Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Crow Search Algorithm (CSA) for Task-Scheduling in Cloud Computing

Arvind Upadhyay[1], Ramesh Thakur[2], Archana Thakur[3]

[1]Research Scholar, Institute of Engineering & Technology Devi Ahilya Vishwavidyalaya, Indore &
IPS Academy, Institute of Engineering & Science, Indore
[2]International Institute of Professional Studies Devi Ahilya Vishwavidyalaya, Indore
[3]School of Computer Science Devi Ahilya Vishwavidyalaya, Indore
Email: upadhyayarvind10@gmail.com

A significant breakthrough in the IT sector has emerged in cloud computing. Utilizing cloud facilities as services has made this approach possible for users. In the world of cloud computing, scheduling tasks are a major issue. A scheduler of tasks has required in cloud computing to efficiently assign the tasks to the appropriate virtual machines (VMs). Through the use of a task scheduler, tasks in cloud computing are properly mapped to the right VMs to reduce make-span time. To overcome the scheduling issue in cloud computing, numerous scholars in the research have employed evolutionary algorithms. For scheduling tasks on the cloud, the CSA (crow search algorithm) is suggested in this research. It takes its cues from crows' propensity for gathering food. In truth, the crow is constantly keeping an eye on its fellow birds to discover a stronger source of food beyond its present one. By doing this, the CSA reduces the make-span and locates a VM that is appropriate for the task. Cloud-sim is used in studies to evaluate the effectiveness of the CSA in comparison with the ACO and Min-Min algorithms. The outcomes of the simulation show that the CSA algorithm outperforms the ACO and Min-Min algorithms.
Keywords: Cloud computing, task scheduling, make-span, task scheduler, and crow search algorithm.

1. Introduction

Cloud computing (Abdullahi and Ngadi 2016), has evolved in the fields of computing and preserving by offering countless ways to access informational technologies in a broad variety of areas like networks, mobile systems, environmental computation, pharmaceuticals, and businesses. This resulted in the choice to employ cloud computing in more recent years due to its efficiency in providing IT solutions. To avoid spending money to manage the IT equipment, it also contains pay-per-use facilities (Ebadifard and Babamir 2018).

Task scheduling, which is still a hard problem, is a common issue in cloud computing. In an IaaS cloud, several servers are made available to give clients the tools they need. The IaaS cloud offers consumers not just hardware yet also software that guarantees the flexibility and efficient administration of cloud supplies (Pradeep and Prem Jacob 2018). The resource administration subsystems in IaaS are utilized to carry out the duties that need to be performed in a highly effective way of VM describing the dynamic and heterogeneous characteristics. Heuristic techniques are used to find the best strategy for the problem (Krishnadoss and Jacob 2018). The basic goal is to minimize the scheduling of tasks both the cost and the make-span time. Due to the virtual machines (VMs) ability for heterogeneous computation, load management between VMs is possible. It is utilized to accomplish shorter make-span, task scheduling, and give synchronization. The algorithms of task scheduling, therefore, control VM processing time and load balance (Zhong et al 2016).

The algorithms of heuristic and metaheuristic optimization are widely accessible. Trial and error are the basis of heuristic algorithms, while metaheuristic algorithms operate at a higher scale (Ismael et al 2020). The metaheuristic algorithms seem to be those that get their inspiration from nature. The four categories of metaheuristic algorithms nature-inspired include human, swarm, physics, and evolution-oriented methods. The instances of those algorithms are genetic (GA), ACO (ant colony optimization), CSA (crow search algorithm), CS (cuckoo search) (Krishnadoss et al 2021), PSO (particle swarm optimization), etc (Pasandideh and Khalilpourazari 2018). Figure 1 depicts the various classification of metaheuristics optimization.

