Energy Efficient Scheduling in Cloud Computing using Black Widow Optimization

Manoj Kumar¹ and Suman²
¹ Research Scholar, CSE Department, DCR University of Science and Technology, Murthal, India
² Professor, CSE Department, DCR University of Science and Technology, Murthal, India
E-mail: manojsarasviya@gmail.com, suman.cse@dcrustrm.org

Abstract. The delivery of services at IaaS, SaaS, and PaaS levels in Cloud Computing through internet proves to a promising technology. The increase in demand of services of cloud computing increased the creation of cloud datacenters in the world. This leads to increase in demand of energy consumption by datacenter. The integration of green computing with cloud tries to minimize the consumption of power, to minimize their energy costs, and increases their profit. Minimizing the energy consumption with promise of better Quality of Service (QoS) seems to be mutually explosive task for cloud service providers (CSP). To deal with these, a meta-heuristic technique called Energy Efficient Black Widow Optimization based Scheduling algorithm (EEBWOSA) is developed and analyzed in this paper. The special stage of cannibalism which excludes bad solutions will not be used for generating new solution which leads to early convergence. The performance of EEBWOSA is tested on workload taken from HPC2N dataset in CloudSim tool. It exhibits reduction in energy consumption by 25.69% and 13.52% as compared to GA and PSO.
Keywords: Cloud Computing, Energy, Black Widow, Makespan

1. Introduction
The growth of Communication and Information Technology (IT) has made human life more comfortable to live. Sustainable and Wearable Computing added a passion to the growth of cloud computing. Every smartphone is now connected to cloud service to store data and use cloud services on demand at pay per use [1], [2]. The interaction of cloud computing with mobile, fog, IoT has disseminated the cloud providers to develop more datacenters to provide services to its users. The development and expansion of more datacenters has increased a burden of energy consumption bill for the cloud services providers (CSP) [3], [4]. Due to huge demand of the energy, Google has built up new datacenter near river to get electricity at cheap price[5].

Green computing [6] has played a vital role in minimizing the energy consumption in various technology like wireless networks, sensors, computer hardware, cloud, fog, mobile etc. The computation intelligence has also played a vital role in minimizing the energy consumption in cloud computing. Several heuristics and meta-heuristics algorithms have been developed in cloud computing to minimize energy consumption [7] such as dispensing the load to the virtual machines (VM) with minimum energy consumption, called as load balancing. VM consolidation also helps in saving the energy by VM migration from one host to another. Some new techniques have been proposed to perform Live migration in order to save the energy. Server consolidation also helpful in saving energy by placing many underutilized VMs onto one host and place the rest host on sleep mode [8]. However, quality of service (QoS) should not be disturbed while minimizing the energy in datacenter. Therefore, minimizing energy with maintaining QoS is a challenging issue in cloud computing.
In recent years, computational intelligence has shifted towards physical science-based computation like Gravitational Search Algorithm (GSA) [9], Black Hole Algorithm (BHA) [10], Sine Cosine Algorithm (SCA) [11], Big-Bang Big-Crunch (B3C) optimization algorithm [12], Water Cycle Algorithm (WCA) [13] etc., human-social based computation like Teaching-Learning Based Optimization (TLBO) [8], Soccer League Competition (SLC) algorithm [14], Socio Evolution and Learning Optimization Algorithm (SELOA) [15] etc. and hybrid computation intelligence like improving Genetic Algorithm (GA) with Min-Max or Max-Min [16], GA with fuzzy theory [17], modified GA with particle swarm optimization (PSO) [18], hybrid PSO and Simulated Annealing (SA) [19], Ant Colony Optimization with GA, PSO, [20,21] [22] etc. to solve various problems. These techniques are applied in cloud computing to address the various challenges of cloud computing.

In this article, an Energy Efficient Black Widow Optimization based Scheduling Algorithm (EEBWOSA) is proposed to minimize energy and maintain QoS in terms of makespan. The rest of the paper contains following sections; section I gives a brief introduction of energy efficient in cloud computing. Section 2 reviewed some of the related work. Problem formulation is presented in section 3. The working of proposed work is explained with flowchart in section 4. Simulation environment and results are discussed in section 5 and 6 with some analysis. Finally, conclusion is presented last with some valuable suggestion to proceed the work in future.

