International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Minimization of Makespan and Energy Consumption in Task Scheduling in Heterogeneous Cloud Environment

R. Priyadarshini*[1], Mukil Alagirisamy[2], N. Rajendran[3], Dr.Arun Kumar Marandi[4], Vikas Vilasrao patil[5], Dr. Vivek [6]

Submitted: 12/08/2022 **Accepted:** 23/11/2022

Abstract: In today's IT industry, cloud computing is one of the technologies that is increasingly being used for regular corporate operations. The cloud is becoming more and more popular among businesses and research communities due to its many benefits, including on-demand self-service, quality of service, pay-per-usage pricing, virtualization, and elasticity. This research propose novel technique in makespan reduction with energy consumption for task scheduling. Here the cloud environment task scheduling for makespan reduction is carried out using gravitational grey wolf hybrid cuckoo scheduling. The simulation is suggested in the cloudsim programming environment, and the outcomes demonstrated the value of the energy-minimizing and makespan parameters. The simulation is suggested in the cloudsim programming environment, and the outcomes demonstrated the value of the energy-minimizing and makespan parameters. Proposed technique attained makespan of 67%, energy efficiency of 96%, execution speed of 79%, resource utilization of 63%, average waiting time of 66% for 500 number of tasks.

Keywords: *Cloud computing, energy consumption, makespan reduction, task scheduling, cloudsim*

1. Introduction

The use of cloud computing allows for pay-as-you-go access to in-demand IT resources from any location at any time. A typical datacenter in cloud computing consists of a few computers connected via fast networks. This setting is ideal for the computation of numerous different types of large-scale workloads. Different users' tasks are no longer distinguishable from one another. In this situation, the scheduling problem is to assign additional jobs to run on the available processing machines [1]. The Cloud computing is characterised by three specific traits; 1) To enable a high degree of agility and scalability that meets business demands, unlimited computing resources, such as processing speed, data storage capacity, and applications, are available on demand as needed. 2) There are no long-term commitments; because computing resources are purchased on a month-to-month or even minute-to-minute basis, they are immediately available and may be used for as long as necessary before being decommissioned. 3) Pay-as-you-go pricing; as there are no long-term contracts, the price of cloud computing resources is based on how much is used. Task scheduling is the process of allocating n customers' jobs to m resources or clouds in such a way as to reduce the overall processing time, or makespan [2]. Be aware that n > m. Here, the clients' demands differ in terms of the quantity of resources, the price, the timeframe, etc. The resources, however, differ in terms of processing speed, capacity, bandwidth, service level, etc. As a result, the performance of the customer task varies depending on the resource. In heterogeneous environments like cloud computing systems, where the main goal is to reduce the makespan, it poses the difficulty of resource selection for each activity [3]. The execution of some jobs predominates over the completion of others, hence minimising makespan does not always equate to customer happiness. As a result, client happiness must be prioritised in work scheduling. Instead of concentrating on the overall makespan of all the jobs, it must concentrate on minimising the individual makespan of the activities [4].

The contribution of this research is as follows:

1. To propose novel technique in makespan reduction with energy consumption for task scheduling.
2. the cloud environment task scheduling for makespan reduction is carried out using gravitational grey wolf hybrid cuckoo scheduling.

2. Related Works

Numerous techniques and heuristics have been put out to address the scheduling issue in cloud systems while taking a variety of factors into account, such as cost, makespan, and energy

[1] Post-Doctoral Fellow, Computer Science Engineering, Faculty of Engineering,Lincoln University College, Malaysia. rspdarshini@gmail.com
[2] Associate Professor, Department of Electrical and Electronics Engineering, Faculty of Engineering, Lincoln University College, Malaysia, mukil.a@lincoln.edu.my
[3] Assistant Professor (Sr.Gr.), Department of Information Technology, B.S.Abdur Rahman Crescent Institute of Science and Technology, Chennai, India, rajendran.n81@gmail.com
[4] Assistant Professor, Department of Computer Science, Arka Jain University, Jamshedpur, Jharkhand, India. Email Id- dr.arun@arkajainuniversity.ac.in,
[5] Vikas Vilasrao patil, Assistant Professor, Bharati Vidyapeeth (Deemed to be University) Y. M. Institute of Management Karad Vikas.patil@bharatividyapeeth.edu
[6] Associate Professor, Department of Computer Science Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, v.vullikanti@jainuniversity.ac.in

