Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

Performance Analysis of the Normalized Distribution and Ranking with Optimization Based Task Scheduling Techniques

O.S. Abdul Qadir [1], Dr. G. Ravi [2]

[1]Research Scholar in Computer Science, Jamal Mohamed College (Autonomous),
(Affiliated to Bharathidasan University), Tiruchirappalli – 620020.
[2]Associate Professor& Head, Department of Computer Science, Jamal Mohamed College (Autonomous),
(Affiliated to Bharathidasan University), Tiruchirappalli – 620020.
[1]abdulqadir@jmc.edu

ABSTRACT

Cloud computing is becoming more popular as a way to pay for IT services. Many IT service providers use cloud computing in their day-to-day operations. Mood services are located at swing locations in cloud computing. Because of the system's regional spread, operating actions, and heterogeneity of resources, resource supervision and scheduling become a hidden claim. User satisfaction is increased by completing cloud computing scheduling. In cloud computing, efficient task scheduling reduces the time it takes to get a system up and running. The client's requirement for QoS is the key motivator for task scheduling. The task with the high QoS requirement is scheduled after the task with the low QoS requirement. Users have enough resources to pay for facilities depending on utilization period, so the aim of task scheduling is to reduce costs by shortening the makespan era. A comparison of task scheduling algorithms using optimization techniques such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Whale Optimization Algorithm is presented in this paper (WOA). In addition, for reducing the makespan in a cloud setting, this paper proposed a Normalized Distribution and Ranking task scheduling method.

Keywords
Cloud computing, Task Scheduling, Optimization based Techniques, Particle Swarm optimization, Genetic Algorithm, Antlion Optimizer, Differential Evolution.

1. Introduction

After Cluster and Grid computing, Cloud computing has been distributed as utility computing in a manner close to the distribution of conventional services such as water, electricity, gas and telephony [1][2][3]. Cloud computing is described by the National Institute of Standards and Technology as a model for enabling or enhancing convenient, on-demand network access to a common pool of configurable computing resources such as software, storage, networks, servers, and services that can be quickly provisioned and released with minimal management effort or service provider involvement. [4][5][6][7][8][9] Cloud computing is a framework that encompasses both deployment and distribution models, as well as the five basic characteristics [6] [7]. Cloud computing offers a slew of advantages for both businesses and users of their services. More businesses, institutes, and consumers in need of computing services are migrating to the cloud as a result of these advantages.

Resource management has been one of the most persistent problems of cloud computing for decades. It is a procedure that deals with the acquisition, release, and control of resources.

Resources are virtualized and shared by multiple cloud users in cloud environments [8][9]. The findings of task scheduling in cloud computing, on the other hand, were not promising [10]. Load balancing, energy efficiency, resource management, execution time, and response time are only a few of the task scheduling challenges.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

1.1. Task Scheduling

Task scheduling [11][12] is a critical problem in a cloud setting. It is used to plan tasks for better resource use by allocating specific tasks to specific resources at specific times. The main goal of a task scheduling algorithm is to increase service performance and quality while also preserving task productivity and lowering costs. Task scheduling makes the best use of available virtual tools. Cloud computing can be done at a high level thanks to effective resource scheduling. Completion time, task completion cost, and other parameters are all taken into account by scheduling algorithms. In cloud computing, task scheduling is an NP-hard problem that is evaluated using meta-heuristic approaches such as PSO, GA, and others.

1.2. Scheduling Factors

Some of the usual factors in any form of task scheduling are presented below

1.2.1. Makespan:

Makespan is the total execution time of the scheduling to complete the execution of all tasks, defined from the time when the request arrives to the time of the completion of the last task, or it can be termed as the total time taken by the resource which finishes last.

1.2.2. Load Balancing:

Resource utilization is the process of calculating the usage of the individual resources in the completion of the execution of all the tasks in the scheduling process. Load balancing refers to the equal or near to equal utilization of all the resources.