2. Related Work

Energy consumption has always been a challenging issue for scientist, researchers and academicians since its origin. Many techniques or algorithms have been developed to minimize energy consumption with better utilization of resources. Single objective optimization tries to minimize cost, makespan in scheduling, but has to compromise with other like QoS, service cost, response time, makespan, deadline. Some of the related work in energy consumption is discussed here.

Ben Alla et al. [23] designed energy efficient scheduling with deadline technique to reduce energy consumption with minimizing makespan under deadline as a constraints by taking user’s priority. The proposed hybridization was comprised of Differential Equation (DE) and Multiple Criteria Decision-Making method. However, no real-world dataset was used in simulation and number of tasks are very less to perform experiments. Abdullahi et al. [24] addressed multi objective problem of scheduling in terms of makespan and cost using symbiotic organisms search algorithm and chaotic optimization strategy. The proposed algorithm able to minimize trade-off between budget and execution time with no computational overhead. Nayak et al. [25] developed a backfilling mechanism to improve the scheduling of tasks in cloud computing environment with deadline as a constraints.

Chen et al. [26] proposed minimizing the schedule dimension using the budget level (MSLBL) algorithm taking budget as constraint for various applications running in cloud environment. The proposed algorithm is tested on scientific workflow where each task is connected to another task. The algorithm achieves better performance as compared to HBCS algorithm [27]. Jena [28] introduced competent task scheduling algorithm based on Clonal Selection Algorithm (CSA) to minimize energy consumption and dealing time with deadline as a constraint and each user is assigned to one datacentre. However, a datacentre may be assigned to many users in real cloud environment.

Lawanyashri et al. [29] improved fruitfly optimization algorithm with simulated annealing to complete fast performance time and optimal utilization of cloud resources. The algorithm achieved degree of balancing and save energy as compared various other algorithms. Zhu et al. [30] developed a heuristic techniques to execute task with minimum energy consumed by datacentre. Author restrict multiple tasks execution on same virtual machine and no migration is allowed during execution. The algorithm outperforms in saving energy as compared to EARH, EDF and FCFS algorithm.

Adhikary et al. [6] framed all VMs into a clusters and further grouped into global and local cluster for scheduling based on energy calculation, resource requirement and availability. It manages to save energy in datacentre and networking devices, minimizing number of hosts in sleep mode and bandwidth utilization as compared to EHS and ESI algorithm. Du Guangyu et al. [31] introduced a novel method for minimizing the energy for heterogeneous cloud environment where all tasks must finished their
execution within deadline. The method is implemented in CloudSim toolkit [32] and comparisons are made with GA, HEFT, HPPO techniques in saving energy from 33.23% to 55.36%. Kumar et al. [33] modified PSO algorithm to save energy, and optimize execution cost and time, and increase throughput as compared to PSO, honey bee and max-min algorithm. Weipeng Jing et al. [34] also used PSO taking time, cost and reliability to improve the performance.

As far from literature, it is concluded that meta-heuristics techniques have played a vital role in minimizing energy while taking deadline and cost as a constraint. These algorithms have delivers QoS in one parameter under constraints circumstances, while there are situations where any constraint is not available for cloud service provider and it has to minimize the energy consumption of datacentres, while maintaining QoS in terms of makespan, service cost, resource utilization and degree of balance. To address the problem of efficient QoS in terms of makespan and service cost while minimizing energy consumption, an EEBWOSA method is proposed in this article.

3. Problem Formulation
Scheduling in cloud computing is assignment of tasks to best available virtual machines in the datacentres to fulfil the demands of the users. Consider a set of physical hosts PM = {PM1, PM2, PM3, ....... PMa} is available in datacentre. Virtual machines, VMj = {VM1, VM2, VM3, ..... VMb}. there are k number of users that are using the services of cloud and they have submitted activities in the form of tasks that will be executed on various virtual machines. The set of activities can be defined as A = {A1, A2, A3, ......... An}. In this, each activity is assigned to virtual machine VMj in PM, so that fitness of objective function is minimized.

