Optimizing Workload Scheduling in Cloud Paradigm using Robust Neutrosophic C-Means Clustering Boosted with Fish School Search

S. YUVARAJ GANDHI, T. REVATHI
Department of Computer Science,
PSG College of Arts and Science, Coimbatore, Tamilnadu,
INDIA

Abstract:- In the present internet world, accessing cloud resources for a low cost, according to their needs, is available to all users. Sharing resources is becoming increasingly necessary as people complete their activities in the cloud. It becomes essential for distributed workloads to be optimized to perform efficient workload scheduling and progressing resource utilization in a cloud environment. Scheduling cloud resources considerably benefits from the invention of machine learning and metaheuristic models to address this scenario. Though many existing algorithms are developed in cloud-based task scheduling using unsupervised clustering methods, the problem of unknown task requirements or resource availability in adverse conditions is still challenging. In this study, an uncertainty-based unsupervised technique is constructed to group incoming tasks according to the required resources, and it is scheduled to the most suitable resources more prominently. This paper introduced a Robust Neutrosophic C-Means Clustering boosted with the fish school search algorithm (RNCM-FSSA) for clustering the incoming tasks and the resources based on their requirement and availability. With the degree of indeterminacy, neutrosophic C-means discriminating the deterministic and indeterministic schemes and scheduling them to the optimal resources more effectively. Using the fitness value computed by FFSA, the potential cluster centroids are utilized for clustering, thus avoiding the early convergence in the grouping process. The simulation results explore that the robustness of the proposed RCNM-SSA achieves better resource utilization, the degree of imbalance is minimal, and computation complexity is also considerably decreased compared with other unsupervised models.

Key-Words: - Work Scheduling, Uncertainty, Neutrosophic C-Means clustering, Fish School Search Algorithm, indeterminacy.

Received: May 14, 2022. Revised: August 17, 2023. Accepted: September 19, 2023. Available online: November 14, 2023.

1 Introduction

Cloud computing has ushered in a transformative era in computing, offering a dynamic and scalable environment that combines a multitude of resources to meet the ever-growing demand for services. However, at the heart of this cloud paradigm lies a formidable challenge – efficient resource allocation. The relentless surge in service requests collides with the limited availability of resources, creating a pivotal concern, [1]. Consequently, a pressing need exists for the establishment of a robust mechanism to distribute task workloads effectively among the available resources.

As this challenge looms, and in response to the needs of cloud users, this study embarks on a journey to create a robust load-balancing-based resource scheduling policy. The overarching goals include a substantial reduction in task execution response times and the optimization of resource utilization, [2], [3]. Although various resource scheduling strategies are currently available, they are often tailored for homogeneous resource types, which can lead to inefficiencies when dealing with heterogeneous environments. This homogeneity-centric approach inevitably results in the cumbersome task of scanning the entire list of virtual machines for each incoming work request, [4].

In contrast to these challenges, our work harnesses the power of clustering techniques, designed to meet cloud service requirements effectively. Through the intelligent grouping of incoming jobs and virtual machines based on their capacity, this technique efficiently mitigates the overhead linked to the screening process. Consequently, it offers an effective solution to the challenges associated with resource scheduling in a heterogeneous environment and ensures the efficient distribution of workloads, [5]. Conventional clustering algorithms are often plagued by issues such as local optima and premature convergence due to suboptimal centroid selection. In response to these shortcomings, our current research places a strong emphasis on the imperative goal of optimizing load balancing. Achieving a well-balanced distribution of workloads across suitable cloud resources, however, becomes increasingly complex due to the inherent inconsistencies and indeterminacies associated with incoming task requests and the fluctuating availability of cloud resources, a common reality in real-time cloud applications, [6].

In addition to these novel approaches, our study provides several key benefits. It excels in optimizing load balancing, significantly reducing task execution times, and maximizing resource utilization. Furthermore, it effectively addresses the unique challenges posed by heterogeneous resource environments, streamlining workload distribution. Hence, in this paper, the knowledge of uncertainty is acquired by devising a robust model known as the Neutrosophic C-Means clustering model for grouping similar types of task requests and the available resource capacity by computing the degree of membership using truthiness, falsity, and indeterminacy. The centroid selection is accomplished by inducing the metaheuristic algorithm called fish school searching. The assignment of resources in an optimized way is explained in the following sections.

2 Related Work

The authors in, [6], devised a modified heterogenous dominant sequence clustering by ranking tasks based on their priority and balanced load distribution. Handling outliers and noisy task requests are ignored; thus, it is not feasible in a dynamic environment.

Fuzzy C-means clustering, which uses a streamlined scanning procedure, and was used by, [7]. to conduct a clustering-based load balancing. The workload associated with screening the set of accessible virtual machines is clustered according to their capabilities, and the resource demands for the issue are more than met. But quick convergence is the outcome of local search optima.

The authors in, [8], conducted a detailed survey on scheduling resources using various clustering techniques, and the author stated multiple problems faced during the clustering process load balancing, selection of appropriate architecture for scheduling, assignment of resources for the specific job request, etc.

The authors in, [9], devised a distributed inert fog-based scheduling of cloud resources with a time-based restriction model. They focused on constructing adaptability and scalability-based resource utilization architecture with noteworthy methods. However, the problem of handling noise or outliers is not considered during workload scheduling.

The authors in, [10], constructed a task scheduling algorithm using the K-Means clustering concept. The two parameters, virtual machine capacity and task length, are used for computation. The task is clustered using its size, and the resources are grouped by their processing ability. Finally, tasks are assigned with the respective cluster resource type.

The authors in, [11], developed a chicken swarm optimization-oriented evolutionary model with the mutation and crossover operator in vehicular networks. The Brownian motion-based bacteria foraging optimization is used for selecting the features of vehicles to be clustered. Depending on the resource availability, the scheduling is carried out in the cloud.