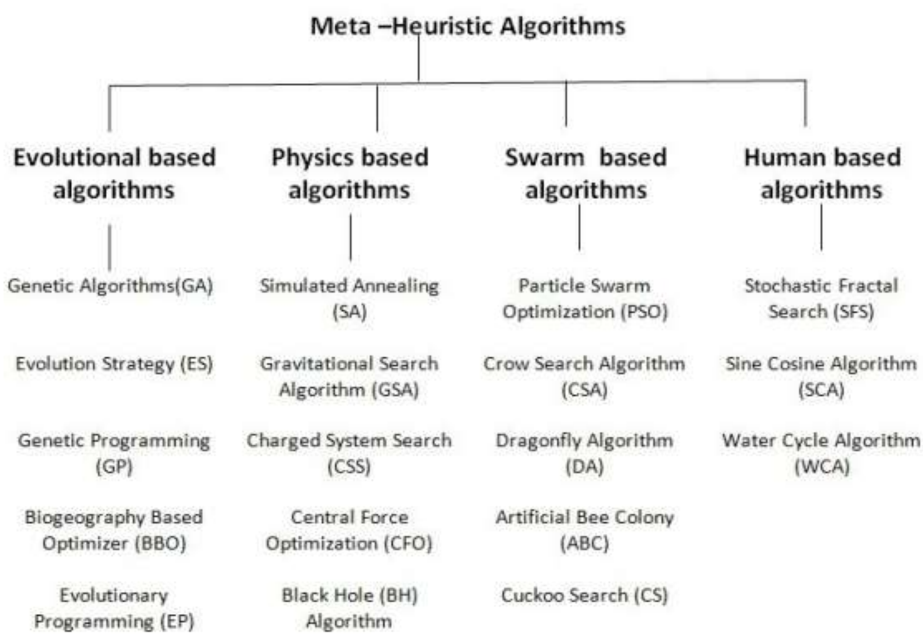

Figure 1: The classification of algorithms of meta-heuristic

1.1 Research Objective

The primary objective of this research is,

- To schedule tasks in cloud computing based on nature-inspired algorithm CSA
- To examine the make-span value of CSA in comparison with ACO and Min-Min algorithm

1.2 Research Layout

Therefore, the subsequent of this research article is structured as below. The review of the literature is presented in section 2, and the research methodology is examined in section 3. The outcomes are illustrated in Section 4 and evaluated. The research article's conclusion is depicted in the last section 5.

2. Literature Review

Numerous algorithms derived from nature are used to solve task scheduling issues. To reduce the make-span it takes for tasks to complete in cloud computing, an algorithm for task scheduling is developed (Huang et al 2020). Employing the PSO, this algorithm is represented. To achieve rapid convergence, this method employed a logarithmic diminishing strategy. It is contrasted to the synthetic bee colony and dragonfly algorithms that are currently in use, and the suggested approach outperforms the current algorithms in terms of convergence and make-span. Crow search optimization(Alaa Sheta et al 2023) can efficiently optimize many problems, it needs more searchability and early convergence. They enhanced the location updating mechanism by allowing two adaptive parameters: flight duration (fl) and awareness probability (AP) in order to control the CSA's exploration and exploitation activities in the search space. This approach exploits the use of well-known growth functions and the randomization of crows in CSA.

To maximize tool use and reduce overall work completion duration, researchers (Singh et al 2018) suggested a task scheduling method. To map the tasks onto the cloud services, this program uses a GS as a paradigm. The suggested method outperforms fundamental round-robin and GAs when it is executed on cloud-sim as well as put up against them concerning the given metrics.

Generally speaking, the majority of scheduling algorithms focus on the make-span reduction or execution time (Mapetu et al 2019). The scheduling algorithm then assigns the tasks to the most appropriate resources. Sometimes, a small number of VMs might be sufficient for the majority of workloads. Therefore, those VMs are given a greater level of tasks. As a result, the VM strain would rise, which would lower the VM's functionality. The PSO (Mapetu et al 2019) task scheduling method is used to handle the load disparity difficulties. PSO algorithm takes into account both execution time and VM load. Whenever the task hits the optimum limit, it would not be assigned to the asset. PSO enhances the VM's functionality in this manner.

Asper Kanisha et al 2018, the cuckoo bird's reproductive cycle serves as the foundation for Cuckoo Search. Since cuckoos hatch their eggs in the nests of different birds (hosts), breeding in them is difficult. The host bird tosses the cuckoo's eggs or shifts its eggs to a fresh nest whenever it notices significant differences in the eggs. Therefore, the cuckoo must locate a nest whereby the host bird and cuckoo eggs vary slightly from each other. The viable VMs for the execution of tasks is found by using the CS for scheduling tasks.

3. Methodology

3.1 Formulation of Problem

Let's take into account the k number of tasks, i.e t1-tk, and the n virtual machines, i.e V1-Vn. The virtual machines of n that have been active in j datacenters must be mapped towards the k count of tasks. The data centers are denoted by the letters d1-dj. These tasks of k have been originally presented to the task organizer; following task upload, the entry tasks need to be processed to determine the VMs' and tasks' relative priorities depending on the expense of power per unit at the relevant data centers.

