IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

Modified Artificial Fish Swarm Algorithm for Efficient Task Scheduling in Cloud Environment

H. Krishnaveni[1*], V. Sinthu Janita[2]

[1] Department of Computer Science, Cauvery College for Women, Trichy, India
[2] Department of Computer Science, Cauvery College for Women, Trichy, India

**Corresponding Author: hkrish6677@gmail.com, Tel.: +00-98435-33377*

DOI: https://doi.org/10.26438/ijcse/v7i5.13631371 | Available online at: www.ijcseonline.org

Accepted: 23/May/2019, Published: 31/May/2019

Abstract— In cloud computing, task scheduling plays an important key role. The tasks provided by the user are to be allocated to the resources in cloud and the users have to pay for the usage. Even though there are number of popular schedulers available for task scheduling in Grid and other distributed environments, they are not suitable for cloud. Cloud is different from other distributed environments in resource pool and encounters less failure rate. Task scheduling in cloud has to give attention to the QoS parameters such as deadline and budget. Most conventional heuristic algorithms are proposed in the literature. But the meta-heuristic algorithm like fish swarm approach for the task scheduling in cloud is expected to give way the optimal results. A new meta-heuristic technique inspired from the swarm intelligence of fish, namely Modified Artificial Fish Swarm (MAFS) Optimization for Efficient Task Scheduling in Cloud Environment, has been proposed to solve the task scheduling problem. Then the proposed algorithm is compared with existing algorithms such as Particle Swarm Optimization (PSO) and Genetic algorithm (GA). The experimental result shows that the proposed MAFS greatly reduces the makespan and execution cost.

Keywords— Task scheduling, resource utilization, cost, makespan

I. INTRODUCTION

Cloud computing provides an effective platform for provisioning services in terms of storing and maintaining user's data into the third party server [1]. This service is provided and managed by cloud service provider. By utilizing cloud service, the particular user need not store their data to the physical location of the user instead, they can utilize the service provider's datacenter. For getting this service, the customer has to pay the money to the service provider based on pay per use or subscription basis [2]. The cloud service provider provides the service based on the requirements of the user therefore; the user requests a service with a collection of requirements to the cloud service provider (CSP) [3]. The user requires virtual machine instances for provisioning their applications in the cloud service provider's platform [4] [5]. The cloud provides another effective scheme that is federated public cloud based on various CSP in which the consumer can achieve availability and cost effective services as well as avoids the vendor lock-in. This scheme make the cloud providers to render many opportunities to distribute their infrastructure, improving the capability of perishable resources, energy saving, unexpected loads can be addressed, and avoid the additional cost attained from abundance of provisioning of data centers. In addition federated public cloud is recommended as an effective cloud deployment model. With the advance feature of this technology researcher can able to accomplish compute-intensive scientific applications on various public IaaS platforms [6].

On the other hand, the role of datacenters is inevitable in cloud service. They carry multiple physical machines where each machine posses various virtual machines based on the requirement of the user for computing application. Moreover, the request comes from users perhaps carries different parameters constants, such as start time, execution time, number of virtual machines, cost and deadlines. In addition, cloud has incorporated with many applications and allocates on-demand resources [7]. In case of shortage of physical resources, a cloud service provider utilized optimized resource mechanisms to every customer. However, the physical resources are constrained in the capacity of CPU, memory, storage space, I/O devices, bandwidth and considerably more for the data center. The actual quantity of resources should be given to the user on demand, which ignores over-utilization and under-utilization.

Today, the modern researches have been held for focusing on maximizing cost effectiveness and utilization for applications effectively [8]. The goal of resource management is to use the resource as much as possible. Nowadays, the cloud service providers pay their interest on maximizing resource utilization and throughput and more specifically their profit

[9]. The resource usage can be accomplished by actualizing better task scheduling in cloud computing [10]. Also, the benefits of cloud computing make it, utilized by the various business environments. For instance, Amazon Web Services (AWS), Microsoft and Google are some service providers, earn huge profit with this service [11] [12]. Yet, cloud based services faced some sort of challenges especially scheduling and managing resource due to the reason of rapid growth of users into it [13]. Hence, cloud has to focus on improving its efficiency in terms of customers request or minimizing the computation time and execution cost. The scheduling process is based on mapping the clients request to the cloud resources and monitoring the previously assigned client's request execution process. Nowadays, many types of scheduling schemes are utilized such as Service Level Agreement based scheduling, heuristics scheduling, workflow scheduling, static scheduling and dynamic scheduling [13]. Moreover, some task scheduling algorithms are developed and utilized in cloud computing such as ABC algorithm, PSO, Min-Min, GA etc [14] [15]. But more efficient technique is needed to get optimum solution for scheduling problem. The next section of this work shows a clear information about cloud computing and task scheduling process where various research articles are utilized for attaining better understanding about cloud computing task scheduling process. In this paper, related work is given in section II. In section III, problem formulation is presented. Section IV describes about Artificial Fish Swarm Algorithm and Solution frame work using Modified Artificial Swarm Algorithm elaborately. Simulation results and discussion are given in section V. Section VI brings the conclusion.