The cloud broker is responsible to assign virtual machines for each activity/task with least completion time. The expected to complete is defined as the execution time of each activity/task that are executed on any virtual machine and can be expressed using Expected Time to Complete (ETC) matrix as shown in Eq. (1).

\[
ETC(Ac, Vb) = \begin{bmatrix}
A_1V_1 & A_1V_2 & A_1V_3 & \cdots & A_1V_b \\
A_2V_1 & A_2V_2 & A_2V_3 & \cdots & A_2V_b \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
A_nV_1 & A_nV_2 & A_nV_3 & \cdots & A_nV_b 
\end{bmatrix}
\] (1)

Similarly, the expected cost of execution (EC) of each activity/task in cloud that user has to pay to cloud service provider can be expressed using Eq. (2) by set of cost \(C = \{C_1, C_2, C_3, \ldots, C_c\}\).

\[
EC(Cc, Vb) = \begin{bmatrix}
C_1V_1 & C_1V_2 & C_1V_3 & \cdots & C_1V_b \\
C_2V_1 & C_2V_2 & C_2V_3 & \cdots & C_2V_b \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
C_cV_1 & C_cV_2 & C_cV_3 & \cdots & C_cV_b
\end{bmatrix}
\] (2)

The energy consumption produced by activity/task \(A_c\) while executing on virtual machine \(VM_b\) is calculated by using Eq. (3).

\[
E_T = E_0 + \sum EA_c
\] (3)

\[
EA_c = R \times A_c
\] (4)

Where \(E_0\) is the idle needed to run datacenter, and R is the energy consumption rate of virtual machine running for activity/task \(A_c\). Therefore, fitness value of the problem is to minimize makespan, cost, and energy.

The makespan of all the activities/tasks is calculated by using Eq. (1).

\[
Makespan, M(x) = \min \sum T_i , i = 1, 2, 3 \ldots c
\] (5)

Where \(T_i\) is the completion time activity/task \(A_c\).
The cost of execution all activities/tasks is calculated by using Eq. (2).

\[ \text{Cost}, P(x) = \min \sum T_i \times C_c, i = 1,2,3 \ldots , c, \]  

(6)

Where \( C_c \) is the cost of execution of activity/task, \( A_c \). The total energy is calculated using the Eq. (3). So total energy can be expressed using Eq. (7).

\[ E(x) = \min \sum E_f \]  

(7)

Where \( E_c \) is the sum of energy consumed by all activities/tasks in the datacenter. Therefore, fitness function of problem can be formulated by combining the Eq. (5), Eq. (6), and Eq. (7) and is presented in Eq. (8).

\[ \text{Fitness Function}, F(x) = \min[\alpha \times M(x) + \beta \times P(x) + \gamma \times E(x)] \]  

(8)

Where \( \alpha+\beta+\gamma=1 \), and \( \alpha, \beta, \gamma \) are the balancing factors in optimizing the fitness function.

4. Proposed Energy Efficient Black Widow Optimization based Scheduling Algorithm (EEBWOSA)

Black widow optimization algorithm is developed by Hayyolalam et al. [35] due to its unique mating of female black widow with male black widow. It includes the stage of cannibalism, in which inappropriate fitness are removed from the population. Secondly, it doesn’t get trapped in local minima and leads to early convergence. Due to these features, it can be used to address the scheduling problem in cloud computing while optimizing multiple parameters like energy, cost, and makespan.

![Figure 1. Flowchart of Black Widow Algorithm](image-url)
The black widow algorithm begins with the initialization of populations, in which each spider represents a possible solution. These initial population generate new solution in pairs. Female black widow either kill or eat male black widow during mating. This phenomenon will not forward the new solution to next stage. The different phases of the proposed algorithm are explained and presented in Figure 1.