reduction. The authors of [5] provide a multi-model estimation distribution parallel application approach that aims to save time and energy. The suggested technique outperforms parallel bi-objective genetic algorithms and heuristics in terms of energy conservation and makespan minimization. However, neither the heterogeneity of the cloud nor the element of uncertainty is taken into account in the paper. As opposed to this, the authors of [6] suggest an energy aware Min-Min algorithm (EAMM) that aims to reduce both energy consumption and processing time. When compared to the original Min-Min method, the algorithm typically produces better results. The developers of [7] created a real-time dynamic scheduling system with the intention of reducing the amount of energy and time needed to complete tasks for cloud-based task-based applications. The authors of [8] put forth a strategy meant to reduce the amount of energy used in cloud data centres. In contrast, the authors of [9] suggested the fuzzy dominance sort-based heterogenous earliestfinish-time (FDHEFT) algorithm to reduce the cost and duration of operations in IaaS Clouds. Additionally, the authors of [10] proposed a heuristic to reduce cloud energy, throughput, and time. The outcomes demonstrate that it is capable of achieving the three goals in a positive manner. Additionally, each individual target produces greater outcomes when compared to other algorithms. A task meet-to-deadline strategy has been devised and used in [11] to improve the makespan time in cloud systems. The Author did not take other QoS metrics into account. In [12], the author designed and implemented a method to reduce the overall execution time and obtain a decent schedule length in a cloud environment, but the load balancing in the cloud system caused the author to suffer. In [13], the author designed and put into practise work scheduling methods to enhance the cloud-based Min-Min algorithm while decreasing the overall completion time.

3. System Model

This section discuss novel technique in makespan reduction with energy consumption for task scheduling. Here the cloud environment task scheduling for makespan reduction with energy efficiency is carried out using gravitational grey wolf hybrid cuckoo scheduling.

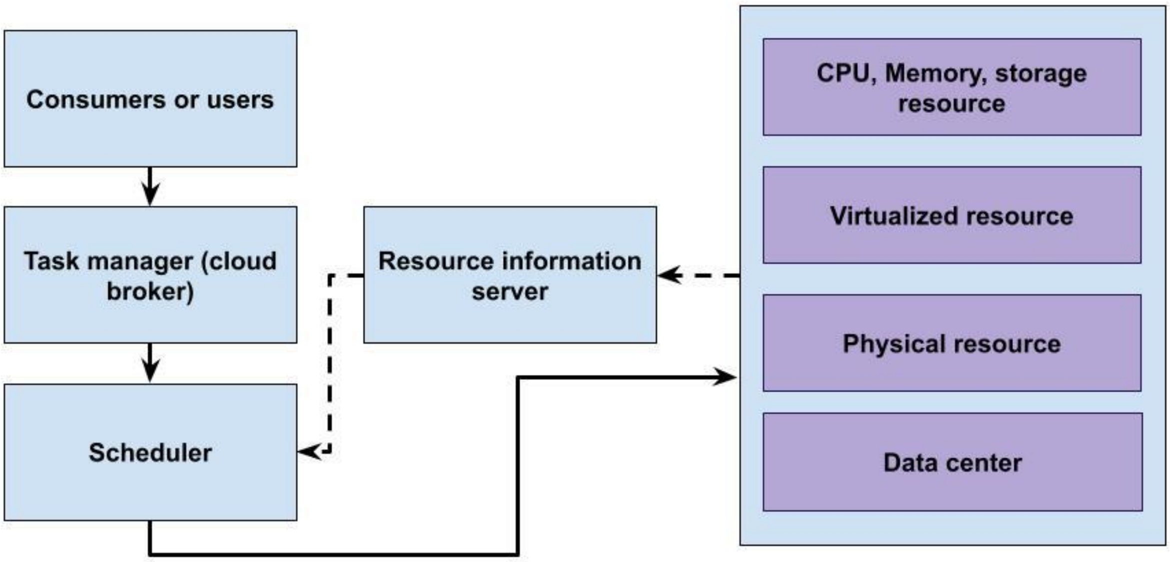

Figure-1 Proposed task scheduling architecture

Makespan reduction with energy efficiency by gravitational grey wolf hybrid cuckoo scheduling (GGWHCS):