The authors in, [12], introduced a resource provisioning method using a decision-making algorithm that uses distance measures to cluster tasks. The tasks are clustered depending on their need, and the time series prediction is used for energy saving to schedule the lesson with the appropriate sources in the cloud.

This literature review analysis identifies a particular part of workload scheduling and clustering, [6], [7], [8], [9], [10], [11], [12]. Our proposed research has taken various key points from the above-proposed model research article. For improving the cloud environment-based cluster, this proposed research study was inspired by the fish school search method for improving the efficiency of the cloud environment. This fish school search method helps to reduce the risks in a cloud environment. The above literature review analysis also motivates us to ameliorate the voraciousness, lack of precision, and completeness in workload scheduling and clustering. This helps to make better extreme distribution of cloud resources. The system's performance was enhanced with the help of optimization process resources like cloud resource scheduling.

3 Robust Neutrosophic C-Means Clustering (RNCM)

Implementing the Robust Neutrosophic C-Means (RNCM) clustering algorithm involves several steps to handle uncertainties and noise in the workload data and efficiently group cloud tasks into clusters. Before applying the RNCM clustering algorithm, workload data is pre-processed to address any missing values or outliers. And then performed data normalization to ensure that all features were on a similar scale, [13].

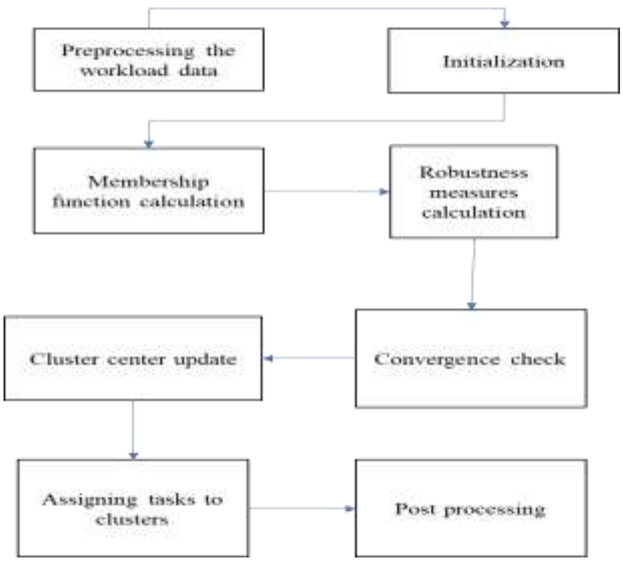

Fig. 1: Robust Neutrosophic C-Means Clustering

Figure 1 discusses the parameters required for the RNCM algorithm, including the number of clusters (c), the fuzziness parameter (m), and the robustness threshold (θ). The robustness threshold θ determines the sensitivity to outliers in the data, and it should be carefully chosen based on the dataset's characteristics. Next, Calculate the membership degree of each cloud task to each cluster using the RNCM membership function. The membership function accounts for uncertainty and partially allows a task to belong to multiple groups. We computed the robustness measures for each collection to assess the influence of noisy and uncertain data. It was updated. The cluster centers are based on the computed membership degrees and the robustness measures. By evaluating the difference between the current and previous cluster centers, convergence is checked. (If the difference is below a predefined threshold, stop the iterations; otherwise, repeat steps 3 to 5 until convergence.) After the algorithm converges, assign each task to the cluster with the highest membership degree, [14] and finally, perform post-processing, such as analyzing the results for further insights.

3.1 Cloud-based Machine Learning

Cloud-based machine learning offers optimized load balancing by leveraging its inherent scalability, flexibility, and resource management capabilities. Load balancing in cloud computing refers to the efficient distribution of incoming workload across available computing resources to ensure optimal resource utilization, minimize response time, and avoid overloading any single help. Cloud platforms provide auto-scaling capabilities that automatically adjust the number of computing resources based on the current workload demand. Machine learning models for load balancing can be designed to monitor the current workload and trigger auto-scaling actions when certain thresholds are reached. When the workload increases, additional resources are automatically provisioned to handle the load, and when the workload decreases, unnecessary resources are scaled down, optimizing resource allocation.

Figure 2 discusses the resource management needs such as business intelligence, business reporting, capacity planning, integration, resource forecasting, resource planning, and resource scheduling, access to the services is very important. In that scenario, cloud providers get access to the resource management feature by using Cloud-based machine learning systems. This access helps to allocate and reallocate tasks even the different kinds of resources on the workload scheduling. Hence the resource management features reduce the overload on the load balancing with the help of Machine learning algorithms. In addition to this, simultaneous processing is also processed for task distribution with the help of resource management features to improve cloud computing-specific actions like task distribution across multiple nodes.