4.1. Initial population
Initial population is randomly generated like in GA and PSO, and individuals in each population is called as widow, which shows the solution of each problem variables. The widow is represented using Eq. (9).

\[
\text{Widow} = [x_1, x_2, x_3, \ldots \ldots \ldots x_N]
\]  

(9)

Where each of the variable values is a floating-number. The fitness of the widow is obtained by evaluation of objective function \( F \) at a widow of \((x_1, x_2, x_3, \ldots \ldots \ldots \ldots \ i\. x_N)\) as in Eq. (1). So,

\[
\text{Fitness} = f(\text{widow}) = f(x_1, x_2, x_3, \ldots \ldots \ldots \ldots x_N)
\]  

(10)

A matrix of size \( N_{\text{pop}} \times N \) is generated with an initial population which is called as candidate widow. The procreating is performed by random selection of parents by mating, and female kills or eats male widow.

4.2. Procreate
The couples are unknown to each other, and they start mating with each other to produce new generation. The offspring is produced using Eq. (11) and (12) using array called theta. The process is repeated for \( N/2 \) times, while duplicated selected number must not be repeated. Mothers and their children are stored in array according to fitness value and some individuals will be added to newly generated population as per cannibalism rating.

\[
y_1 = \alpha \times x_1 + (1- \alpha) \times x_2
\]  

(11)

\[
y_2 = \alpha \times x_2 + (1- \alpha) \times x_1
\]  

(12)

4.3. Cannibalism
There are three kinds of cannibalism, firstly female black widow eats or kills her husband during mating. Secondly, weak siblings are eaten by strong siblings. Thirdly, some baby spiders eat their mother. The cannibalism rating is set to determine the number of survivors with their fitness values, stronger or weaker.

4.4. Mutation
In this stage, Mutepop number of individuals are selected randomly from population according to mutation rate. Each of randomly selected solutions randomly exchange their two elements in the array. Mutation is presented in Fig. 2. All parameters values of proposed EEBWOSA is presented in table 1.

![Figure 2. Mutation](image)

Where each of the variable values is a floating-number. The fitness of the widow is obtained by evaluation of objective function \( F \) at a widow of \((x_1, x_2, x_3, \ldots \ldots \ldots \ldots \ i\. x_N)\) as in Eq. (1). So,

\[
\text{Fitness} = f(\text{widow}) = f(x_1, x_2, x_3, \ldots \ldots \ldots \ldots x_N)
\]  

(10)

A matrix of size \( N_{\text{pop}} \times N \) is generated with an initial population which is called as candidate widow. The procreating is performed by random selection of parents by mating, and female kills or eats male widow.

4.2. Procreate
The couples are unknown to each other, and they start mating with each other to produce new generation. The offspring is produced using Eq. (11) and (12) using array called theta. The process is repeated for \( N/2 \) times, while duplicated selected number must not be repeated. Mothers and their children are stored in array according to fitness value and some individuals will be added to newly generated population as per cannibalism rating.

\[
y_1 = \alpha \times x_1 + (1- \alpha) \times x_2
\]  

(11)

\[
y_2 = \alpha \times x_2 + (1- \alpha) \times x_1
\]  

(12)

4.3. Cannibalism
There are three kinds of cannibalism, firstly female black widow eats or kills her husband during mating. Secondly, weak siblings are eaten by strong siblings. Thirdly, some baby spiders eat their mother. The cannibalism rating is set to determine the number of survivors with their fitness values, stronger or weaker.

4.4. Mutation
In this stage, Mutepop number of individuals are selected randomly from population according to mutation rate. Each of randomly selected solutions randomly exchange their two elements in the array. Mutation is presented in Fig. 2. All parameters values of proposed EEBWOSA is presented in table 1.