$$T_{\theta_n} = \frac{L_{\theta_n}}{VM_{i,j}(mips)} \quad (1)$$

where Ln is the task's difficulty, which is often stated in terms of millions of instructions (MI). The computation power of V Mi,j is expressed as V Mi,j (mips), where MIPS is the unit. The working time of the V Mi,j is determined by adding the execution times of all jobs on the V Mi,j, which is done using the formula in equation (2) because the tasks on the VM are queued.

$$T_{VM_{i,j}} = \sum_{\theta_n \in \Theta_i} T_{\theta_n} \quad (2)$$

Prey that is being circled by grey wolves during a hunt might be adapted using the following equation (3).

$$\vec{D} = \left|\vec{C} \cdot \vec{X}_p(t) - \theta \cdot (\vec{X}(t))\right| \quad (3)$$

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (4)$$

$$\vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha - \theta \cdot \vec{X}(t)\right| = \vec{X}_p(t) - \vec{A} \cdot \vec{D}\ \vec{C} = 2 \cdot \vec{r}_2$$

$$\vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta - \theta \cdot \vec{X}(t)\right|$$

$$\vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta - \theta \cdot \vec{X}(t)\right|$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha)$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta)$$

(5)

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}$$

This research proposes a minimum makespan scheduling system, which is displayed in Fig. 1. The task queue is where the framework keeps the tasks after they have been received from various users. The tasks are regarded as independent, pre-emptible tasks with equal importance. There are n tasks and m virtual machines. With a minimum makespan and minimum resource consumption rate, "n" jobs are scheduled on "m" VMs.

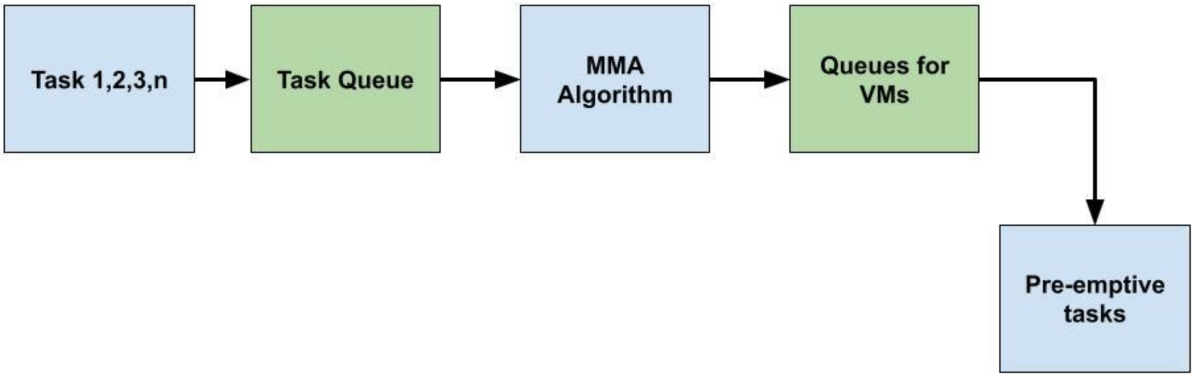

Figure 2. Minimum makespan scheduling framework

The duty Scheduling issues are thought of as multi-objective optimization issues [36]. Minimizing the makespan and raising the VM usage rate are the goals. The cuckoo search algorithm is used to solve optimization problems, and it has been found to perform better than other meta heuristic algorithms. Each and every egg in the nest represents a solution; the cuckoo egg stands for a fresh solution. The primary motivation of the cuckoo egg is to find the best option and replace the less-than-ideal solutions in the nests. One egg is included in every nest. The following guidelines govern cuckoo searches: 1. Each cuckoo only lays one egg at a specific time, after which the egg is discarded in a randomly selected nest. 2. The best nest is used to produce high-quality eggs for the upcoming generation. 3. In most cases, the hosts' nests are fixed, and the likelihood that a cuckoo bird will lay an egg depends on the host bird (0, 1). After determining this, we can proceed to work on the worst nests; the resultant solution is then discarded for more calculations.