II. RELATED WORK

Scheduling was significantly a decision-making process that provided resource distribution among different activities by determining their execution order on the collection of existing resources. The emergence of distributed systems brought new problems on scheduling in computer systems, including clusters, grids, and more recently clouds. On the other hand, the plethora of research made it hard for both newcomers and researchers to understand the coordination between various kinds of scheduling problems and schemes, proposed in the literature, which hampered the identification of new and relevant research avenues. Luiz F. Bittencourt et al. [16] introduced a classification of the scheduling problem in distributed systems by presenting a taxonomy that incorporated recent developments, especially that in cloud computing. They reviewed the scheduling literature to corroborate the taxonomy and analyzed the interest in different branches of the proposed taxonomy. Finally, they identified relevant future directions in scheduling for distributed systems.

Hui Jiang et al. [17] proposed the cloud-based scheduling to execute the disassembled tasks. They built a mathematical model that considered the uncertainty nature of the process and precedence relationships of disassembly tasks. Two objectives included in this work are minimizing the expected total makespan and minimizing the expected total cost of the disassembly service.

Zhang Qian et al. [18] have analyzed load balancing task scheduling algorithm in light of Feedback Mechanism for Cloud Computing, which is principally centered around task scheduling algorithm in view of weighted irregular and input instruments was proposed in this paper. At first, the chosen cloud scheduling host picked resources by requirements and made a static evaluation, and after that sorted them; also the algorithm chooses resource that are sorted by weight haphazardly.

A.I.Awad et al. [19] proposed a technique namely load balancing mutation particle swarm optimization after identifying the drawbacks of standard particle swarm optimization. The cons are some tasks are not allocated to virtual machines, certain tasks are allocated to more than one machine and premature convergence. The basic idea behind the work is to resechedule the failed tasks to the available virtual machines by taking into account the load of each vm. It guarantees that all vms executed the assigned tasks appropriate with their load. It is concluded that the technique produces better makespan, round trip time and transmission cost over other existing algorithms. Execution cost is not considered in this paper.

M. Krishna Sudha et al. [20] designed a coherent genetic algorithm to overcome the limitations of existing task scheduling algorithm Shortest Cloudlet to Faster Processor Algorithm with Genetic Algorithm(GA-SCFP). The algorithm computes the cost with mean value from GA-SCFP and compared with the original task completion time of GA-SCFP. Then the scheduling process is initiated. The experimental results shows that the proposed algorithm produces better makespan, and utilizes the resources efficiently with low scheduling cost.

Xuezhi et al. [21] explained a cost-effective task scheduling in cloud computing environment. Here they utilized the MapReduce concept and introduced a greedy-based MapReduce application scheduling algorithm. This algorithm mainly focused on user's constraints in order to minimize the cost of renting resource based on the budget and deadline constraints given by the user in Service Level Agreement.

Moreover, Demyana et al. [22] proposed a job scheduling technique in the cloud using firefly algorithm. This is based on data of jobs and assets, for example, lengths of the job, the speed of asset and identifiers. The planning capacity in the proposed job scheduling component right off the bat, makes an arrangement of occupations and assets to produce the populace by allocating the employments to assets haphazardly and assesses the populace utilizing a wellness esteem which speaks to the execution time of employment. Furthermore, the capacity utilized emphasis to recover populaces in light of firefly conduct, to create the best employment plan that gives the base execution time of occupations. Similarly, Ramezani et al. [23] have explained an Evolutionary algorithm-based multi-objective task scheduling optimization model in cloud environments.

To reduce the computation burden brought by scheduling algorithms, Tongxiang Wang et.al. [24] have proposed a novel technique Dynamic Tasks Scheduling algorithm based on Weighted Bi-graph model (DTSWB). At first, the scheduling problem was translated into a maximum weighted bi -graph matching problem. An integer programming model was formulated based on the matching problem. The task scheduling process mainly consisted of four parts: information collection of offloaded tasks and service providers, establishment of mapping relationship, determination of weight matrix and generation of optimal matching strategy based on Kuhn Munkras (KM) algorithm. Compared with existing batch-based scheduling schemes, their scheduling algorithm took the dynamic of tasks and service providers into consideration. At last, the effectiveness and validity of the proposed scheduling algorithm was verified by a series of simulations.

Nima Jafari et al. [25] proposed Cuckoo search algorithm for scheduling the tasks, which is based on the obligate brood parasitic behavior of some cuckoo species combined with levy flight behavior of some birds and fruit flies. They tried to reduce the execution time but the cost of execution is not considered. Also the results are not compared with other existing algorithms.

Dyah Pythaloka et al. [26] developed Artificial Fish Swarm Optimization Algorithm for job shop scheduling. It aims to minimize the completion time of entire job. It produces better efficiency value of 75%, but unsatisfactory based on the completion time resulted. The execution cost is not taken into account. Good level of efficiency is achieved by using 10,000 artificial fishes with 100 iterations within 372e+41 solution spaces.

Among the above algorithms, the meta-heuristics have acquired great achievements and become a popular tool for solving NP hard combinatorial optimization problems[27]. The artificial fish swarm algorithm (AFSA) introduced by Li [28] is a population-based meta-heuristic. It is insensitive to initial values and possesses good performance such as fast convergence, robustness and high fault tolerance [29]. So that it has gained an increasing study and wide applications such as multi objective optimization [30], job shop scheduling problem [31] and clustering problem[32]. Sebagenzi Jason et al. [33] revealed that energy efficient scheduling algorithms are not sufficient for cloud as the balancing of load of physical machines is also very important. G.U.Tambe et al. [34] applied assignment algorithm for the completion of the tasks in the heterogeneous environment with or without scheduling and achieved good result. Motivated by these perspectives, we propose an efficient artificial fish framework for Task scheduling in cloud environment.