![Figure 2. Mutation](image)
Table 1. Parameters Values in Heuristic Algorithms.

| Algorithms | Parameters       | Values |
|------------|------------------|--------|
| GA         | Population Size  | 1000   |
|            | Max Iteration    | 1000   |
|            | Crossover rate   | 0.5    |
|            | Mutation Rate    | 0.1    |
| PSO        | Particle Size    | 100    |
|            | Self-coefficient, c1, c2 | 2     |
|            | Max Iteration    | 1000   |
|            | Inertial Weight  | 0.9-0.4|
| EEBWOSA    | Population Size  | 1000   |
|            | Procreate Rate   | 0.6    |
|            | Cannibalism Rate | 0.44   |
|            | Mutation Rate    | 0.4    |
|            | Rate of keeping habitat | 0.6 |

5. Simulation Environment
Simulation was carried out with Apache NetBeans IDE 11.3 configured with CloudSim Toolkit [32], written in Java, installed on HP ProDesk desktop PC having 64-bit Windows 10, 16 GB RAM, 3.20 GHz CPU with 8 cores. All parameters for settings of CloudSim are presented in Table 2. The proposed EEBWOSA technique was tested on HPC2N dataset, workload taken from [36], recognized by CloudSim. The pricing model of all virtual machines was Google App Engine.

Table 2. Simulation Parameters in CloudSim.

| S No | Entities | Parameters       | Values       |
|------|----------|------------------|--------------|
| 1    | User     | No. of Users     | 10-100       |
|      | Broker   |                  | 1            |
| 2    | Cloudlet | No. of Cloudlet  | 100-1000     |
|      |          | Length           | 1000-50000   |
|      |          | File Size        | 600          |
| 3    | VM       | No. of VM        | 50           |
|      |          | Policy           | Space Shared |
|      |          | Bandwidth        | 1000         |
|      |          | MIPS             | 600-2400     |
|      |          | VMM              | Xen          |
|      |          | OS               | Linux        |
|      |          | No. of CPU       | 1 Each       |
| 4    | Datacenter | No. of Datacenter | 2           |
|      |          | No. of Hosts     | 20           |
6. Results and Discussion

This section presents the performance of the proposed EEBWOSA. BWO is a swarm based computational intelligence algorithms, so GA and PSO are taken for comparison of proposed EEBWOSA in terms of makespan, cost, and energy consumption. Finally, statistical significance is tested and compared with GA and PSO.

![Figure 3. Cost of Execution on HPC2N](image)

Figure 3 shows the cost of execution of EEBWOSA algorithm computed on HPC2N. The simulation has been performed on various number of cloudlets ranging from 100-1000. The cost of execution for proposed EEBWOSA is 18.02 % less as compared to GA and 12.41 % less as compared to PSO. It has been observer that the performance of the proposed algorithm increases with large number of users as compared to GA and PSO.

![Figure 4. Makespan on HPC2N](image)
Figure 4 shows the makespan computed by GA, PSO, and proposed algorithm. The proposed algorithm performs better and is able to finish the execution of all cloudlets in less time. Simulation has been performed on different number of cloudlets ranging from 100 to 1000 as an input. The makespan rises with increase in number of cloudlets. However, with increasing the number of cloudlets to some limit, the makespan increases with constant rate. This is because of the increase in waiting time of cloudlets. However, the proposed algorithm performs better as compared to GA with an average of 20.22%, and 13.21% as compared to PSO.

Figure 5 shows the energy consumption computed by GA, PSO, and EEBWOSA in KWH. The simulation is carried out on different number of cloudlets ranging from 100 to 1000 and their energy is calculated. From the graph, it is concluded that energy consumed by our proposed algorithm is very less on higher number of cloudlets. It means that our proposed algorithm can reduce energy consumption cost and increases the profit to cloud service provider.

The statistical significance is carried out on the results obtained by three algorithms. The statistical significance is tested with regression and correlation values. It is found from the table 3 that energy, cost, and makespan has significant correlation with each other at R=1 and confidence level was 99%.

