang, H. Xiong and Z. Guo, "DRESS: Dynamic resource-reservation scheme for congested data-intensive computing platforms," in *Proc. of the IEEE 11th Int. Conf. on Cloud Computing (CLOUD)*, San Francisco, CA, USA, pp. 694–701, 2018.
- [34] J. Liu and H. Shen, "Dependency-aware and resource-efficient scheduling for heterogeneous jobs in clouds," in *Proc. of the IEEE Int. Conf. on Cloud Computing Technology and Science (CloudCom)*, Luxembourg City, pp. 110–117, 2016.
- [35] W. Ahmad, O. Hasan, U. Pervez and J. Qadir, "Reliability modeling and analysis of communication networks," *Journal of Network and Computer Applications*, vol. 78, pp. 191–215, 2017.
- [36] E. Sousa, F. Lins, E. Tavares and P. Maciel, "Cloud infrastructure planning considering different redundancy mechanisms," *Computing*, vol. 99, no. 9, pp. 841–864, 2017.
- [37] A. Castiglione, M. Gribaudo, M. Iacono and F. Palmieri, "Modeling performances of concurrent big data applications," *Software: Practice and Experience*, vol. 45, no. 8, pp. 1127–1144, 2015.
- [38] J. Xu, Z. Pei, L. Guo, R. Zhang, H. Hu *et al.*, "Reliability analysis of cloud service-based applications through SRGM and NMSPN," *Journal of Shanghai Jiaotong University (Science)*, vol. 25, no. 1, pp. 57–64, 2020.
- [39] I. Ignatova and A. V. Biehun, "Estimating the reliability of the elements of cloud services," *Operations Research and Decisions*, vol. 27, no. 3, pp. 65–80, 2017.
- [40] O. Ivanchenko, V. Kharchenko, B. Moroz, L. Kabak and K. Smoktii, "Semi-markov's models for availability assessment of an infrastructure as a service cloud with multiple pools of physical and virtual machines," in *Proc. of the IEEE 9th Int. Conf. on Dependable Systems, Services and Technologies (DESSERT)*, Kyiv, Ukraine, pp. 98–102, 2018.
- [41] C. Shekhar, N. Kumar, A. Gupta, A. Kumar and S. Varshney, "Warm-spare provisioning computing network with switching failure, common cause failure, vacation interruption, and synchronized renegeing," *Reliability Engineering & System Safety*, vol. 199, pp. 106910, 2020.
- [42] E. Sousa, P. Maciel, L. Medeiros, F. Lins, E. Tavares and E. Medeiros, "Stochastic model generation for cloud infrastructure planning," in *Proc. of the IEEE Int. Conf. on Systems, Man, and Cybernetics*, Manchester, United Kingdom, pp. 4098–4103, 2013.
- [43] R. Ghosh, K. S. Trivedi, V. K. Naik and D. S. Kim, "End-to-end performability analysis for infrastructure-as-a-service cloud: An interacting stochastic models approach," in *Proc. of the IEEE 16th Pacific Rim Int. Symp. on Dependable Computing*, Tokyo, Japan, pp. 125–132, 2010.
- [44] M. R. Mesbahi, A. M. Rahmani and M. Hosseinzadeh, "Highly reliable architecture using the 80/20 rule in cloud computing datacenters," *Future Generation Computer Systems*, vol. 77, pp. 77–86, 2017.
- [45] M. Saini, O. Dahiya and A. Kumar, "Modeling and availability analysis of data center: A fuzzy approach," *International Journal of Information Technology*, vol. 13, pp. 1–8, 2020.