1.3 Scheduling Methods

The task scheduling is an NP-hard problem. The scheduling methods are either heuristic or Meta heuristic. The heuristic methods include First Come First Served (FCFS), Opportunistic Load Balancing (OLB), Minimum Execution Time (MET), Minimum Completion Time (MCT), Min-Min, Max-Min, Resource Aware Scheduling Algorithm (RASA), etc.,. The Meta heuristic techniques are based on the inspiration derived from the behavior of nature. These techniques include Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), League Championship Algorithm (LCA), BAT algorithm, Cuckoo scheduling algorithm, Crow search algorithm and so on. A comparative study of some of these algorithms is given by A. Jain and A. Upadhyay [13].

2. Literature Review

Abualigah, Laith, and Ali Diabat [14] presented a novel hybrid antlion optimization algorithm for solving multi-objective task scheduling problems in cloud computing environments using elite-based differential evolution. The multi-objective nature of the problem, which is referred to as MALO in the proposed process, stems from the need to simultaneously reduce makespan while optimizing resource utilization. To boost its exploitation potential and avoid getting stuck in local optima, the antlion optimization algorithm was improved by using elite-based differential evolution as a local search technique. Khorsand, Reihaneh, and Mohammadreza Ramezanpour [15] centered on best-worst (BWM) and the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

methodology, proposed an energy-efficient task-scheduling algorithm The primary goal of this paper is to decide which cloud scheduling solution should be chosen. The assessment criteria are first defined by a decision-making group. Since the chosen criteria are of varying significance, a BWM method is used to assign importance weights to each criterion. TOPSIS then uses these weighted parameters as inputs to determine and assess each alternative's success.

Ziyath, S. Peer Mohamed, and S. Senthilkumar [16] proposed a metaheuristic optimization strategy with load balancing to increase the performance of cloud infrastructure service providers by eliminating scheduling issues. The proposed methodology is applicable to both static and dynamic task conditions, with static methods VM parameters set and dynamic means parameters chosen at runtime. For dealing with the static and dynamic properties of the task submitted, the proposed algorithm has two phases: MHOS-S and MHO-D.

Praveenchandar, J., and A. Tamilarasi [17] for an effective dynamic resource allocation mechanism, an improved task scheduling and an optimized power minimization approach were proposed. The efficiency of resource allocation in terms of task completion and response time is achieved using a prediction mechanism and a dynamic resource table updating algorithm. Since it eliminates data center resource consumption, this framework provides an effective outcome in terms of power reduction. For updating the resource table, the proposed method provides correct values. An enhanced task scheduling strategy and a reduced power consumption approach are used to achieve effective resource allocation.

Shukri, Sarah E., et al [18] Cloud computing is a popular technology that allows users to rent computing services on a pay-per-use basis from a remote location. Task scheduling is one of the most challenging challenges in cloud computing environments, as tasks must be planned effectively to minimize execution time and expense while optimizing resource usage. As a better task scheduler, an Enhanced Version of Multi-Verse Optimization (EMVO) was proposed. In cloud environments, the proposed EMVO is compared to both the original MVO and the PSO algorithm.

Alsaidy, Seema A., Amenah D. Abbood, and Mouayad A. Sahib [19] the use of heuristic algorithms to improve particle swarm optimization (PSO) initialization is suggested. The PSO is initialized using the longest task to fastest processor (LJFP) and minimum completion time (MCT) algorithms. The proposed LJFP-PSO and MCT-PSO algorithms are evaluated in terms of their ability to decrease the makespan, total execution time, degree of imbalance, and total energy consumption metrics.

In [20], the authors proposed and implemented a new heuristic scheduling methodology called, novel heuristic-based task scheduling [NHBTS] that produced a better makespan compared with many of the standard heuristic scheduling algorithms including Min-min.

In [21], a new scheduling technique called, Dual Objective Task Scheduling Algorithm in Cloud Environment (DOTS) is proposed that has a better makespan value along with good utilization of resources.