$$\beta \geq \left(\frac{n}{m} - L(n/m)]\right) \times m$$
(6)

In this case, the proposed algorithm requires less than or equal to (/ n m) iterations as (n – β) < (/ n m) × m by eq. (7).

$$\beta < \left(\frac{n}{m} - \lfloor(n/m)\rfloor\right) \times m$$
(7)

4. Experimental Analysis

The focus of this part is on the computational tests that are used to gauge how well the suggested method works. The cloudsim tool has been used to simulate the suggested approach. This toolkit's underlying platform is based on Java. All of these tests have been verified on a computer with an Intel(R) Core(TM) i5-457 processor, four CPUs running at 2.9 GHz, eight GB of RAM, and a 64-bit Windows operating system. Here, we analyse the simulation results that used the least amount of time and energy. Table 1 is a representation of the parameters. Two separate datasets are used in this experiment to test the effectiveness of the suggested technique. They have 100–500 tasks in each distribution, which is left and right skewed. In contrast to right skewed, which has more small sized activities and fewer large sized duties, left skewed has fewer small sized tasks and larger sized tasks.

Table 1: Experimental settings

Parameters	Value
Number of data center	5
Number of host	10
Host memory capacity	10 GB
Host bandwidth	2800 Mbps
Number of VMs	50
VM policy	Time_shared
VMM	Xen
Number of vCPU	[1-5]
Task MIPS	[200-15000]

Table-2 Comparison of proposed and existing method

Parameters	**EAMM**	**FDHEFT**	**MMEC_TS_HCE**
Makespan	63	66	67
Energy efficiency	89	93	96
Execution speed	71	76	79
Resource utilization	58	61	63
Average waiting time	63	63	66