By computing the priorities of both VMs and tasks while lowering the expenses of power and the tasks of make-span at the data centers, this research may describe the issue to ensure that the relevant tasks ought to be mapped effectively into the compatible virtual machines.

3.2 Crow Search Algorithm

As per Askarzadeh (2016),in the natural world, crows are intelligent birds. They possess a unique method of gathering food and storing it in certain locations. The locations where their food has been kept are easy for them to recall. Because of how they behave around other crows, they can recall other crows' faces and caution other crows. They are cunning enough to converse with fellow crows, and they are observant enough to know wherein other crows have been keeping their food. In cases whereby other crows attempted to thieve food from them, they may also protect it.The majority of the time, crows reside in bunches. They pursue fellow crows to locate various people's food storage areas. In this method, every crow would retain the whereabouts of the place of food for each repetition.Askarzadeh suggested a brand-new population-oriented algorithm termed as Crow Search Algorithm (CSA), which resembles a crow's habit of collecting and storing food. Additionally, Figure 2 teaches us something about the exploration and utilization of CSA.

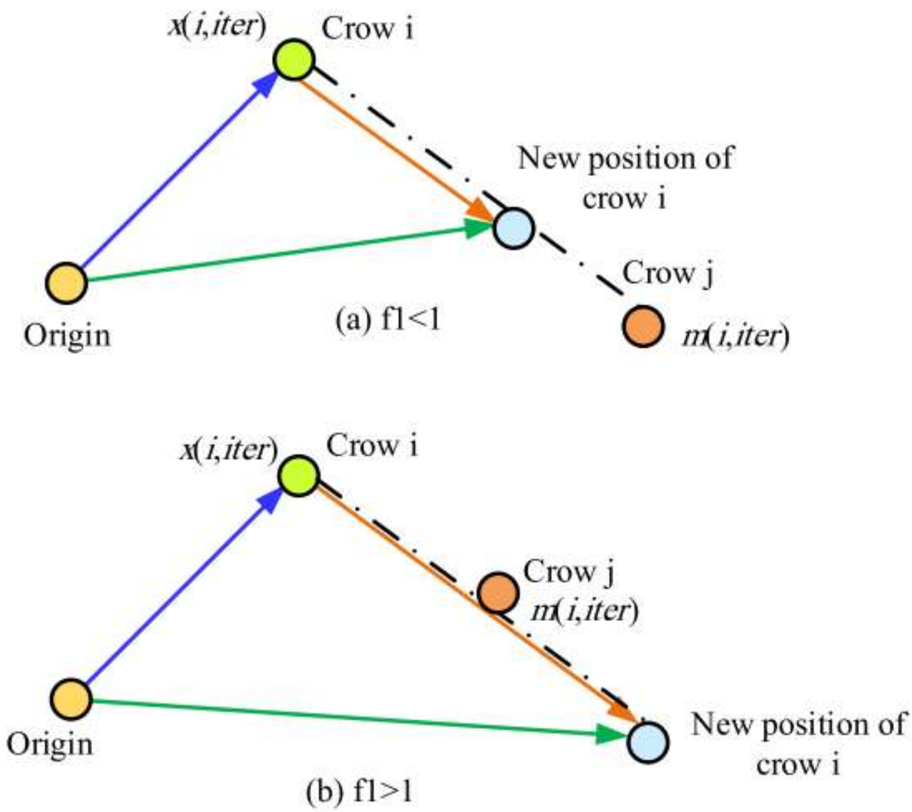

Figure 2: CSA exploration and utilization

Ultimately, Algorithm 1's representation of the CSA pseudocode could be employed to represent it. Figure 4 depicts the CSA flow diagram and highlights its key stages are described in the following:

1) Starting the d-dimensional flock of crows at randomness.

2) Every crow is evaluated using a fitness function, and its result is stored as the initial level for every cell of memory. The memory parameter mi contains the location of each crow's hideout.

3) Crow upgrades its location by choosing another crow randomly, i.e., x_j, and producing a randomized figure.

Crow x_i would pursue x_j to find out what $m_j$is when this number is higher than AP-(Awareness Probability).