7. Conclusion future work
Scheduling is cloud computing environment is still a big challenge for researchers. There is always quest for optimal solution to provide better Quality of Service. The problem further increases in case of multi-optimization where, more than one parameter needs to be optimized and they are inversely dependent on each other. In this paper, an energy efficient scheduling algorithm based on Black Widow Optimization is proposed to reduce energy consumption in datacentres and provide better Quality of Service in terms of makespan and cost to its users. The cannibalism stage avoids bad solutions to next stage for new generation, that leads to early convergence. The simulation results show that energy consumption can be reduced up to 12% and 13% with better Quality of Service in terms of makespan and cost when compared to popular algorithm namely, GA and PSO. This research work can be further extended to minimize the energy consumption at VM migration process. BWO is not trapped at local minima, so this property can be promising for new computing paradigm.
Table 3. Statistical Analysis of GA, PSO, and EEBWOSA on makespan, Cost, and Energy

| Makes plan | Number of Job | GA | PSO | EEBWOSA | GA | PSO | EEBWOSA | GA | PSO | EEBWOSA |
|------------|---------------|----|-----|---------|----|-----|---------|----|-----|---------|
| Number     | 10            | 1.000** | .986** | .994** | .999** | .995** | .997** | .995** | .998** | .993** |
| Pearson Correlation | 1.000** | .986** | .994** | .999** | .995** | .997** | .995** | .998** | .993** |
| Sig. (2-tailed) | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| N           | 10            | 10     | 10     | 10     | 10     | 10     | 10     | 10     | 10     |

Acknowledgement
This research work is supported by University Grant Commission under National Fellowship Program, Ministry of Social Justice and Empowerment, Govt. of India, grant no. 29146. Author also acknowledge the CSE Department of Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Sonipat for providing research facilities in research lab.
References

[1] Kumar M., S and Singh S 2018 A Survey on Virtual Machine Scheduling Algorithms in Cloud Computing Int. J. Comput. Sci. Eng. 6 485–90.

[2] Kumar M 2019 Priority-Based Virtual Machine Selection Algorithm in Cloud Computing Int. J. Recent Technol. Eng. 8 1457–62.

[3] Gupta S and Sanghwan S 2015 Load Balancing in Cloud Computing: A Review 4 5

[4] Poonam and Suman 2018 LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING ENVIRONMENT Int. J. Adv. Res. Comput. Sci. 9 397–401

[5] Kaur T and Chana I 2015 Energy Efficiency Techniques in Cloud Computing: A Survey and Taxonomy ACM Comput. Surv. 48 1–46.

[6] Adhikary T, Das A K, Razzaque Md A and Sarkar A M J 2013 Energy-Efficient Scheduling Algorithms for Data Center Resources in Cloud Computing 2013 IEEE 10th International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing 2013 IEEE International Conference on High Performance Computing and Communications (HPCC) & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (EUC) (Zhangjiajie, China: IEEE) pp 1715–20

[7] Chen H, Zhu X, Guo H, Zhu J, Qin X and Wu J 2015 Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment J. Syst. Softw. 99 20–35

[8] Rao R V, Sivasani V J and Vakharia D P 2011 Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems Comput.-Aided Des. 43 303–15

[9] Rashedi E, Nezhambadi-pour H and Saryazdi S 2009 USA: A Gravitational Search Algorithm Inf. Sci. 179 2232–48

[10] Hatamlou A 2013 Black hole: A new heuristic optimization approach for data clustering Inf. Sci. 222 175–84

[11] Mirjalili S 2016 SCA: A Sine Cosine Algorithm for solving optimization problems Knowl.-Based Syst. 96 120–33

[12] Erol O K and Eksin I 2006 A new optimization method: Big Bang–Big Crunch Adv. Eng. Softw. 37 106–11

[13] Eskandar H, Sadollah A, Bahreininejad A and Hamdi M 2012 Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems Struct. 110–111 151–66

[14] Moosavian N and Kaseee Roodsari B 2014 Soccer league competition algorithm: A novel meta-heuristic algorithm for optimal design of water distribution networks Swarm Evol. Comput. 17 14–24

[15] Kumar M, Kulkarni A J and Satapathy S C 2018 Socio evolution & learning optimization algorithm: A socio-inspired optimization methodology Future Gener. Comput. Syst. 81 252–72

[16] Cui Y F, Li X M, Dong K W and Zhu J L 2011 Cloud Computing Resource Scheduling Method Research Based on Improved Genetic Algorithm Adv. Mater. Res. 271–273 552–7