Velliangiri, S., et al [22] proposed Hybrid Electro Search with a Genetic Algorithm (HESGA) to improve task scheduling behavior by taking into account parameters like makespan, load balancing, resource usage, and multi-cloud cost. A genetic algorithm and an electro search algorithm were combined in the proposed process. The best local optimal solutions are provided by the genetic algorithm, while the best global optimal solutions are provided by the Electro search algorithm.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

Rjoub, Gaith, Jamal Bentahar, and Omar Abdel Wahab [23] BigTrustScheduling is a three-stage trust-aware scheduling approach that includes VMs' trust level computation, task priority level determination, and trust-aware scheduling.

Chen, Xuan, et al [24] to boost the WOA-based method's optimal solution search capability, researchers suggested an advanced approach called Improved WOA for Cloud Task Scheduling (IWC). The authors provided a thorough implementation of IWC, and our simulation-based experiments show that the proposed IWC outperforms current metaheuristic algorithms in terms of convergence speed and accuracy when looking for optimal task scheduling plans.

Sanaj, M. S., and PM Joe Prathap [25] In an Infrastructure as a Service (IaaS) cloud environment, a chaotic squirrel search algorithm (CSSA) was suggested for optimal multitask scheduling. The methods produce work plans on a continuous basis, making existing approaches more cost-effective. The early eco system was created with messy optimisation for the productive eco-system in order to ensure greater global convergence. To allow the exploring authority to complement Squirrel search algorithm (SSA) algorithms, the suggested chaotic squirrel search algorithm was eventually synthesised with the messy local search algorithm. Other QoS criteria, such as compatibility and protection for very large situations, may be added to the proposed technique to cover it.

Abd Elaziz, Mohamed, and Ibrahim Attiya [26] for optimum task scheduling, researchers presented an updated Henry gas solubility optimization (HGSO) based on the whale optimization algorithm (WOA) and a detailed opposition-based learning (COBL).

3. PROBLEM STATEMENT

Task scheduling has long been regarded as one of the most difficult aspects of cloud resource management. It's worth noting that QOS-based optimization task scheduling is an important feature of both cloud users' and cloud service providers' service level agreements (SLAs). Several QOS optimization strategies have been proposed to solve task scheduling problems, which include convergence, imbalance, and computational complexity among the selected nodes, based on the literature reviewed. For a long time, metaheuristic techniques that are known as either bio-inspired or swarm intelligence have been used to solve NP-hard task scheduling problems that impose certain constraints. The majority of the strategies mentioned above do not discuss how to achieve a global or local solution. Premature convergence or entrapment at the local optimum contributed to the poor output or result obtained.

4. OPTIMIZATION BASED TASK SCHEDULING TECHNIQUES

4.1. PSO based Task Scheduling

PSO is an evolutionary algorithm that simulates bird flocking to find food and fish schooling to defend themselves from predators. In PSO, and solution candidate is referred to as a particle that is traveling through a search space. The number of particles in the search space is represented by population in PSO. Each particle's velocity guides the movement of flying particles in search space. The location of a particle in a population is determined by its best position (pbest) and the position of the best particle (gbest). The entire population is started at random. In each generation, the fitness value of particles, which is used to measure their efficiency, is evaluated

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

and optimized. The velocity and location of the particle are changed as follows in each generation: $V_{k+1}^{i} = W_k V_k^i + c_1 r_1 \left(P_k^i - X_k^i\right) + c_2 r_2 \left(P_k^g - X_k^i\right) X_{k+1}^i = X_k^i + V_{k+1}^i$

The variables $r1$ and $r2$ are random numbers ranging from 0 to 1. The acceleration co-efficient are $c1$ and $c2$. The inertia weight is W. The velocity of particle I at iteration k is called Vik. The current location of particle I at iteration k is represented by Xki. The classic PSO uses three components to update a particle's velocity: inertia of previous velocity provides momentum and controls the balance between exploration and exploitation in the search space, social component represents particle cooperation in moving towards the global best position found in the search space, and cognitive component represents the particle's privacy in the search space.