4)Crow updates its location by picking an unrelated crow randomly, say x_j, and accompanying it to find m_j. Then, fresh x_j is determined as beneath:

$$x_{i,iter+1} = \begin{cases} x_{i,iter} + r_i \times fl_{i,iter} \times (m_{j,iter} - x_{i,iter}) & r_j \geq AP_{j,iter} \\ \text{Randomly} & \text{otherwise} \end{cases} \quad (1)$$

whereby $fl_{i,iter}$is the distance of the crow I flight to represent the crow j memories and $AP_{j,iter}$ is the crow j AP, iter is the number of iterations, and r_i,r_j is the randomized total.

5)Memory updating

$$m_{i,iter+1} = \begin{cases} x_{i,iter+1} & f(x_{i,iter+1}) \leq f(m_{i,iter}) \\ m_{i,iter} & \text{otherwise} \end{cases} \quad (2)$$

Algorithm 1 CSA: Crow Search Algorithm

Input: *n* Number of crows in the population.
$iter_{max}$ Maximum number of iteration.
Output: Optimal crow position
Initialize position of crows.
Initialize crows' memory
while $iter < iter_{max}$ **do**
  for $crow_i$belong to crows **do**
    choose a random crow.
    determine a value of awareness probability AP
    Update $x_{i,iter+1}$ using **Eq.**(1)
  end for
  Check solution boundaries.
  Calculate the fitness of each crow
  Update crows' memory using **Eq.**(2)
end while

Figure 3: Algorithm 1 for CSA

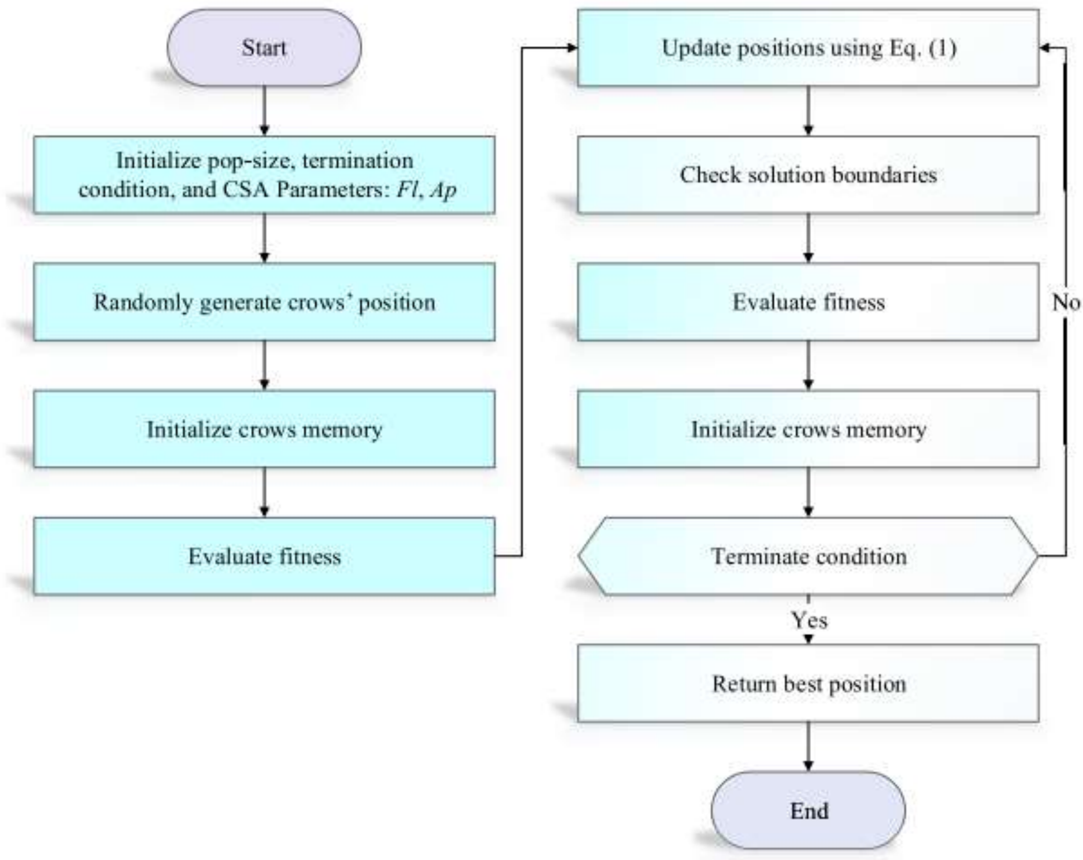

Figure 4: CSA flow diagram

4. Result and Discussion

Dynamically responding to user requests, the cloud offers services. It is challenging to create such a setting for testing purposes. There are numerous cloud-based open-source applications accessible for investigation purposes. Among similar open-source program that is utilized by the majority of investigators to conduct trials is Cloud-sim [29]. In cloud-sim, CSA is performed; the simulation setting is listed in below table 1:

Table 1: Settings of simulation

Entity	Quantity
Cloudlets	100-1000
Task lengths	500000 to 200000000
VM capacity	4096MB
Number of VMs	100
Number of hosts	500
Physical host RAM capacity	32GB
The capacity of processing elements	174/247/355 MIPS

Tasks and virtual machines (VMs) are diverse in a practical cloud context [30]. VM heterogeneity is indeed the term used to describe how different VMs would undertake tasks [28] at different periods. Three alternative consistencies are employed, including inconsistent, consistent, and moderately consistent, to enable the tasks of VM mappings more natural [28]. The VMj performs any tasks quicker than the VMk in consistent (c), and after that, the VMj performs all tasks quicker than the VMk. When contrasted to VMk, the VMj will indeed perform some activities more quickly while performing others more slowly in an inconsistent (ic) situation. The intersection of inconsistent as well as consistent instances is known as the partly consistent (pc).There are four categories for the tasks and VM heterogeneous nature: Low Task High VM, High Task High VM, Low Task Low VM, and High Task Low VM. The 2 current algorithms ACO and Min-Min algorithm have been selected to assess the functionality of CSA. The 12 examples' make-span results are recorded after running the CSA, ACO, and Min-Min algorithms.

There are situations whereby tasks call for a lot of power for processing and the VMs are powerful enough to handle them. For the instances of consistent HTHVM, inconsistent HTHVM, and partly consistent HTHVM, Figure 5 displays the make-span readings for the CSA, ACO, and Min-Min algorithms. The fact that primarily the ACO make-span readings and CSA algorithms show minimal changes could be attributed to VMs' powerful computational capabilities. The outcomes demonstrate how CSA reduces the schedule's make-span.

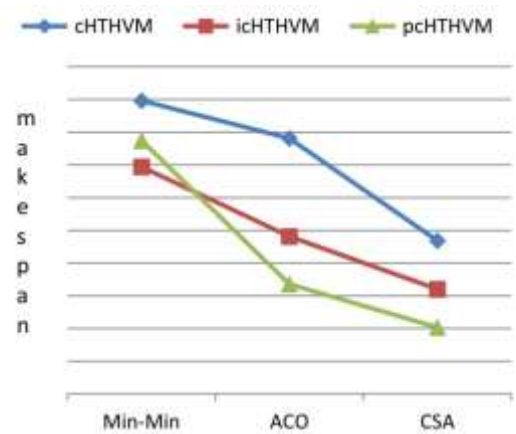

Figure 5: Graph of HTHVM

While the VMs in the HTLVM scenario have minimal processing capability, the tasks are highly computationally intensive. However, even though the work can still be completed by the VM, it does so slowly.Therefore, the algorithm used for scheduling must determine which VM is most suited for every task. The performances of the algorithms of CSA, Min-Min, and ACO for the HTLVM scenario are shown in Figure 6. In comparison to ACO and Min-Min, CSA raises the make-span level.

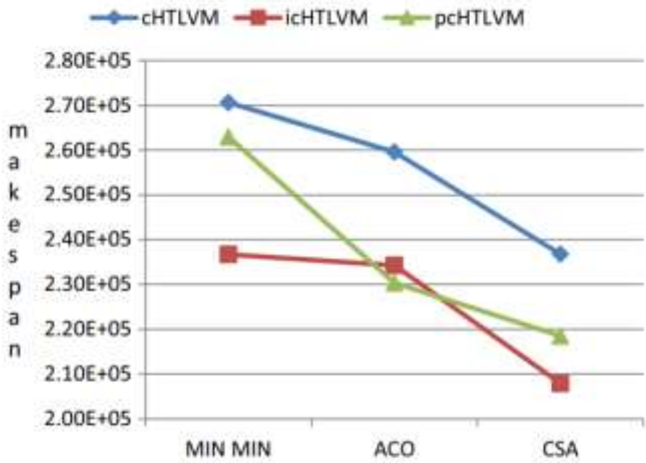

Figure 6: Graph of HTLVM

When using LTLVM, tasks require little computer power, hence the VMs are set up with less processing power. The evaluation of the performance of LTLVM is displayed in Figure 7. The make-span level of CSA in consistent LTLVM is closer to the level of make-span of ACO.