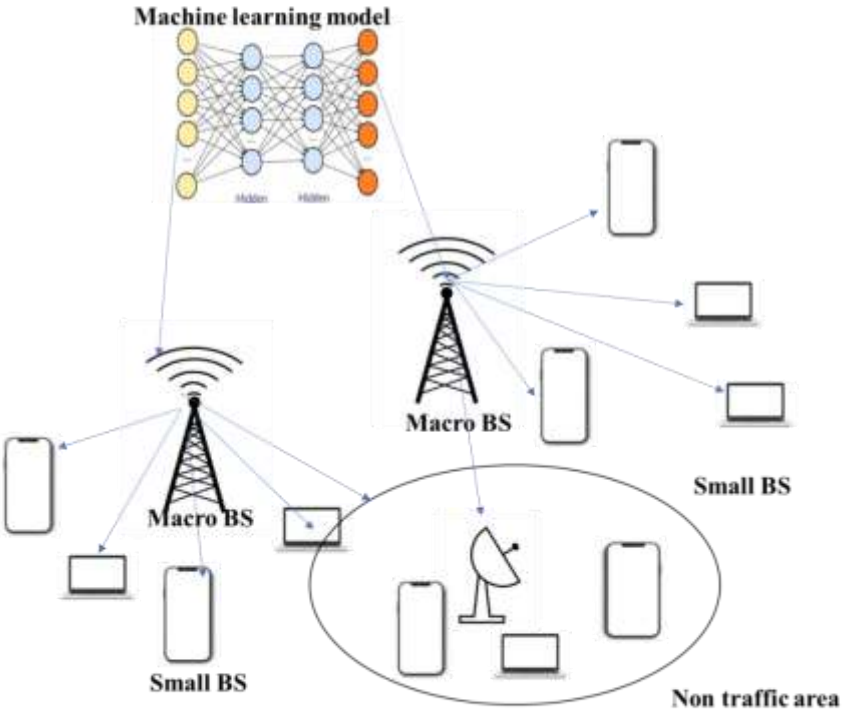

Fig. 2: Cloud-based machine learning to optimize load balancing

This proposed scenario helps to manage and maximize resource utilization by improving the load-balancing decisions, and workload distribution in an efficient manner. Simultaneously, our suggested system helps to compute the used resources and workload by using the Cloud-based machine learning models. Hence, indicated optimized resource allocation are step-up and increment as the workload distribution task by load-balancing. The real-time performance metrics and resource feedback, are formulated by machine learning. Even the load-balancing decisions are also improved by the criteria like predictive analytics. Finally, the resource management features are developed according to load balancing workload load distribution.

Cloud providers often have data centers located in different geographical regions. Load balancing algorithms can exploit this geographic distribution by directing workload to the nearest available data center with sufficient resources. This reduces latency and improves response times for end-users. Machine learning models for load balancing can consider service level agreements defined for different workloads. By considering SLAs, the algorithm can prioritize critical workloads and allocate resources accordingly, ensuring that performance targets are met. By combining cloud computing capabilities with machine learning algorithms, cloud-based machine learning can efficiently and dynamically distribute workloads to suitable resources in the cloud, achieving optimized load balancing and maximizing the utilization of cloud resources, [15].

3.2 Workload Scheduling Problem in Cloud Computing

A novel approach to tackle the workload-scheduling problem in cloud computing using the combined power of two advanced techniques, [16]. This article proposes a novel approach to tackle the workload-scheduling problem in cloud computing using the combined power of two advanced techniques:

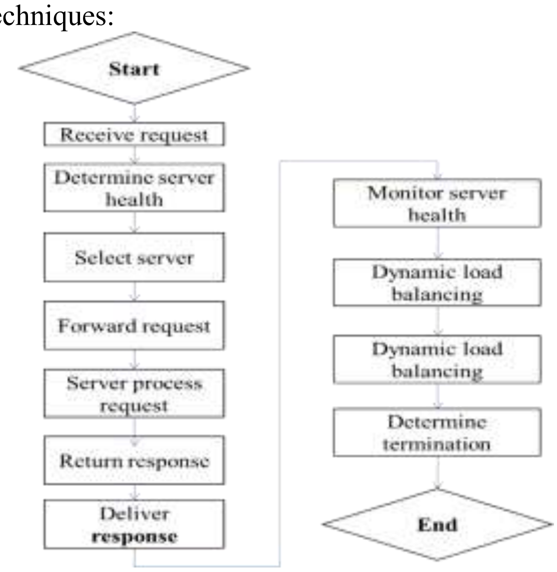

Fig. 3: Flow of load balancing in cloud computing

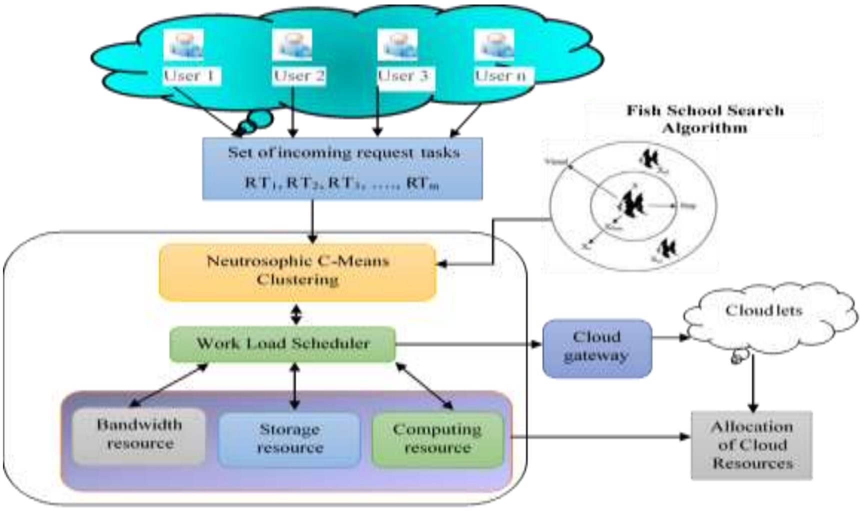

Fig. 4: Robust Neutrosophic Fuzzy C-Means Clustering fused with fish school search optimization for effective workload scheduling and allocation of resources in a cloud

Figure 3 discusses the Robust Neutrosophic C-Means (RNCM) clustering and Fish School Search (FSS) algorithm flow. By integrating these methodologies, the project aims to achieve an enhanced and optimized workload scheduling solution that can adapt to changing cloud conditions while ensuring efficient resource allocation and improved system performance. In this work, the process of workload scheduling in the cloud is accomplished by considering the type of resource required in a vague and indeterministic environment by developing a robust multivalued uncertainty theory known as Neutrosophic logic, which introduces the degree of indeterminacy to cope with the inconsistent and ambiguous information about the workload distribution and resource availability, [17]. It becomes necessary to schedule work evenly among the cloud resources to use them more effectively when there is a strong demand for work requests. Neutrosophic C-Means with Fishing school search behavior provides a resilient model with a Quality of Service at a reduced computation complexity with the best possible use of cloud resources.