4.2 GA based Task Scheduling

The Genetic Algorithm (GA) is based on the biological idea of population generation.
[1] [2] GA is a rapidly developing field of Artificial Intelligence. The Genetic Algorithms were influenced by Darwin's theory of evolution (GAs). The phrase "survival of the fittest" is used in Darwin's theory to define a scheduling system in which tasks are allocated to resources based on the importance of the fitness function for each parameter of the task scheduling mechanism [13]. The GA's fundamental concepts are as follows [1] [2]:

- Initial Population: The initial population is made up of all of the people who will be included in the GA to find the best solution. Every population solution is referred to as a person. Every individual is interpreted as a chromosome in order to facilitate genetic operations. Individuals are chosen from the initial population, and operations are performed on them to create the next generation. The mating chromosomes are chosen using a set of parameters.
- Fitness Function: Any individual's efficiency is determined by their level of fitness. It is a metric for determining an individual's population dominance. The health value depicts an individual's success in comparison to the rest of the population. As a consequence, individuals live or die depending on their fitness or function importance. As a consequence, the GA's motivating factor is the fitness function.
- Selection: Centered on Darwin's law of survival, the selection process is used to choose an intermediate option for the next generation. This operation serves as the GA's performance-based guiding channel. The best chromosomes can be chosen using a variety of methods, including the roulette wheel, Boltzmann technique, tournament selection, and rank selection.
- Crossover: Selecting two parent individuals and then alternating and reforming the sections of those parents to create a new individual tree is a crossover process. In the GA, the hybridization operation is a guiding method that improves the searching mechanism.
- Mutation: Mutation occurs as a result of crossover. The operator is the one that adds genetic diversity to the population. If a population becomes homogeneous as a result of repeated use of replication and crossover operators, mutation occurs. It happens during evolution based on a user-defined mutation probability, which is typically set to a low value. A mutation changes the values of one or more genes in the chromosome from their original state. This will result in the addition of completely new gene values to the gene pool.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

4.3. Whale Optimization Algorithm (WOA) based Task Scheduling

A humpback whale in the search space is a candidate solution in the optimization problem, also known as a search agent, in the WOA algorithm, and the WOA uses a collection of search agents to find the feasible or approximately global optimal solution. The quest for a solution to a given problem starts with a set of random solutions, and the candidate solution is modified using optimization rules until the end condition is satisfied. Encircling prey, bubble-net attack, and quest for prey are the three key stages of the WOA algorithm.

1. *Encircling Preying:* When the prey is surrounded, humpback whales do not know the best location in the search space at first. The goal prey in WOA is the current best option, and the whale nearest to the prey is considered the best search agent. Then, as other whales approach the target prey, they will progressively update their positions.
2. *Bubble-Net Attack (Exploitation Phase):* The action of whales attacking with bubble nets is modeled using shrinking encircling and spiral location updating.

Search for Prey (Exploration Phase): As |A| > 1, the search agents are pulled away from each other to ensure that an approximately global optimal solution can be found. In this case, the location of the current optimal search agent will be replaced by a randomly chosen search agent.

5. PROPOSED NORMALIZED DISTRIBUTION AND RANKING IN TASK SCHEDULING

The order in which tasks are assigned to resources has a greater effect on the makespan in task scheduling. As a result, the ETC matrix is normalized to help with this. The tasks are scheduled to the resources using the greedy method after the ETC matrix has been normalized.

A. Objective

The execution time of the resources against the task is represented using an ETC matrix. Fair scheduling is required, with the goal of reducing the total completion time of the tasks listed in the ETC matrix.

The duration of the makespan is the resource's maximum finish time.