III. PROBLEM FORMULATION

In business viewpoint, the virtual machines should execute the tasks as right on time as could reasonably be expected and these VMs continue running in parallel. This prompts issues in the scheduling of the customer task inside the available resources. The scheduler should do the scheduling procedure profitably in order to utilize the available resources totally. More than one task is assigned to atleast one VMs. This kind of conditions should guarantee that the tasks are loaded in all VMs i.e., it should guarantee that the tasks are not locked vigorously on one VM and some VMs don't remain to sit out of gear as well as under loaded. For this circumstance, it is the duty regarding the schedule to confirm the loads over the machines. A load balancing algorithm endeavors to upgrade the execution time of customer's submitted applications by ensuring maximal utilization of available resources.

Consider that $VM = \{VM_1, VM_2, ..., VM_i\}$ is a set of virtual machines (VM) types and $T = \{T_1, T_2, ..., T_m\}$ is a set of task. Each task contains a task set $T_i = \{t_1, t_2, ..., t_n\}$. Here, each node has computing power such as Energy, execution time, communication time, execution cost and resource utilization. Assume that, the execution time and and the cost of every resource per unit time are known and can be expressed by matrix ETC and ECC individually. ETC (i,j) represents the execution time of task i on node j while ECC(i,j) denotes the cost of node i per unit time. Task scheduling is seeking the mapping between tasks and virtual machines which meet the constraints and optimization goals.

IV. ARTIFICIAL FISH SWARM ALGORITHM

As AFSA is an optimizer based on population, the system is started with a set of randomly generated potential solutions and then performs the search for the optimum solution interactively. The dimension of the search space is declared as n, the scale of fish is N. Each AF is represented as a vector of n dimension $X_i=(x_{i1},x_{i2},\dots x_{in})(i=1,2,3,\dots,N)$. Fish usually stay in the place with lot of food, so that the researchers simulate the behaviors of fish based on this characteristic to reach the global optimum. Function Y=f(x) is the food concentration of AF that represents the objective value. X_i-X_j is the distance between the AFs in position X_i and X_j. δ implies the crowding factor and trynumber is the number of attempts in preying behavior.

Figure 1.shows the vision concept of the artificial fish (AF). Here, X is the current state of a AF, Visual is the field of vision of the AF and X_v is the visual position at some moment. If the state of the visual position is better than the current state, the AF goes forward one step in this direction and reaches at X_{next} position. Otherwise continues the inspecting tour in the vision. If the tour is greater in number then the AF obtains more knowledge about the overall states of the vision.

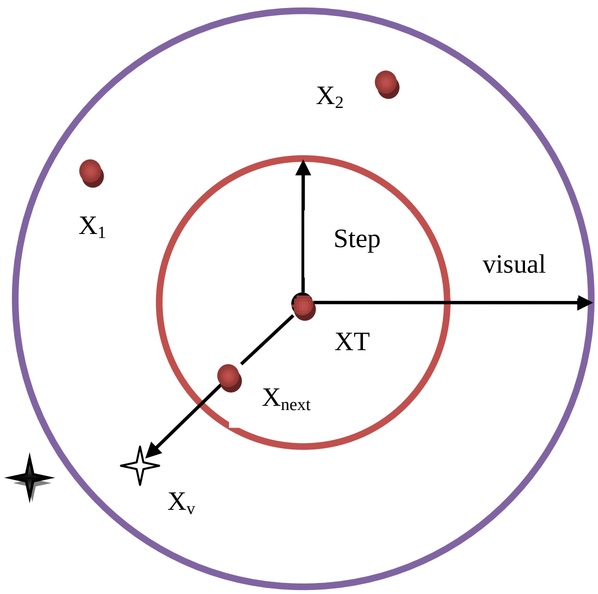

Figure 1 Visual concept of the Artificial fish

Solution frame work using Modified Artificial Fish Swarm Algorithm

This paper proposes a modified artificial fish swarm algorithm based multi-objective task scheduling. Figure 2 displays the architecture of the proposed method.

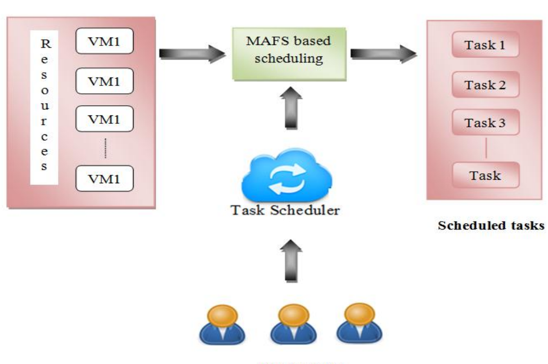

Figure 2 Architecture of proposed method

Modified Artificial Fish Swarm Algorithm

The major contribution in Modified Artificial Fish Swarm Algorithm (MAFS) over conventional AFS is Oppositional behaviour learning in which it creates reverse population. It will increase the efficiency of task scheduling. In light of the behavior depiction of the previously mentioned artificial fish, every artificial fish searches its natural conditions and its associates to pick a proper behavior to move at the speediest towards the optimal path. At last, the artificial fish assembles around a few neighborhood extrema. The practices of artificial fish incorporate prey behavior, swarm behavior along with follow behavior. These practices are depicted in the underneath area.