Figure 4 shows the suggested robust neutrosophic C-Means enhanced with fish school searching-based clustering. Cloud computing offers a flexible and scalable infrastructure that plays a crucial role in the efficient implementation and execution of the proposed workload scheduling algorithm. The incoming task requests from cloud users are grouped according to their storage, bandwidth, and processing speed needs, [18]. Additionally, the cloud groups the virtual machines according to their configuration and availability. To exhibit the uncertainty features associated with clustering, each request and resource are characterized in the triplet components of membership degree of truthiness (T), indeterminacy (I), and falsehood (F). Assigning tasks and the most potential cloud resources as centroids requires understanding the fish school searching method. Search refinement enables the cloud resource scheduling policy by preventing chaotic cluster centroid selection and inducing the fish school's prey-seeking behavior.

3.3 The preamble of Neutrosophic Fuzzy C-Means Clustering

The Neutrosophic concept is a versatile framework that encompasses various types of logic, such as Intuitionistic fuzzy, paraconsistent, ambiguous, and multivalued logic. In Neutrosophic, each element is described in terms of three membership degrees: truthiness (T), falsity (F), and indeterminacy (I), [19], [20].

Step 1: Set the incoming task requests and resource allocation using Neutrosophic.
Step 2: Check the uncertain demands with the resource availability except for the Vague information
Step 3: Create independent membership degrees T, F, and I with the ranges among (0 to 1)
Step 4: 0 ↔ absence and 1 ↔ presence || (vague nuances, uncertain information)
Step 5: Execute the Neutrosophic C-Means Clustering into the clusters.
Step 6: Create a deterministic cluster for on-task requests

Step 7: Create an indeterministic cluster for resource requirements
Step 8: Calculate the resource needs using an objective function (OJ) of the Neutrosophic C-Means Clustering
Step 9: Calculate the membership degrees and weight factors.
The objective function is expressed as follows:

$$\mathrm{OJ}(\mathrm{NT},\mathrm{NI},\mathrm{NF},\mathrm{NC}) = \sum_{i=1}^{P}\sum_{j=1}^{NC}(\mathrm{wt}_1 NT_{ij})^{\mathcal{M}}\left\|z_i - cl_j\right\|^2 + \sum_{i=1}^{P}(\mathrm{wt}_2 NI_i)^{\mathcal{M}}\left\|z_i - \overline{cl}_{imx}\right\|^2 + \sum_{i=1}^{N}\vartheta^2(\mathrm{wt}_3\mathrm{NF}_i)^{\mathcal{M}} \tag{1}$$

NT, NI, and NF refer to the degree of membership of truthiness, indeterminacy, and falsity of the resource/task 'Z_i.' Nc denotes the number of classes, $\mathrm{wt}_{i=1,2,3}$ belongs to the weight factor, and P is the number of work requests or resources available in the cloud. cl_j refers to the cluster centroid. The control parameter ϑ is introduced to manage outlier elements within the clustering process. It plays a role in determining which data points may be outliers and need special consideration. The mean value of the first two most significant clusters is represented by $\overline{cl}_{imx}$. It is computed as the average between two cluster centroids (A_i and B_i), which are determined based on the maximum membership degree of truthiness (NT_{ij}).

$$\overline{cl}_{imx} = \frac{Cl_{Ai}+Cl_{Bi}}{2} \tag{2}$$

$$A_i = \underset{=1,2\ldots CN}{\mathrm{argmax}}(NT_{ij}); \quad B_i = \underset{j\neq f_i \cap j=1,2\ldots CN}{\mathrm{argmax}}(NT_{ij}) \tag{3}$$

Clusters are formed based on the degrees of truthiness, falsity, and indeterminacy. The classification helps in grouping tasks and resources effectively, taking into account the varying degrees of clarity in their requirements. By employing the Neutrosophic Fuzzy C-Means Clustering technique with the detailed components and equations explained above, the research aims to create more efficient and flexible resource allocation strategies in cloud computing.

Algorithm: Preamble of Neutrosophic Fuzzy C-Means Clustering
Input: Z= {z_1, z_2, ..., z_N}
Output: Return final cluster centroids cl_j, membership matrices NT, NI, NF
Initialization: no, of data points- N, no. of clusters- C, fuzzifier parameter-m, cluster centers V
Procedure:
Begin
Step 1. **for** j = 1 to C **do**:
    Randomly initialize cl_j
End for
Step 2. **for** iteration = 1 to maxIterations **do**
    for each data point, z_i **do**
        for each cluster center, cl_j **do**
            Calculate membership degree $\mathrm{OJ}(\mathrm{NT},\mathrm{NI},\mathrm{NF},\mathrm{NC})$
        End for
    End for
End for
Step 3. **For** each cluster center, cl_j **do**
    Update cluster center $\overline{cl}_{imx} = \frac{Cl_{Ai}+Cl_{Bi}}{2}$
End for
End

3.4 Fish School Search Algorithm (FFSA)

The fish school searching model (FFS) is based on the feeding behavior of fish and uses the contraction and expansion of fish during their feeding cycles, [21]. In n-dimensional space, maximizing the procedure of seeking approach is carried out depending on the agent/fish location, and the primary metric used for evaluating the search for the solution is accomplished by employing the weight variable, [22]. The primary tasks of FFS include consuming food and motion, [23].

A collection of essential responding agents, known as fish, conducts the search procedure in the FSS. The aquatic area is used as a search space for the search agents. A location within the search space, $z_i(t)$, and weight wt_i (t) represent each fish. According to how much food is found in the tank, the weight of the fish is revised iteratively.

3.4.1 Movement Operator of FFSA

To perform the movement of fish, individual, collective-volatile, and collective instinctive are three operators used. Individuals are moved randomly using the operator individual as depicted in the equation.

$$\mathrm{Z_j}(l+1) = \mathrm{Z_j}(l) + \breve{R}(l+1) \tag{4}$$

Where is the position of the j^{th} fish after and before movement concerning $\breve{R}(l+1)$, a random vector assigned during each dimension, iteration, and its value is predefined using a uniform probability distribution.