Let FT be a set consisting of the finishing time of the resources.

$$\mathrm{FT} = \{\mathrm{FT[R1]}, \mathrm{FT[R2]}, \mathrm{FT[R3]}, \ldots\ldots \mathrm{FTR[n]}\}$$

FT[R] = Finish time of the resource r i.e. the total execution time of all the tasks allocated to it. It is represented by the following equation.

$$\mathrm{FT[R]} = \left(\sum_{i=1}^{n} (ETC[i][R])\right)$$

Where, n is the number of tasks allocated to the resource R. Using this equation (3), the finish time of all the resources is calculated. Then the makespan is represented by the equation

$$\text{Makespan} = \max\{\mathrm{FT}\}$$

B. Ranking of Tasks by Normalization

As a result, a rating is needed to decide the minimum duration of the makespan. The entropy in the ETC matrix must be overcome in order to formulate an algorithm that meets the objective. The data set in the ETC matrix is normalized to eliminate entropy by scaling the values between 0 and 1. The equation is used to normalize the values.

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

$$X' = \frac{\hat{x} - Xmin}{Xmax - Xmin}$$

Where, X' is the normalized value of x. Xmax and Xmin are the maximum and minimum values in the considered data range.

In task scheduling, normalized distribution and ranking focuses on the sequences in which tasks are allocated to the resources. After normalizing the values in the ETC matrix using the above-mentioned equation (5), the task sequence is calculated. The normalized values are given a ranking. The order of sequencing is calculated based on this rating. -

C. Selection of Resource

The activities with the highest rating are chosen first. It will be given to the resource with the shortest execution time. The resources are then chosen using the greedy approach to reduce the makespan. After choosing tasks based on their rating, the greedy approach assigns them to resources with the shortest cumulative execution time. Equation is used in a greedy way to generate the shortest makespan from the initial assignment to the assignment of the remaining tasks to resources (7). Let L[R] be the duration of a resource's execution at a specific point in time. The following is the resource Rx that will be assigned to the next task I

$$Rx = \min\{L[1]+ETC[i], L[2]+ETC[i][2], \ldots\ldots L[n]+ETC[i][n]\}$$

So that the mapping of a task to a resource x has the minimum total execution time when compared with others. It can be expressed as

$$\text{Mapping of Task}[i] = \text{Min}\left(\sum_{j=1}^{r} CET[j] + ETC[i][j]\right)$$

Where, CET is the Current Execution time of the resource.

Rx is the resource to be allocated for the tasks that are being chosen based on their rating in equation (6). The resources are allocated to the tasks in a similar manner until there are no more tasks to complete. As a result of the foregoing explanations, algorithm 1 for task scheduling to resources was developed. The makespan of the scheduling is calculated by this algorithm, which is known as Normalized Distribution and Ranking in Task Scheduling (NRTS). After scaling the values between 0 and 1 using equation, the function Normalize of Algorithm 2 assigns ranks to the tasks (5). Algorithm 3's ResSel feature handles the collection of suitable resources for the graded tasks. On an ETC matrix with T tasks and R resources, the algorithm takes O time (TR). Note that the for loop in algorithm 3 takes o(TR) time to complete. As a consequence, the NRTS algorithm takes O(TR) time to simply decide the resources in the ETC matrix are being allocated for task execution, resulting in a reduction in execution time as shown in equation (4).

Algorithm NRTS (M, T, R)

```
// M is the ETC matrix with T number of tasks and R number of resources.
{
  For (i=1; i<=R; i++)
  {// Initialize the length of execution of resources
          L[i] = 0
  }
```

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

```
//Normalize the values of ETC Matrix ETCN =
Normalize (M)
//Selecting the Resource for the remaining tasks ResSel
(L[])
}
```