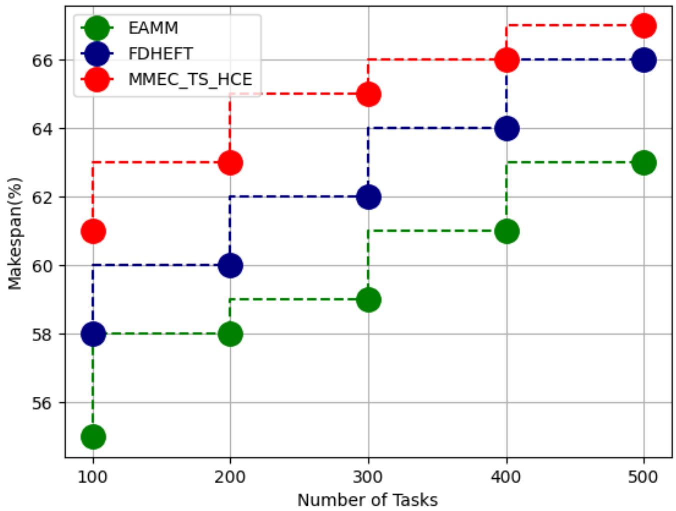

Figure-3 Comparison of Makespan

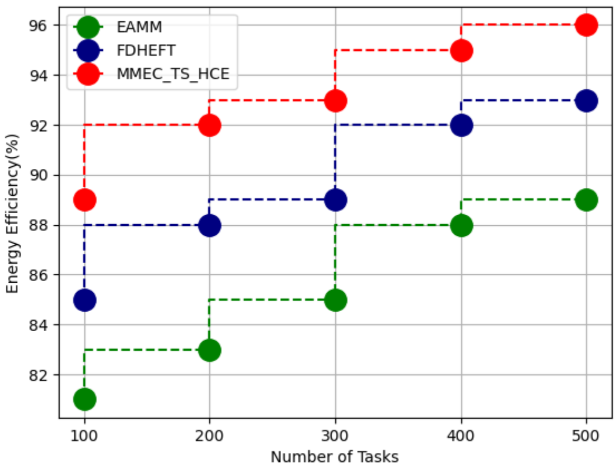

Figure-4 Comparison of Energy Efficiency

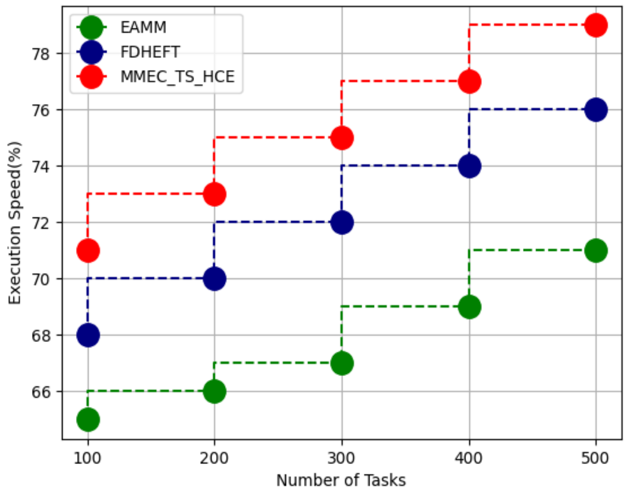

Figure- 5 Comparison of Execution speed

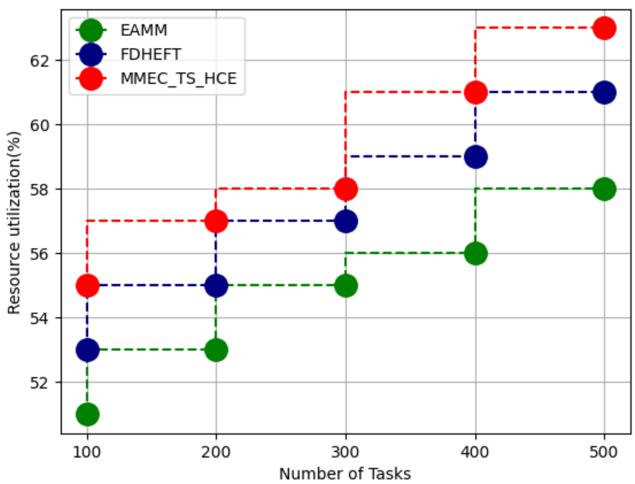

Figure- 6 Comparison of Resource utilization

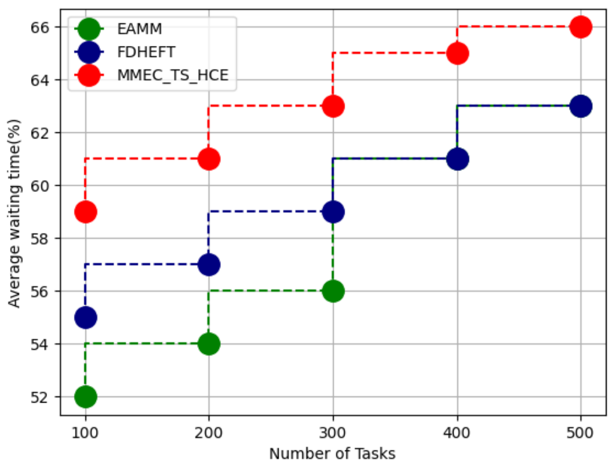

Figure- 7 Comparison of Average Waiting time

The projected mean-performance GWO's outcome is assessed for makespan. For task scheduling, the proposed approach is contrasted with EAMM and FDHEFT. The left and right skewed distributions are the datasets that were utilised. There are between 100 and 500 missions total. The tasks in this simulation are run 30 times, and the average is then determined. We take into account 50 VMs for each of these datasets. Similar to the left skewed, the makespan values for 200 cloud tasks in the right skewed are 103.45, 111.13, and 105.57. However, the makespan numbers are 232.96, 256.02, and 243.22 when there are 400 cloud tasks. The number of tasks and the task scheduling algorithms are both configurable in this experiment, whereas the parameters of the VMs and PMs are fixed. The power consumption of the suggested algorithm is contrasted with that of alternative algorithms for various task counts. The degree of freedom, abbreviated df, in test results is determined by calculating the difference between the number of samples and tests. Here, because the t statistic value is greater than the t critical one-tail and the t critical two-tail, we reject the null hypothesis for each dataset. We assert that the population means are not equal as a result. This is primarily supported by the fact that both p-values—namely, p one-tail and p two-tail—are extremely low (i.e., less than 0.05). The suggested method achieved 500 tasks with an average wait time of 66%, a makespan of 67%, energy efficiency of 96%, execution speed of 79%, resource utilisation of 63%, and.