3.4.2 Feeding Operator of FFSA

After applying the movement operation, next, the feeding operator is implied by searching the neighbor based on the computation of their fitness function. If the neighbor's location is better than its present location, it moves to the new position or remains in the same situation. When the condition $ff(Z_i(l+1)) > Z_i(l))$ is satisfied, then only the new location $Z_i(l+1)$ will be agreed. Otherwise, the fish remain in the same location so that its succeeding location will not be updated $Z_i(l+1) = Z_i(l)$. The average movement of all the fish Z in the school is used for commuting the collective, instinctive part of the movement. The biased mean of shifts for discrete Z_i vector $\nu \in D^N$ It is represented as:

$$M = \frac{\sum_{i=1}^{s} \nabla z_i \nabla ff_i}{\sum_{i=1}^{N} \nabla ff_i}, \quad (5)$$

Where M signifies displacement of the fish from one location to another location, s is the size of the fish school. After the computation of M, each individual will move towards the new location as mathematically modeled.

$$Z_i(l+1) = Z_i(l) + M \quad (6)$$

The collective-violent component controls fish school predation or probing during the search process. It begins by calculating the fish school's barycenter $\wp \in R^N$ as shown in the below section concerning the fish's position Z_i of the fish and its weight wgt_i

$$\wp(l) = \frac{\sum_{i=1}^{sz} z_i(l) wgt_i(l)}{\sum_{i=1}^{N} wgt_i(l)} \quad (7)$$

While the total school weight $\sum_{i=1}^{sz} wgt_i$ If iteration l to l+1 is improved, the search agents (fishes) move toward the barycentre $\wp$.

$$wg_i(l+1) = wgt_i(l) + \frac{\nabla ff_i}{\mathrm{MX}(|\nabla ff_i|)}, \quad (8)$$

wgt_i(t) is a hyperparameter, its value lies between 1 to wgt_s. The initial value of each weight is $wgt_{sc}/2$.

Algorithm: Fish School Search Algorithm (FFSA)

Initialization: fish positions-Z_i , weights-wgt_i , step size-α, c_1 and c_2 -collective-instinctive and collective-volitive movement

Procedure:

Begin

Step 1. **for** iteration = 1 to maxIterations **do**
  for each fish, i **do**
    Generate random vector R_i
    Update position $Z_j(l+1) = Z_j(l) + \breve{R}(l+1)$
  End for
  Step 2. **For** each fish, i **do the following**:
    Compute fitness value
    if $ff(Z_i + \alpha * R_i) > ff_i$ **then:**
      $Z_i(l+1) = Z_i(l) + M$
    Else
      $Z_i(l+1) = Z_i(l)$
    End if
  End for
  Step 3. **Compute** barycenter $\wp(l) = \frac{\sum_{i=1}^{sz} z_i(l) wgt_i(l)}{\sum_{i=1}^{N} wgt_i(l)}$
  Step 4. **Update** weight $wg_i(l+1) = wgt_i(l) + \frac{\nabla ff_i}{\mathrm{MX}(|\nabla ff_i|)}$,
  Step 5. **Compute** mean displacement $M = \frac{\sum_{i=1}^{s} \nabla z_i \nabla ff_i}{\sum_{i=1}^{N} \nabla ff_i}$,
  Step 6. **Update** position $Z_i(l+1) = Z_i(l) + M$

End for

End

3.5 Robust Neutrosophic C-Means Clustering boosted with Fish School Search Algorithm

The finding was comprehensively compared to standard neutrosophic clustering, where the cluster centroids are selected randomly depending on their neighborhood. As a result, clustering begins to converge early, and because local optima are compromised, its effectiveness will be significantly reduced in ambiguous and inconsistent circumstances. Hence, in this robust Neutrosophic C-Means clustering, the centroid selection is boosted by acquiring the intelligence of the fish school search algorithm, which works in a specific problem and achieves the highest optimal solution. The centroid selection among the available resources and the workload scheduling is made by computing their fitness value, and the best resource is selected for the specific workload with the characteristic of neutrosophic values, [24].

Algorithm: Robust Neutrosophic C-Means Clustering boosted with Fish school search Algorithm for Workload Scheduling among available Cloud Resources
Input: Cloud User Incoming Task {I_{tsk}}, Resource Available {RA}
Output: A_{tsk} load distribution with optimum resource usage
Procedure:
Begin

Step 1. **For** each $I_{txk(i=1\ldots n)}$ incoming task request **do**
Pri-value = LEN(I_{tski}) * PR(I_{tski}) * DL(I_{tski}) * CST(I_{tski})
Where LEN = Length, PR = priority, DL = deadline , CST = cost
End for

Step 2. **Apply** C-Means clustering

Step 3. **Determine** potential centroid $\wp(l) = \frac{\sum_{i=1}^{sz} z_i(l) wgt_i(l)}{\sum_{i=1}^{N} wgt_i(l)}$

Step 4. **For** each RA($_{i=1\ldots m}$) **do**
Discover RA_i(parameters) //Capacity of RAM, Bandwidth, Memory, MIPS
end for

Step 5. Cluster low_resources = [], medium_resources = [], high_resources = []

Step 6. **For** resource_index in range (len (data)), **do**
resource_objective_value =ObjectiveFunction (data [resource_index])
if (resource_objective_value <= low_threshold) **then**
low_resources.append (resource_index)
elif(low_threshold < resource_objective_value <= high_threshold) **then**
medium_resources.append (resource_index)
else
high_resources.append(resource_index)
end if
end for

Step 7. **for** each $I_{tsk(i=1\ldots n)}$ **do**
Allocate the corresponding resources relative to the concern cluster model
End for
End

4 Results and Discussions

This part discusses the evaluation of the proposed model Robust Neutrosophic C-Means Clustering boosted with Fish school search algorithm for optimized cloud resource scheduling by uniformly distributing the cloud users' workload. The proposed model RNCM-FFSA is simulated using a cloudsim simulator. The task ranges from 250 to 1000. The evaluation metrics used for comparison are Makespan, degree of imbalance, resource utilization, and Execution Time.