Algorithm 1

```
Algorithm Normalize (ETC)
//The matrix ETCN is the normalized values of M.
{
        For (i=1; i<=R; i++)
        {
                For (j=1; i<=T; j++)
                {// Let Xmax and Xmin are the maximum and minimum values of the resource i
                        ETCN[i][j] = (ETC[I][J] – Xmin) / (Xmax – Xmin)
                }
        }
        //Rank the tasks based on the normalized values
        //calculate the row total in ETCN.
        For (i=1; i<=T; i++)
        {
                For (j=1; i<=R; j++)
                {
                        RowSum[i] = (RowSum + ETCN[i][j])
                }
        }
        //Rank the tasks in non-increasing order
        last = T;
        For (i=1; i<T; i++)
        {
                for (j=i+1; j<=last; j++)
                {
                        if(a[j] < a[j+1])
                        {
                                a[j] ↔ a[j+1]
                        }
                        last=last-1;
                }
        }
}
```

Algorithm 2

```
Algorithm ResSel (L[])
{
        For(i=1;i<=T;i++)
        {
                SR = 1 //SR is the Selected Resource
                For (j=2; i<=R; j++)
                {// Select the resource for the task i
                        if(L[SR]+ETC[i][SR]>L[j]+ETC[i][j]
                        {
                                SR = j
                        }
                }
                L[SR]=L[SR]+ETC[i][SR]
```

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

```
        }
}
```

6. RESULT AND DISCUSSION

In this section, the performance of the optimization based task scheduling techniques like PSO-based, GA-based, WOA-based and proposed Normalized Ranking Task scheduling are evaluated with various metrics like Average Execution Time (in seconds), Utilization Rate (in %), Average Makespan (in seconds), and Average Response Time (in seconds) for the various tasks varying from 100 to 1000.

1. Cloud Simulator Parameter Configuration

Table 1 depicts the CloudSim test setting for evaluating the performance of the proposed NRTS, optimization techniques based Task Scheduling.

Table 1: CloudSim Configuration

Element	Parameter	Values
Data Center	No. of data-center	2
Cloudlet	No. of cloudlets	100–1000
	Length	1000–2000
Virtual machine	RAM	512 MB
	MIPS	100–1000
	Size	10000
	Bandwidth	1000
	Policy type	Time Shared
	No. of CPUs	1
Host	No. of Hosts	2
	RAM	2048 MB
	Storage	1 million
	Bandwidth	10000

Table 2 depicts the average makespan (in seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling algorithms for the varying number of tasks. From the table 2, it is clear that the proposed NRTS technique consumes less makespan for executing the tasks.

Table 2: Makespan (in Seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling techniques

Number of Tasks	Proposed NRTS	Number of Resources	Optimization based Task Scheduling Techniques		
			PSO	GA	WOA
100	0.51	7	0.79	0.88	0.59
200	0.61	10	1.119	0.95	0.88

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

300	0.84	12	1.49	1.26	1.25
400	1.29	14	1.74	1.54	1.62
500	1.27	15	1.84	1.57	1.54
600	1.24	16	1.84	1.92	1.82
700	1.66	18	2.27	2.19	1.93
800	2.07	19	3.15	2.81	2.55
900	2.75	20	3.99	3.96	3.88
1000	3.44	21	4.75	4.87	5.01

Table 3 depicts the average execution time (in seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling algorithms for the varying number of tasks. From the table 3, it is clear that the proposed NRTS technique consumes less average execution time for executing the tasks.

Table 3: Average Execution Time (in Seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling techniques

Number of Tasks	**Proposed NRTS**	**Number of Resources**	**Optimization based Task Scheduling Techniques**		
			PSO	GA	WOA
100	31	7	103	85	84
200	105	10	173	174	154
300	174	12	214	224	211
400	242	14	384	404	403
500	398	15	598	693	584
600	614	16	887	854	774
700	781	18	1127	1194	1151
800	962	19	1387	1374	1269
900	1015	20	1542	1447	1475
1000	1247	21	1796	1802	1836

Table 4 depicts the Utilization Rate (in %) by the proposed NRTS, GA, PSO and WOA based task scheduling algorithms for the varying number of tasks. From the table 4, it is clear that the proposed NRTS technique takes maximum utilization rate for executing the tasks.