Prey or searching behavior:

This is a vital biological method that tends to the food. The fish update the position based on the more food concentrated region. Let Xi be the current state of AF and select a state Xj randomly in its visual distance, Y be the food concentration.

If Yj<Yi then

$$\vec{X}_{i+1} = \vec{X}_i + \frac{\vec{X}_j - \vec{X}_i}{\left\| X_j - X_i \right\|}.step.rand(r1) \tag{1}$$

Otherwise,

$$\vec{X}_{i+1} = \vec{X}_i + visual.rand(r2) \tag{2}$$

where,

$rand(r1)$ and $rand(r2)$ produces random numbers in the range [0,1] and [0,1] respectively.

Swarm behavior:

All fish will normally swim based on the food concentration and crown in order to avoid danger. A fish located at X_i has neighbours within its visual. X_{sc} identifies the center position of those neighbours and is used to describe the attributes of the entire neighboring swarm. If the swarm center has a greater concentration of food than is available at the fish's current position X_i (i.e., $Y_c < Y_i$), and if the swarm ($X_{sc)}$ is not overly crowded ($t_s/n < \delta$), the fish Here, t_s represents number of individuals within the X_c's visual. will move from X_i to next X_{i+1}, toward $X_{sc.}$ Swarming behavior is executed for a fish based on its associated X_c; otherwise, searching behavior guarantees a next position for the fish.

The central position X_{sc} of the swarm is

$$X_{sc} = \frac{1}{n}\sum_{i=1}^{n} X_i \qquad \text{----------(3)}$$

$$\vec{X}_{i+1} = \vec{X}_i + \frac{\vec{X}_{sc} - \vec{X}_i}{\left\| X_{sc} - X_i \right\|} .step.rand(r1)$$

-------------------(4)

Subject to

$$Y_{sc} < Yi \text{ and } \left(\frac{t_s}{n}\right) < \delta$$

Where, t_s -Number of individuals within the X_{sc} visual.

X_{SC} - swarm center of t_s neighbors;

Follow or chasing behavior:

When a single fish or several ones find food, the neighbourhood partners will trail and reach food quickly. If X_i is the current position of AF i, it checks neighbor position X_{ne}, if t_f is the number of AFs in the Visual of AF ne, if $f(X_{ne}) < f(X_i)$ and $\delta > (t_f/N)$, i.e. position X_{ne} has a better food consistency than the current position of AF i and population density in its neighbourhood is not much, therefore AF i moves one step toward AF by equation (5)

$$\vec{X}_{i+1} = \vec{X}_i + \frac{\vec{X}_{ne} - \vec{X}_i}{\left\| X_{ne} - X_i \right\|} .step.rand(r1)$$

--------------------(5)

Subject to

$$Y_{ne} < Y_i \quad and \quad \left(\frac{t_f}{n}\right) < \delta$$

Where, X_{ne} -Position of neighbour AF X_i

Within a fish's visual, certain fish will be perceived as finding a greater amount of food than others, and this fish will naturally try to follow the best one (X_{ne}) in order to increase satisfaction (i.e., gain relatively more food [$Y_{ne} < Y_i$] and less crowding [$t_f/n<\delta$]. t_f represents number of fish within the visual of artificial fish in position X_{ne}. Searching behavior will be commenced if following behavior is unable to determine a fish's next position.

Oppositional Behavior Learning

From the "Opposition based learning (OBL)", the initial and the dependable reverse solution is created. This will increase the effectiveness of our proposed algorithm to convalesce the accuracy of solutions. OBL is a method used to compute the opposite solution for each solution in the population then the best solution is selected according to the value of its objective function. In general, this method is used to enhance the convergence of the meta heuristic algorithms to find the global solution. Tizhoosh et al. [35] has proposed OBL technique to enhance computational speed and quicken the rate of convergence of various optimization algorithms. For randomly generated population number, it is not possible to guess about the value optimal solution and that is why time required to reach optimal solution is large. According to Mehmet Ergezer et al. [36], in OBL, opposite numbers or solutions are introduced along with population numbers. It is found that OBL has the capability to reach optimal solution in minimum time due to introduction of these opposite numbers.

Solution initialization

In optimization algorithm, the solution encoding is the important process. The solution is based on the number of task and virtual machines. Here, at first, we randomly initialize the solution based on the number of task and vms. The structure of the artificial fish is given in table 1.

Table 1 Structure of the artificial fish

T1	T2	T3	…	Tn
vm_1	vm_2	vm_3	…	vm_n

Objective Function:

The main objective is to minimize the total completion time of all the tasks by determining the appropriate set of tasks that are to be assigned to vms. f1(x) is the minimum makespan function (ie) the total completion time CT.f2(x) is the minimum execution cost function(ie) the total execution cost of all the 'n' tasks scheduled in the appropriate 'm' vms.