Table 1 presents a comparison of the execution times for four different clustering algorithms, namely K-Means Clustering (KMC), Fuzzy C-Means Clustering (FCM), Neutrosophic C-Means Clustering (NCM), and Robust Neutrosophic C-Means Clustering boosted with the fish school search algorithm (RNCM-FSSA), as the number of tasks increases. In task scheduling, Makespan represents the time to execute all assignments and achieve a balanced workload distribution among available resources, [25].

Table 1. Makespan Comparison over Small Number of Tasks

Methods	**Makespan over an increasing number of tasks**			
	250	500	750	1000
KMC	105	140	275	440
FCM	95	120	230	345
NCM	75	100	190	285
Proposed RNCM-FSSA	55	78	125	148

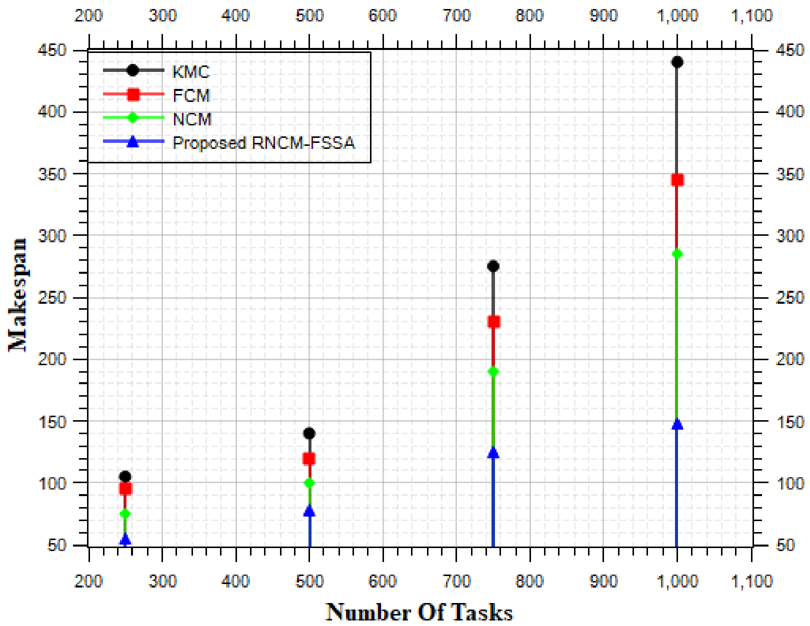

Fig. 5: Evaluation based on Makespan

Table 2. Makespan Comparison over a Large Number of Tasks

Methods	**Makespan over a large number of tasks**			
	1250	1500	1750	2000
KMC	1350	1590	1760	1960
FCM	1280	1550	1790	1930
NCM	1220	1450	1690	1860
Proposed RNCM-FSSA	1150	1380	1650	1800

Figure 5 examines the effectiveness of four clustering methods based on how long it takes to assign a workload to the available cloud resources. KMC exhibits makespan values of 105, 140, 275, and 440 units for 250, 500, 750, and 1000 tasks, respectively. FCM performs slightly better with makespan values of 95, 120, 230, and 345 units for the corresponding task numbers. NCM further improves the Makespan, achieving 75, 100, 190, and 285 units. However, the Proposed RNCM-FSSA outperforms all other methods significantly, showcasing the lowest makespan values of 55, 78, 125, and 148 units, respectively. By adopting the fish school search utilized for initial centroid selection for clustering and assigning workload based on the resource available, the RNCM-FFSA has a relatively short makespan. K-means clustering makes use of predetermined centroids that are picked at random. Additionally, centroids are chosen by FCM at random, and works are distributed according to the clusters' membership grades, in conventional NCM using, deterministic and indeterministic clustering for workload allocation. However, NCM still chooses the initial centroids at random, and when re-clustering, all three traditional methods merely employ the distance measure to assign workload to particular clusters, [26].

Table 2 presents the Makespan values for four different clustering methods as the number of tasks increases. The methods evaluated are K-Means Clustering (KMC), Fuzzy C-Means Clustering (FCM), Neutrosophic C-Means Clustering (NCM), and the proposed Robust Neutrosophic C-Means Clustering with the fish school search algorithm (RNCM-FSSA).

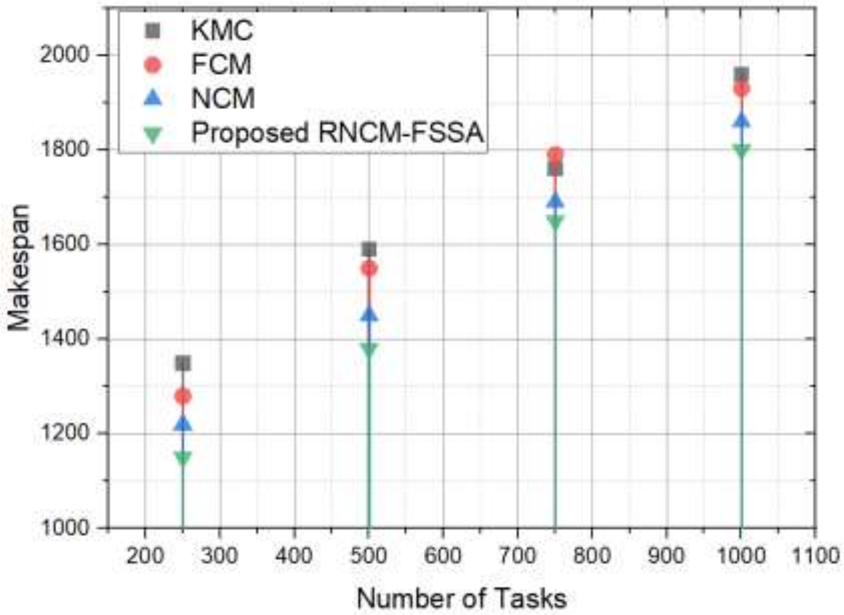

Fig. 6: Evaluation based on Makespan

Table 3. Success rate Comparison over the increasing number of Tasks

Methods	**Success rate over an increasing number of tasks**			
	250	500	750	1000
KMC	66	68	66	70
FCM	74	72	76	77
NCM	76	79	77	83
Proposed RNCM-FSSA	86	88	91	94