Table 4: Utilization Rate (in %) by the proposed NRTS, GA, PSO and WOA based task scheduling techniques

Number of Tasks	**Proposed NRTS**	**Number of Resources**	**Optimization based Task Scheduling Techniques**		
			PSO	GA	WOA
100	13.5	7	9.12	12.22	11.41
200	29.6	10	11.54	21.25	17.77
300	42	12	22.35	26.25	31.54

Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, Pages. 4058 - 4070

Received 05 March 2021; Accepted 01 April 2021.

400	53.45	14	32.43	36.62	37.67
500	69.33	15	41.17	43.38	49.76
600	81.15	16	49.74	53.74	60.17
700	93.85	18	56.36	59.08	64.25
800	94.88	19	57.74	61.25	65.57
900	95.36	20	63.37	62.58	66.87
1000	97.71	21	65.89	66.89	67.26

Table 5 depicts the Average Response Time (in seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling algorithms for the varying number of tasks. From the table 5, it is clear that the proposed NRTS technique takes less time for response for executing the tasks.

Table 5: Average Response Time (in seconds) by the proposed NRTS, GA, PSO and WOA based task scheduling techniques

Number of Tasks	Proposed NRTS	Number of Resources	Optimization based Task Scheduling Techniques		
			PSO	GA	WOA
100	33.5	7	77	81	79
200	97.7	10	184	184	182
300	132	12	232	224	242
400	264	14	442	392	387
500	421	15	654	665	692
600	624	16	917	891	839
700	796	18	1189	1173	1121
800	885	19	1278	1309	1314
900	917	20	1406	1496	1455
1000	1044	21	1554	1624	1587

7. Conclusion

In this research article a new concept has been applied by normalizing the input values. A ranking is applied to the tasks after scaling the values. Based on this ranking the tasks are selected sequentially for execution. The NRTS technique performs better than the well-known min-min algorithm. The performance of the proposed NRTS technique is compared against optimization-based task scheduling techniques like PSO, GA and WOA. From the results obtained, it is clear that the proposed NRTS technique gives consumes less time for makespan, execution and response than the other techniques. It also gives maximum utilization rate in executing the tasks.

References

[1] Gawali, Mahendra Bhatu, and Subhash K. Shinde. "Task scheduling and resource allocation in cloud computing using a heuristic approach." *Journal of Cloud Computing* 7.1 (2018): 1-16.

[2] Panda, Sanjaya K., and Prasanta K. Jana. "An energy-efficient task scheduling algorithm for heterogeneous cloud computing systems." *Cluster Computing* 22.2 (2019): 509-527.

[3] Zhou, Zhou, et al. "An improved genetic algorithm using greedy strategy toward task scheduling optimization in cloud environments." *Neural Computing and Applications* 32.6 (2020): 1531-1541.

[4] Panda, Sanjaya K., and Prasanta K. Jana. "Load balanced task scheduling for cloud computing: A probabilistic approach." *Knowledge and Information Systems* 61.3 (2019): 1607-1631.

[5] Tom, Linz, and V. R. Bindu. "Task Scheduling Algorithms in Cloud Computing: A Survey." *International Conference on Inventive Computation Technologies*. Springer, Cham, 2019.

[6] Bacanin, Nebojsa, et al. "Artificial flora optimization algorithm for task scheduling in cloud computing environment." *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, Cham, 2019.

[7] Tong, Zhao, et al. "A novel task scheduling scheme in a cloud computing environment using hybrid biogeography-based optimization." *Soft Computing* 23.21 (2019): 11035-11054.

[8] Sofia, A. Sathya, and P. GaneshKumar. "Multi-objective task scheduling to minimize energy consumption and makespan of cloud computing using NSGA-II." *Journal of Network and Systems Management* 26.2 (2018): 463-485.