The objective function is given in equation (6)

$$F = f1(x) + f2(x)$$

---------------------- (6)

$$\text{Minimize } f1(x) = \Sigma \, CT\,(Ti, Rj) \qquad (7)$$

$$\text{Minimize } f2(x) = \Sigma \; cost(Rj(Ti)) * ET(Tij) \qquad (8)$$

Subject to the constraints

$$\sum_{j \in R} T_j = T \quad , \quad n > m$$

The best value among all the solution is selected as the optimal solution for initial iteration and it will update for doing the next iteration. Figure 3 represents the flow chart of the proposed algorithm MAFS

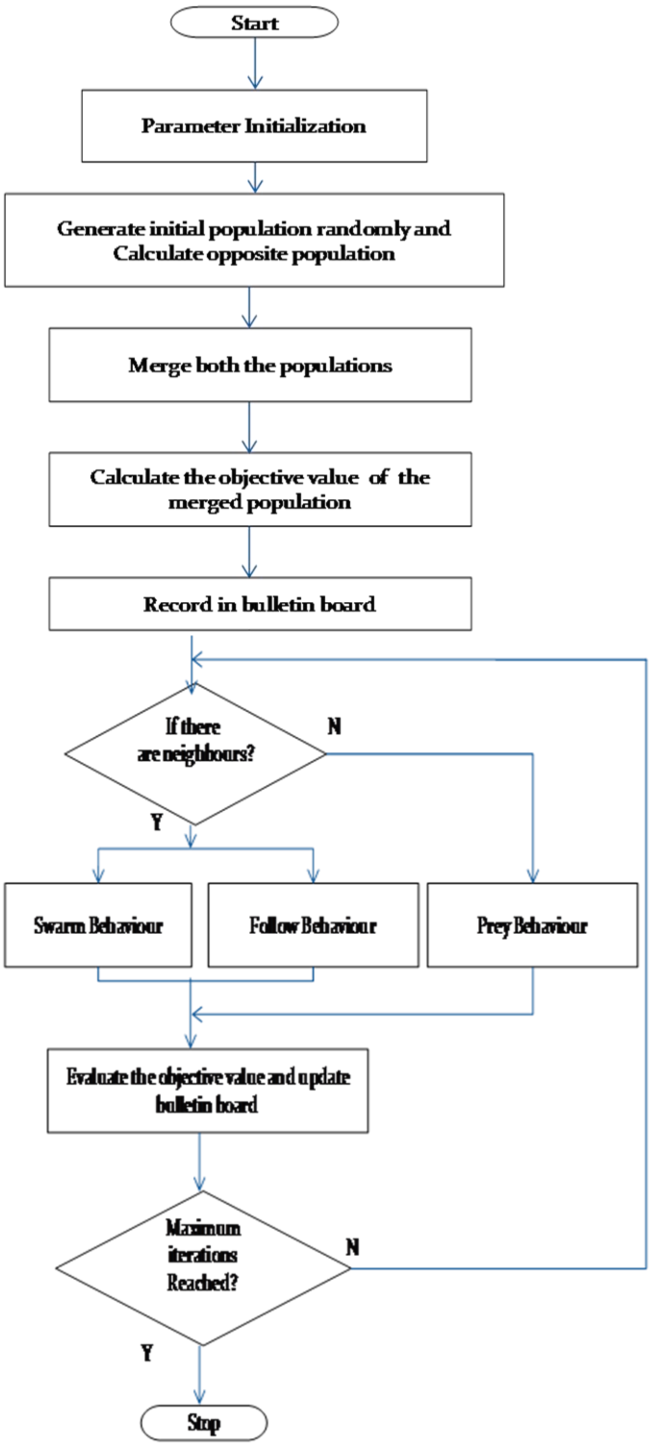

Figure 3 Flow chart of MAFS

The algorithm discontinues its execution only if a maximum number of iterations are achieved and the solution which is holding the best objective value is selected and it is specified as the best solution for scheduling. The pseudo code for modified artificial fish swarm is illustrated in beneath.

Pseudocode:

```
Step 1: Initialize the parameters
Step 2: Generate initial population of artificial fishes
          randomly
Step 3: Calculate opposite population
Step 4: Merge both the populations
Step 5: Compute fitness function  and record the  fittest
         artificial fishes in bulletin board
Step 6: If there are neighbours then
                    { Try  Swarm behaviour
                      Try Follow behaviour }
           Else
                     Try Prey behaviour
           End if
Step 7: Evaluate fitness and Update Bulletin board
Step 8: If maximum iterations reached then
                     Display the bulletin board
         Else
                     Go to step 5.
Step 9: Stop
```

The parameters used in scheduling are given in table 2.

Table 2 Parameters used in scheduling

Parameters	Definition of parameters
X_i	Current position of i^{th} artificial fish
$Y_i = f(X)$	Current concentration of food of artificial fish i
$\|\| X_j - X_i \|\|$	Distance between artificial fish individuals
visual	The field of vision of artificial fish
δ	Crowd factor(0< δ<1)
step	Maximum step size for artificial fish movement
trynumber	Largest trying number of each movement of artificial fish

V. SIMULATION RESULTS & DISCUSSION

In this section, we examine the outcome obtained from the proposed fish swarm optimization algorithm based task scheduling procedure in cloud. We have actualized our proposed undertaking task scheduling using cloud sim 3.0. The experiments are conducted by giving different values for various parameters and obtained better results in terms of convergence, makespan and cost, for a set of values. These values are given in the table 3.