As the task counts grow from 1250 to 2000, the Makespan values for each method reflect the time required to complete tasks under varying workloads shown in Figure 6. KMC demonstrates Makespan values ranging from 1350 to 1960, showing an increase in task execution times with higher task counts. FCM exhibits slightly better performance with Makespan values ranging from 1280 to 1930. NCM further reduces the Makespan, achieving values from 1220 to 1860. Notably, the proposed RNCM-FSSA consistently outperforms the other methods, boasting the lowest Makespan values in all task count scenarios, ranging from 1150 to 1800. These results highlight the efficiency of RNCM-FSSA in distributing workloads and minimizing Makespan, making it a promising approach for optimizing cloud resource scheduling over a large number of tasks.

Table 3 displays the success rates of four different clustering methods as the number of tasks increases. The methods evaluated are K-Means Clustering (KMC), Fuzzy C-Means Clustering (FCM), Neutrosophic C-Means Clustering (NCM), and the proposed Robust Neutrosophic C-Means Clustering with the fish school search algorithm (RNCM-FSSA).

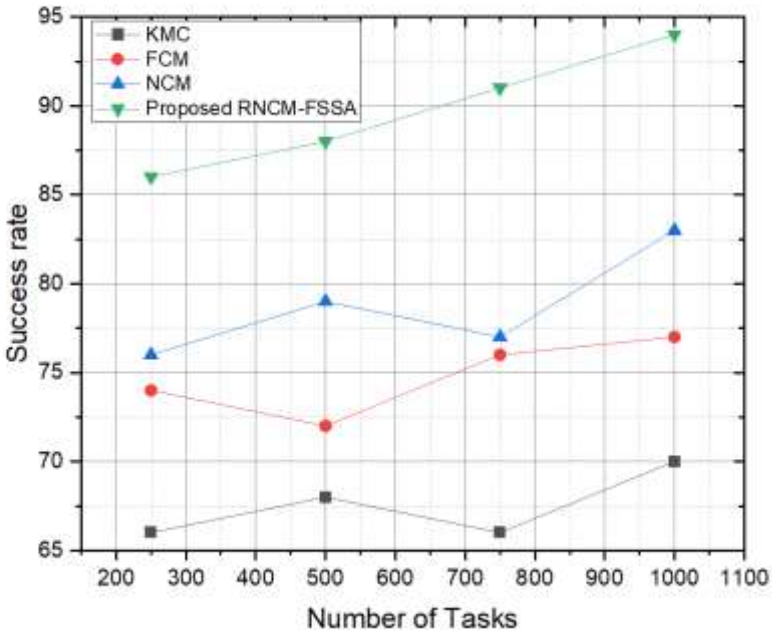

Fig. 7: Evaluation based on Success rate

Table 4. Resource Utilization Analysis of Different Methods

Methods	**Resource Utilization over an increasing number of tasks**			
	250	250	250	250
KMC	48	54	66	70
FCM	62	68	71	76
NCM	67	75	78	82
RNCM-FSSA	78	84	89	93

The success rate measures the effectiveness of these clustering methods in achieving a balanced workload distribution among available resources as the workload grows shown in Figure 7. KMC exhibits success rates that vary between 66% and 70% as the number of tasks increases from 250 to 1000. FCM's performance ranges from 72% to 77%. NCM showcases success rates from 76% to 83%. In contrast, the proposed RNCM-FSSA consistently outperforms the other methods, with success rates increasing from 86% to 94% across the task count scenarios. These results demonstrate that RNCM-FSSA excels in effectively balancing workloads, resulting in higher success rates. It offers a promising approach for optimizing cloud resource scheduling as the number of tasks grows, ensuring better utilization of available resources and timely task execution.

Table 4 presents the Makespan, which represents the completion time of tasks for four different methods. Resource utilization measures the efficiency of resource usage in a system. Cloud computing refers to how effectively the available cloud resources are utilized to execute tasks, [27]. Higher resource utilization indicates that resources are fully used, while lower utilization suggests underutilization or idle resources. The equation for Resource Utilization is as follows,

$$\text{Resource Utilization (\%)} = \left(\frac{\text{Total time resources were busy}}{\text{Total time of the experiment}}\right)*100 \tag{9}$$

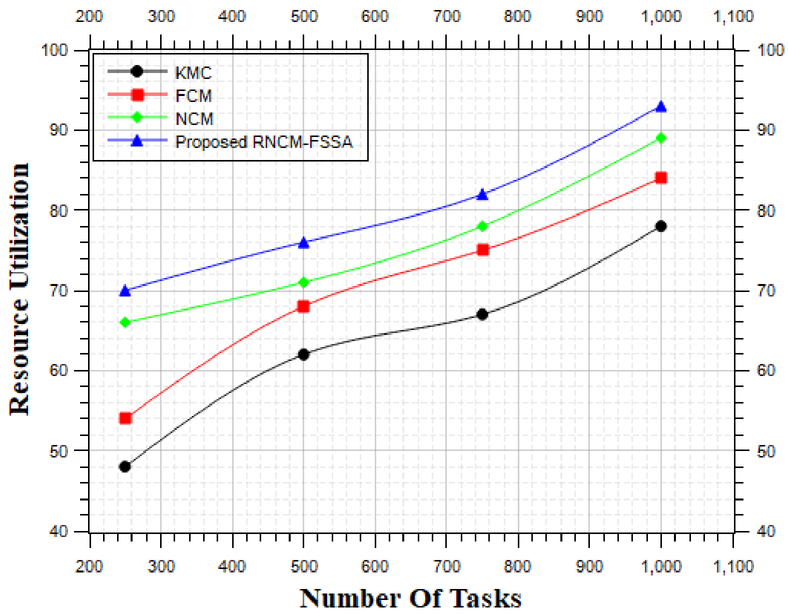

Fig. 8: Evaluation based on Resource Utilization

Table 5. Degree of Imbalance Evaluation over Increasing Number of Tasks

Methods	**Degree of Imbalance over the increasing number of tasks**			
	250	250	250	250
KMC	0.39	0.48	0.59	0.66
FCM	0.38	0.45	0.56	0.60
NCM	0.38	0.35	0.42	0.57
RNCM-FSSA	0.28	0.21	0.19	0.10