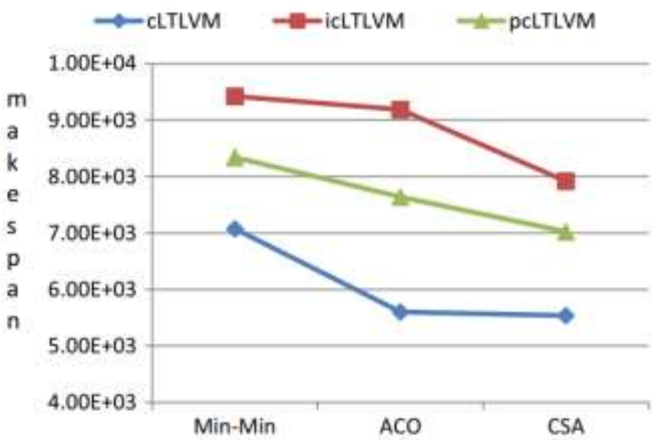

Figure 7: Graph of LTLVM

The task is regarded as having a reduced computational need in LTHVM and the VM as having higher computational capabilities. The evaluation of the make-span of the inconsistent, consistent, and partly consistent LTHVM is shown in Figure 8. The CSA performs higher than ACO and Min-Min.

It is evident from the outcome evaluation that CSA provides a 5 to 15 percent increase in make-span value over ACO. The difference between CSA and Min-Min is around 12 and 20%. Though it consistently provides smaller make-span levels, the CSA occasionally generates make-span values that are closer to those of the ACO. The CSA subsequently finetunes the schedule with every iteration by locating a different VM to perform the task. Therefore, the completion time of the task is reduced, which is seen in the make-span result.

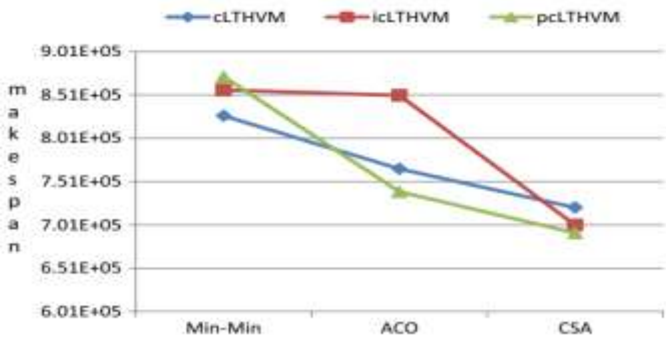

Figure 8: Graph of LTHVM

5. Conclusion

In cloud computing, task scheduling is incredibly difficult. The mapping of tasks into the virtual machines was the researchers' primary concern while developing the current algorithms, which took into account parameters such as make-span and resource consumption. In this research, we present a novel task scheduling method that evaluates the task to determine the task and virtual machine priorities. The scheduler must effectively map the activities to the virtual machines following determining these priorities by reducing the make-span and power consumption at the Datacenters. A natural-inspired algorithm called "crow search" was used to model this one.A newly created algorithm, called CSA, models how crows store and retrieve food. Because of CSA's exceptional qualities, investigators have provided it a lot of emphasis and attention. As a metaheuristic algorithm, CSA is particularly effective in locating the ideal option within a search area.

The scheduling of task issues in cloud computing is addressed via CSA. The make-span decrease in scheduling tasks is emphasized by the suggested, nature-inspired CSA. Empirical studies are carried out, and make-span estimates for the CSA, ACO, and Min-Min algorithms are derived depending on the 12 examples of tasks and heterogeneity of VM. Depending on the findings of the suggested CSA using the Min-Min and ACO algorithms, a comparison investigation has indeed been conducted.According to the outcomes, CSA shortens the make-span value more than Min-Min and ACO algorithms. In this research, CSA is used to schedule tasks with predetermined flight lengths. The subsequent location is determined by the distanceof the flight. The solution area can be substantially extended in the long term by dynamically varying the flight time, and localized search algorithms can be used to identify the alternative VM in place of randomized selections.