Table 3 Parameter initialization

Parameter	Value
Initial Population	10
Visual	0.7
Crowd factor	0.75
Try number	6
step	0.2

We have evaluated the result of our proposed MAFS with other technique such as GA and PSO in terms of execution time, makespan, execution cost and resource utilization.

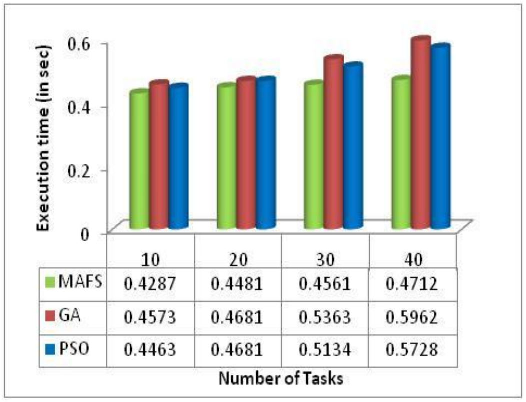

Figure 4 Execution time analysis by varying number of task

Figure 4 shows the performance of proposed method using execution time by varying number of task. If the number of task increases means; execution time of scheduling also increases. Here, our proposed approach takes a minimum time of 0.4286sec for ten tasks, 0.4481 sec for twenty tasks, 0.4561 sec for thirty task and 0.4712 sec for forty tasks.

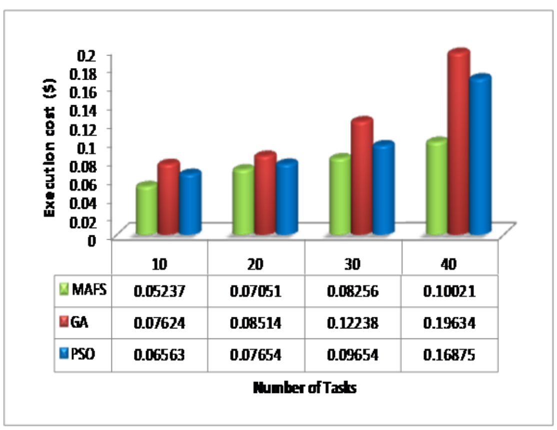

Figure 5 Execution cost analysis by varying number of tasks

Similarly, Figure 5 shows the performance of proposed method using execution cost by varying number of tasks. If the number of task increases means the cost of the scheduling also increases. Here our proposed approach utilizes the minimum cost for scheduling ten tasks.

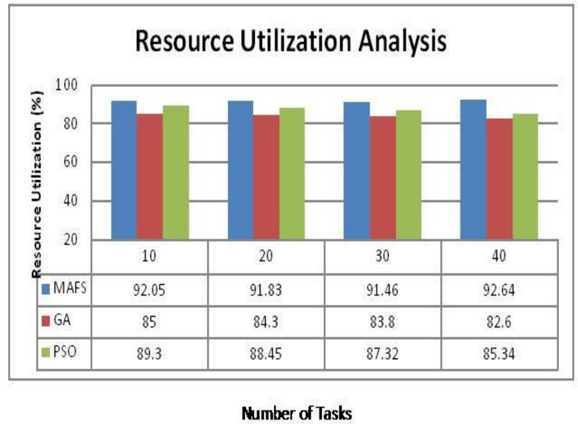

Figure 6 Resource utilization by varying number of tasks

Moreover, figure 6 shows the resource utilization performance. From the above discussion, we clearly understand that our proposed approach achieves the better performance compared to other approaches.

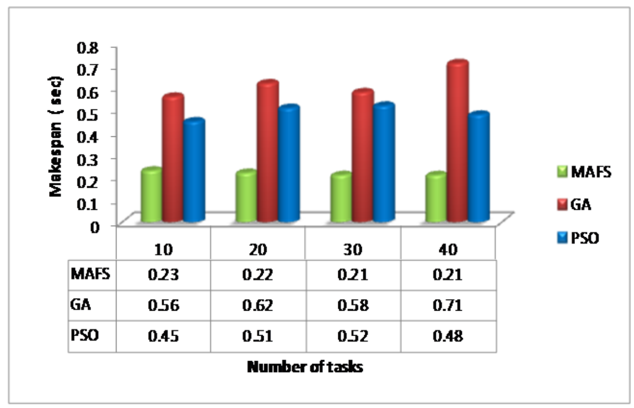

Figure 7 Makespan analysis

Figure 7 presents the makespan analysis. As per the analysis, the makespan gradually decreases when the iteration value goes on increasing. When the task increases to 40, our proposed approach obtains the minimum makespan of 0.211 secs which is low compared to other approaches.

VI. Conclusion and Future Scope

To solve the task scheduling problem on a cloud platform, this paper proposed a MAFS technique which utilizes the principles of evolutionary computation to reduce some redundant computations. The ultimate purpose of this research paper is to decrease the execution cost and makespan while scheduling the task in the cloud. Here, FCFS queue is created to enqueue the task in the task scheduler. The experiment is conducted by different tasks and iterations in the cloud. Experimental results demonstrates that the proposed algorithm not only minimize the execution cost during scheduling , but also enhances the comprehensive efficiency of the entire process and reduce the completion time and utilize vms efficiently. The performance of the proposed MAFS technique is compared with PSO and GA algorithm. The proposed MAFS algorithm is more effective in scheduling the tasks than the existing techniques.