5. Conclusion

Users of the cloud hope to do their activities without delay, while cloud service providers hope to lower the cost of energy, which is one of the biggest expenses in the environment. However, cutting back on energy use lengthens the makespan and causes client unhappiness. Therefore, it is crucial to find a set of non-dominance solutions for these various and incompatible goals (makespan and energy consumption). So this research propose novel technique in makespan reduction with energy consumption for task scheduling which is carried out using gravitational grey wolf hybrid cuckoo scheduling. the proposed technique attained makespan of 67%, energy efficiency of 96%, execution speed of 79%, resource utilization of 63%, average waiting time of 66% for 500 number of tasks. Future work will focus on enhancing the suggested algorithms for scheduling the related IoT tasks. We also want to assess how well the suggested algorithms work on various real-world datasets.

References

[1] Shukla, D. K., Kumar, D., & Kushwaha, D. S. (2021). Task scheduling to reduce energy consumption and makespan of cloud computing using NSGA-II. *Materials Today: Proceedings*.

[2] Azizi, S., Shojafar, M., Abawajy, J., & Buyya, R. (2022). Deadline-aware and energy-efficient IoT task scheduling in fog computing systems: A semi-greedy approach. *Journal of network and computer applications*, *201*, 103333.

[3] Mangalampalli, S., Swain, S. K., & Mangalampalli, V. K. (2022). Multi Objective Task Scheduling in Cloud Computing Using Cat Swarm Optimization Algorithm. *Arabian Journal for Science and Engineering*, *47*(2), 1821-1830.

[4] Ibrahim, I. M. (2021). Task scheduling algorithms in cloud computing: A review. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(4), 1041-1053.

[5] Mangalampalli, S., Swain, S. K., & Mangalampalli, V. K. (2021). Prioritized Energy Efficient Task Scheduling Algorithm in Cloud Computing Using Whale Optimization Algorithm. *Wireless Personal Communications*, 1-17.

[6] Mukherjee, D., Nandy, S., Mohan, S., Al-Otaibi, Y. D., & Alnumay, W. S. (2021). Sustainable task scheduling strategy in cloudlets. *Sustainable Computing: Informatics and Systems*, *30*, 100513.

[7] Emami, H. (2022). Cloud task scheduling using enhanced sunflower optimization algorithm. *ICT Express*, *8*(1), 97-100.

[8] Ijaz, S., Munir, E. U., Ahmad, S. G., Rafique, M. M., & Rana, O. F. (2021). Energy-makespan optimization of workflow scheduling in fog–cloud computing. *Computing*, *103*(9), 2033-2059.

[9] Shukri, S. E., Al-Sayyed, R., Hudaib, A., & Mirjalili, S. (2021). Enhanced multi-verse optimizer for task scheduling in cloud computing environments. *Expert Systems with Applications*, *168*, 114230.

[10] Mubeen, A., Ibrahim, M., Bibi, N., Baz, M., Hamam, H., & Cheikhrouhou, O. (2021). Alts: An Adaptive Load Balanced Task Scheduling Approach for Cloud Computing. *Processes*, *9*(9), 1514.

[11] Kaur, R., & Laxmi, V. (2022). Performance evaluation of task scheduling algorithms in virtual cloud environment to minimize makespan. *International Journal of Information Technology*, *14*(1), 79-93.

[12] Abdel-Basset, M., Mohamed, R., Abouhawwash, M., Chakrabortty, R. K., & Ryan, M. J. (2021). EA-MSCA: An effective energy-aware multi-objective modified sine-cosine algorithm for real-time task scheduling in multiprocessor systems: Methods and analysis. *Expert systems with applications*, *173*, 114699.

[13] Nanjappan, M., Natesan, G., & Krishnadoss, P. (2021). An adaptive neuro-fuzzy inference system and black widow optimization approach for optimal resource utilization and task scheduling in a cloud environment. *Wireless Personal Communications*, *121*(3), 1891-1916.

.