References

1. J Sheta, A., Braik, M., Al-Hiary, H. and Mirjalili, S., (2023). Improved versions of crow search algorithm for solving global numerical optimization problems. Applied Intelligence, 53(22),

pp.26840-26884.

2. Abdullahi, M., &Ngadi, M. A. (2016). Hybrid symbiotic organisms search optimization algorithms for scheduling tasks inthe cloud computing environment. PloS one, 11(6), e0158229. https://doi.org/10.1371/journal.pone.0162054
3. Ebadifard, F., &Babamir, S. M. (2018). A PSO-based task scheduling algorithm was improved using a load-balancing technique for the cloud computing environment. Concurrency and Computation: Practice and Experience, 30(12),e4368. https://doi.org/10.1002/cpe.4368
4. Pradeep, K., & Prem Jacob, T. (2018). A hybrid approach for task scheduling using the cuckoo and harmony search in a cloud computing environment. Wireless Personal Communications, 101, 2287-2311. https://doi.org/10.1007/s11277-018-5816-0
5. Krishnadoss, P., & Jacob, P. (2018). OCSA: task scheduling algorithm in a cloud computing environment. International Journal of Intelligent Engineering and Systems, 11(3), 271-279. https://doi.org/ 10.22266/ijies2018.0630.29
6. Zhong, Z., Chen, K., Zhai, X., & Zhou, S. (2016). Virtual machine-based task scheduling algorithm in a cloud computing environment. Tsinghua Science and Technology, 21(6), 660-667. https://doi.org/10.1109/TST.2016.7787008
7. Ismael, A. N., Al-Asady, M. A. A., & Al-Asadi, H. A. A. (2020). A Critical Comparative Review of Nature-Inspired Optimization Algorithms (NIOAs). International Journal of Simulation--Systems, Science & Technology, 21(3). https://doi.org/ 10.5013/IJSSST.a.21.03.09
8. Krishnadoss, P., Pradeep, N., Ali, J., Nanjappan, M., Krishnamoorthy, P., &KedaluPoornachary, V. (2021). CCSA: Hybrid cuckoo crow search algorithm for task scheduling in cloud computing. International Journal of Intelligent Engineering and Systems, 14(4), 241-250. https://doi.org/ 10.22266/ijies2021.0831.22
9. Pasandideh, S. H. R., &Khalilpourazari, S. (2018). Sine cosine crow search algorithm: a powerful hybrid metaheuristic for global optimization. arXiv preprint arXiv:1801.08485. https://doi.org/10.48550/arXiv.1801.08485
10. Huang, X., Li, C., Chen, H., & An, D. (2020). Task scheduling in cloud computing using particle swarm optimization with time-varying inertia weight strategies. Cluster Computing, 23, 1137-1147. https://doi.org/10.1007/s10586-019-02983-5
11. Singh, K., Kulshrestha, S., Kulshreshtha, A., & Sharma, S. (2018). Optimization of task scheduling using genetic algorithm in a federated cloud environment. International Journal of Computer Engineering and Sciences Research, 1, 24-30. https://ijcesr.com/Openaccess/v1i1/C.pdf
12. Mapetu, J. P. B., Chen, Z., & Kong, L. (2019). Low-time complexity and low-cost binary particle swarm optimization algorithm for task scheduling and load balancing in cloud computing. Applied Intelligence, 49, 3308-3330. https://doi.org/10.1007/s10489-019-01448-x
13. Kanisha, B., Lokesh, S., Kumar, P. M., Parthasarathy, P., & Chandra Babu, G. (2018). Speech recognition with improved support vector machine using dual classifiers and cross-fitness validation. Personal and ubiquitous computing, 22, 1083-1091. https://doi.org/10.1007/s00779-018-1139-0
14. Askarzadeh, A. (2016). A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Computers & structures, 169, 1-12. https://doi.org/10.1016/j.compstruc.2016.03.001.
15. W. Alomoush, A.Alrosan,Y.M. Alomari, A.A. Alomoush,A. Almomani and H.S. Alamri, " Fully automatic grayscale image segmentation based fuzzy C-means with firefly mate algorithm", Journal of Ambient Intelligence and Humanized Computing, vol.13,no.9, pp.4519-4541,2022.