REFERENCES

[1] Keshanchi, Bahman, Alireza Souri, and Nima Jafari Navimipour, "An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification, simulation, and statistical testing", Journal of Systems and Software, vol. 124, pp. 1-21, 2017.

[2] Koutsandria, Georgia, Emmanouil Skevakis, Amir A. Sayegh, and Polychronis Koutsakis, "Can everybody be happy in the cloud? Delay, profit and energy-efficient scheduling for cloud services", Journal of Parallel and Distributed Computing, vol. 96, pp. 202-217, 2016.

[3] Lin, Weiwei, Weiqi Wang, Wentai Wu, Xiongwen Pang, Bo Liu, and Ying Zhang, "A heuristic task scheduling algorithm based on server power efficiency model in cloud environments", Sustainable Computing: Informatics and Systems,2017.

[4] Mao, Li, Yin Li, Gaofeng Peng, Xiyao Xu, and Weiwei Lin, "A multi-resource task scheduling algorithm for energy-performance trade-offs in green clouds", Sustainable Computing: Informatics and Systems, vol.19, pp. 233-241,2018.

[5] Yang, Jiachen, Bin Jiang, Zhihan Lv, and Kim-Kwang Raymond Choo, "A task scheduling algorithm considering game theory designed for energy management in cloud computing", Future Generation Computer Systems, 2017.

[6] Abdi, Somayeh, Latif PourKarimi, Mahmood Ahmadi, and Farzad Zargari. "Cost minimization for deadline-constrained bag-of-tasks applications in federated hybrid clouds", Future Generation Computer Systems, vol.71, pp.113-128, 2017.

[7] Nayak, Suvendu Chandan, Sasmita Parida, Chitaranjan Tripathy, and Prasant Kumar Pattnaik, "An Enhanced Deadline Constraint Based Task Scheduling Mechanism for Cloud Environment", Journal of King Saud University-Computer and Information Sciences,2018.

[8] Gill, Sukhpal Singh, Inderveer Chana, Maninder Singh, and Rajkumar Buyya,"CHOPPER: an intelligent QoS-aware autonomic resource management approach for cloud computing", Cluster Computing, vol.1, pp. 1-39, 2017.

[9] Maria Carla Calzarossa, Marco L. Della Vedova and Daniele Tessera, "A methodological framework for cloud resource provisioning and scheduling of data parallel applications under uncertainty", Future Generation Computer Systems, In press, accepted manuscript, 2018.

[10] Sobhanayak, Srichandan, Ashok Kumar Turuk, and Bibhudatta Sahoo, "Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm", Future Computing and Informatics Journal, 2018.

[11] Panda, Sanjaya Kumar, Shradha Surachita Nanda, and Sourav Kumar Bhoi, "A Pair-Based Task Scheduling Algorithm for Cloud Computing Environment", Journal of King Saud University-Computer and Information Sciences, 2018.

[12] Yuan, Haitao, Jing Bi, MengChu Zhou, and Ahmed Chiheb Ammari. "Time-Aware Multi-Application Task Scheduling With Guaranteed Delay Constraints in Green Data Center", IEEE Transactions on Automation Science and Engineering, vol.15, issue. 3, pp. 1138-1151, 2018.

[13] Arunarani, A. R., D. Manjula, and Vijayan Sugumaran, "Task scheduling techniques in cloud computing: A literature survey," Future Generation Computer Systems, vol.91, pp. 407-415, 2019.

[14] Rashidi, Shima, and Saeed Sharifian. "A hybrid heuristic queue based algorithm for task assignment in mobile cloud." Future Generation Computer Systems, vol.68, pp. 331-345, 2017.

[15] Kumar, Mohit, and S. C. Sharma. "PSO-COGENT: Cost and Energy Efficient scheduling in Cloud environment with deadline constraint", Sustainable Computing: Informatics and Systems, vol.19, pp. 147-164 , 2018.

[16] Bittencourt, Luiz F., Alfredo Goldman, Edmundo RM Madeira, Nelson LS da Fonseca, and Rizos Sakellariou. "Scheduling in distributed systems: A cloud computing perspective", Computer Science Review, vol. 30, pp. 31-54, 2018.

[17] Jiang, Hui, Jianjun Yi, Shaoli Chen, and Xiaomin Zhu. "A multi-objective algorithm for task scheduling and resource allocation in cloud-based disassembly", Journal of Manufacturing Systems, vol. 41, pp. 239-255, 2016.

[18] Zhang Qian, Ge Yufei and Liang Hong "A Load Balancing Task Scheduling Algorithm based on Feedback Mechanism for Cloud Computing ", International Journal of Grid and Distributed Computing, vol. 9, issue. 4, pp.41-52, 2016.

[19] A.I.Awad, N.A.El-Hefnawy, H.M.Abdel_kader, "Enhanced Particle Swarm Optimization For Task Scheduling In Cloud Computing Environments", International Conference on Communication, Management and Information Technology (ICCMIT2015), Elsevier, Science Direct, Procedia Computer Science, vol. 65, pp. 920 – 929, 2015.

[20] M. Krishna Sudha, Dr. S. Sukumaran, "Coherent Genetic Algorithm for Task Scheduling in Cloud Computing Environment", Australian Journal of Basic and Applied Sciences, vol.9, issue.2, pp. 1-8, 2015, ISSN: 1991-8178.