[9] Nayak, Suvendu Chandan, et al. "Task scheduling mechanism using multi-criteria decision-making technique, MACBETH in cloud computing." *Progress in Computing, Analytics and Networking*. Springer, Singapore, 2018. 381-392.

[10] Sarkhel, Preeta, Himansu Das, and Lalit K. Vashishtha. "Task-scheduling algorithms in cloud environment." *Computational Intelligence in Data Mining*. Springer, Singapore, 2017. 553-562.

[11] Nayak, Biswajit, Sanjay Kumar Padhi, and Prasant Kumar Pattnaik. "Static Task Scheduling Heuristic Approach in Cloud Computing Environment." *Information Systems Design and Intelligent Applications*. Springer, Singapore, 2019. 473-480.

[12] Singh, Poonam, Maitreyee Dutta, and Naveen Aggarwal. "A review of task scheduling based on meta-heuristics approach in cloud computing." *Knowledge and Information Systems* 52.1 (2017): 1-51.

[13] A. Jain and A. Upadhyay. Cloud Scheduling using Meta Heuristic Algorithms, *International Journal of Computer Sciences and Engineering (IJCSE)*, 5 (10), (2017).

[14] Abualigah, Laith, and Ali Diabat. "A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments." *Cluster Computing* (2020): 1-19.

[15] Khorsand, Reihaneh, and Mohammadreza Ramezanpour. "An energy-efficient task-scheduling algorithm based on a multi-criteria decision-making method in cloud computing." *International Journal of Communication Systems* 33.9 (2020): e4379.

[16] Ziyath, S. Peer Mohamed, and S. Senthilkumar. "MHO: meta heuristic optimization applied task scheduling with load balancing technique for cloud infrastructure services." *Journal of Ambient Intelligence and Humanized Computing* (2020): 1-10.

[17] Praveenchandar, J., and A. Tamilarasi. "Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing." *Journal of Ambient Intelligence and Humanized Computing* (2020): 1-13.

[18] Shukri, Sarah E., et al. "Enhanced multi-verse optimizer for task scheduling in cloud computing environments." *Expert Systems with Applications* (2020): 114230.

[19] Alsaidy, Seema A., Amenah D. Abbood, and Mouayad A. Sahib. "Heuristic initialization of PSO task scheduling algorithm in cloud computing." *Journal of King Saud University-Computer and Information Sciences* (2020).

[20] O.S. Abdul Qadir and Dr. G. Ravi, A Novel Heuristic Based Task Scheduling Algorithm to Minimize the Makespan in Cloud Environment, *International Journal of Advanced Science and Technology*, 29 (5), 3737- 3746 (2020).

[21] O.S. Abdul Qadir and Dr. G. Ravi, Dual Objective Task Scheduling Algorithm in Cloud Environment , *International Journal of Advanced Trends in Computer Science and Engineering*, 9 (3), (2020), *https://doi.org/10.30534/ijatcse/2020/07932020*.

[22] Velliangiri, S., et al. "Hybrid electro search with genetic algorithm for task scheduling in cloud computing." *Ain Shams Engineering Journal* (2020).

[23] Rjoub, Gaith, Jamal Bentahar, and Omar Abdel Wahab. "BigTrustScheduling: Trust-aware big data task scheduling approach in cloud computing environments." *Future Generation Computer Systems* 110 (2020): 1079-1097.

[24] Chen, Xuan, et al. "A woa-based optimization approach for task scheduling in cloud computing systems." *IEEE Systems Journal* 14.3 (2020): 3117-3128.

[25] Sanaj, M. S., and PM Joe Prathap. "Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere." *Engineering Science and Technology, an International Journal* 23.4 (2020): 891-902.

[26] Abd Elaziz, Mohamed, and Ibrahim Attiya. "An improved Henry gas solubility optimization algorithm for task scheduling in cloud computing." *Artificial Intelligence Review* (2020): 1-39