[17] Shojafar M, Javanmardi S, Abolfazli S and Cordeschi N 2015 FUGE: A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method Clust. Comput. 18 829–44

[18] Cho K-M, Tsai P-W, Tsai C-W and Yang C-S 2015 A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing Neural Comput. Appl. 26 1297–309

[19] Al-Olimat H S, Alam M, Green R and Lee J K 2015 Cloudlet Scheduling with Particle Swarm Optimization 2015 Fifth International Conference on Communication Systems and Network Technologies 2015 Fifth International Conference on Communication Systems and Network Technologies (CSNT) (Gwalior, India: IEEE) pp 991–5

[20] Wen X, Huang M and Shi J 2012 Study on Resources Scheduling Based on ACO Algorithm and PSO Algorithm in Cloud Computing 2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science 2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science (Gujin, China: IEEE) pp 219–22

[21] Yang H 2014 Improved Ant Colony Algorithm Based on PSO and its Application on Cloud Computing Resource Scheduling Adv. Mater. Res. 989–994 2192–5

[22] Abdullahi M and Ngadi M A 2016 Hybrid Symbiotic Organisms Search Optimization Algorithm for Scheduling of Tasks on Cloud Computing Environment ed J Bongard PLOS ONE 11 e0158229

[23] Ben Alla S, Ben Alla H, Touhafi A and Ezzati A 2019 An Efficient Energy-Aware Tasks Scheduling with Deadline-Constrained in Cloud Computing Computers 8 46

[24] Abdullahi M, Ngadi M A, Dishing S I, Abdulhamid S M and Ahmad B I 2019 An efficient symbiotic organisms search algorithm with chaotic optimization strategy for multi-objective task scheduling problems in cloud computing environment J. Netw. Comput. Appl. 133 60–74

[25] Nayak S C, Parida S, Tripathy C and Pattanai K P K 2018 An enhanced deadline constraint based task scheduling mechanism for cloud environment J. King Saud Univ. - Comput. Inf. Sci. S1319157818304270

[26] Chen W, Xie G, Li R, Bai Y, Fan C and Li K 2017 Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems Future Gener. Comput. Syst. 74 1–11

[27] Arabnejad H and Barbosa J G 2014 A Budget Constrained Scheduling Algorithm for Workflow Applications J. Grid Comput. 12 665–79

[28] Jena R K 2017 Energy Efficient Task Scheduling in Cloud Computing Environment Procedia Comput. Sci. 113 222–7

[29] Lawanyashri M, Balusamy B and Subha S 2017 Energy-aware hybrid fruitfly optimization for load balancing in cloud environments for EHR applications Inform. Med. Unlocked 8 42–50.

[30] Zhu X, Hussain M and Li X 2019 Energy-Efficient Independent Task Scheduling in Cloud Computing Human Centered Computing Lecture Notes in Computer Science vol 11354, ed Y Tang, Q Zu and J G Rodriguez Garcia (Cham: Springer International Publishing) pp 428–39.
[31] Du G, He H and Meng Q 2014 Energy-Efficient Scheduling for Tasks with Deadline in Virtualized Environments Math. Probl. Eng. 2014 1–7
[32] Calheiros R N, Ranjan R, Beloglazov A, De Rose C A F and Buyya R 2011 CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms Softw. Pract. Exp. 41 23–50
[33] Kumar M and Sharma S C 2018 PSO-COGENT: Cost and energy efficient scheduling in cloud environment with deadline constraint Sustain. Comput. Inform. Syst. 19 147–64
[34] Jing W, Zhao C, Miao Q, Song H and Chen G 2021 QoS-DPSO: QoS-aware Task Scheduling for Cloud Computing System J. Netw. Syst. Manag. 29 5
[35] Hayyolalam V and Kazem A A P 2020 Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems Eng. Appl. Artif. Intell. 28
[36] Madni S H H, Abd Latiff M S, Abdulkhamid S M and Ali J 2019 Hybrid gradient descent cuckoo search (HGDCS) algorithm for resource scheduling in IaaS cloud computing environment Clust. Comput. 22 301–34