[21] Xuezhi Zeng, Saurabh KumarGarg, Zhenyu Wen, Peter Strazdins, Albert Y.Zomaya and Rajiv Ranjan, "Cost efficient scheduling of MapReduce applications on public clouds", journal of computational science, 2017.

[22] Demyana Izzat Esa and Adil Yousif, "Scheduling Jobs on Cloud Computing using Firefly Algorithm", International Journal of Grid and Distributed Computing", vol. 9, issue. 7 pp.149-158, 2016.

[23] F. Ramezani, J. Lu, J. Taheri, F.K. Hussain, "Evolutionary algorithm-based multi- objective task scheduling optimization model in cloud environments", Journal of world wide web-Internet And Web Information Systems, vol.18, issue.6, pp.1737-1757, 2015.

[24] Wang, Tongxiang, Xianglin Wei, Tao Liang, and Jianhua Fan, "Dynamic Tasks Scheduling Based on Weighted Bi-graph in Mobile Cloud Computing", Sustainable Computing: Informatics and Systems, vol. 19, pp. 214-222, 2018.

[25] Nima Jafari Navimipour and Farnaz Sharifi Milani, "Task Scheduling in the Cloud Computing based on the Cuckoo Search Algorithm", International Journal of Modeling and Optimization, vol.5,issue.1, pp.44-47, Feb 2015.

[26] Dyah Pythaloka, Agung Toto Wibowo, Mahmud Dwi Sulistiyo , "Artificial Fish Swarm Algorithm for Job Shop Scheduling Problem", IEEE, 2015.

[27] X.S. Han, Y.C. Liang, and Z.G. Li, "An efficient genetic algorithm for optimization problems with time consuming fitness evaluation", International Journal of Computer Methods, vol.12, iss.1,pp.1-24, 2015.

[28] X. Li, Z. Shao, and J. Qian, "An optimizing method based on autonomous animates: fish-swarm algorithm", System Engineering Theory and Practice, vol. 22, issue.11, pp.32-38, 2002.

[29] M. Neshat, G. Sepidnam, M. Sargolzaei, and A.N. Toosi, "Artificial fish swarm algorithm: a survey of the state-of the-art, hybridization, combinatorial and indicative applications", Artificial Intelligence Review- Springer, vol. 42, issue. 4,pp. 965-997, 2002.

[30] M. Jiang and K. Zhu, "Multi objective optimization by artificial fish swarm algorithm", Proceeding of the 2011 IEEE International Conference on Automation Science and Engineering, (Trieste, Italy), pp. 506-511, 2011.

[31] K. Zhu and M. Jiang, "The optimization of job shop scheduling problem based on artificial fish swarm algorithm with tabu search strategy", Proceeding of International Workshop on Advanced Computational Intelligent (Hangzhou, China), pp. 323–327, 2013.

[32] S. He, N. Belacel, H. Hamam, and Y. Bouslimani ," Fuzzy clustering with improved artificial fish swarm algorithm", Proceeding of the International Joint Conference on Computational Sciences and Optimization (Hainan, Sanya, China), vol:2, pp.317-321, 2009.

[33] Sebagenzi Jason, Suchithra. R, "Scheduling Reservations of Virtual Machines in Cloud Data Center for Energy Optimization", International Journal of Scientific Research in Computer Science and Engineering **,** Vol.6, Issue.6, pp.16-26, 2018.

[34] G.U.Tambe , P.R. Bhaladhare, "Efficient Resource Sharing in Heterogeneous Environments", International Journal of Scientific Research in Network Security and Communication, Vol.5 , Issue.3 , pp.123-127, 2017.

[35] Hamid R. Tizhoosh,"Opposition-based Learning: A new scheme for Machine Intelligence", Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation and International Confe Intelligent Agents, Web Technologies and Internet Commerce,IEEE,2005.

[36] Mehmet Ergezer, Dan Simon, "Oppositional Biogeography-based Optimization for Combinatorial Problems", IEEE, 2011.

Authors Profile

Mrs H Krishnaveni pursed Bachelor of Science from Bharathidasan University, Trichy in 1997 and Master of Computer Applications in Bharathidasan University in year 2000. She is currently pursuing Ph.D. and currently working as Associate Professor in Department of Computer Science, Cauvery College for Women, Tiruchirapalli since 2001. She is a member of ISSE. She has published more than 5 research papers in reputed international journals and conferences including Springer and it's also available online. Her main research work focuses onCloud Scheduling, Cloud Security and Privacy, Big Data Analytics and IoT. She has 18 years of teaching experience.

Dr Sinthu Janita pursed Master of Computer Applications and Ph.D from Bharathidasan University. She is currently working as a Professor and Head in the PG &Research Department of Computer Science, Cauvery College for Women, Tiruchirapalli. She is a member of IEEE, Computer Society of India and a life member of ISSE.She has published more than 35 research papers in reputed international journals including Thomson Reuters (SCI & Web of Science) and conferences including IEEE, Elsevier, ACM, Springer and it's also available online. Her main research work focuses on Communication Networks, Network Security, Big Data Analytics, Data Mining and Mobile Communications. She has 23 years of teaching experience and 14 years of Research Experience..