As incoming task requests are scheduled using four distinct clustering algorithms, a performance comparison concerning cloud resource utilization is shown in Figure 8. For traditional clustering models such as k-means, FCM, and NCM, the imprecision of the task requirement is more complicated, and cloud resource availability is also tough to predict. For KMC, the resource utilization starts at 48 units and gradually increases to 54%, 66%, and 70% as the number of tasks rises. FCM exhibits resource utilization values of 62%, 68%, 71%, and 76% for the corresponding task increments. NCM shows an escalating resource utilization trend with discounts of 67%, 75%, 78%, and 82%. Finally, RNCM-FSSA displays the highest resource utilization among the methods, starting at 78% and reaching 84%, 89%, and 93% as the number of tasks increases. Based on the requirements, the RNCM-FFSA groups jobs into high, low, and medium categories. When virtual machines are chosen with the assistance of the fish school search algorithm in the proposed RNCM-FFSA, incoming works are distributed evenly, and resource utilization is increased more noticeably than with the other three clustering models, [28].

Table 5 presents the Degree of Imbalance results for four methods (KMC, FCM, NCM, RNCM-FSSA) as task counts increase. It measures how evenly tasks are distributed among resources. RNCM-FSSA consistently demonstrates the lowest imbalance, showcasing its superior workload balancing. KMC exhibits the highest imbalance, indicating its inefficiency. More commentary in the text regarding Table 5 is needed to provide a comprehensive understanding of the results. The degree of imbalance measures how evenly or unevenly the workload is distributed among resources. The equation for the Degree of Imbalance is as follows,

$$\text{Degree of Imbalance} = \left(\frac{(\text{Max Load} - \text{Min Load})}{\text{Max Load}}\right) \quad (10)$$

Max Load is the maximum load among all resources, and Min Load is the minimum load among all resources.

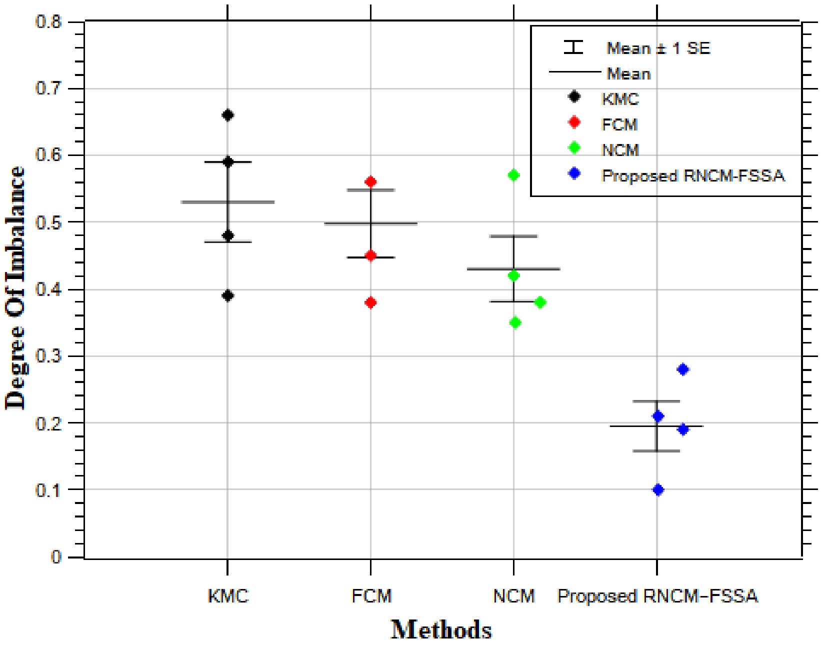

Fig. 9: Evaluation based on the Degree of Imbalance

Table 6. Execution Time Assessment over Increasing Number of Tasks

Methods	**Execution time over an increasing number of tasks**			
	250	500	750	1000
KMC	3.4	5.6	8.7	12
FCM	3.2	4.9	7.8	10.2
NCM	2.8	3.5	6.2	7.5
RNCM-FSSA	1.4	1.7	2.3	2.8

Figure 9 shows how four clustering models—K-means, FCM, NCM, and RNCM-FFSA- handle the degree of imbalance parameter to distribute load among computing resources fairly. KMC consistently shows the highest degree of imbalance, indicating its limitations in addressing class imbalance issues, with values ranging from 0.39 to 0.66 as the number of tasks increases. FCM and NCM exhibit relatively better results but still demonstrate notable imbalance, with values ranging from 0.38 to 0.60 and 0.35 to 0.57, respectively. In contrast, RNCM-FSSA stands out as the most effective method, consistently achieving the lowest degree of imbalance across all tasks, with values ranging from 0.28 to 0.10. The RNCM-FFSA intelligently handles load balancing between virtual servers in the cloud by expressing every work request regarding the degree of truthiness, falsity, and indeterminacy to combat outliers, [29] [30], and adopting the Fish school search algorithm to improve the uncertain condition in resource selection to prevent overloading and local optimum in search of essential resources. The other three conventional clustering are due to the random selection of initial centroids and searching for resource availability using local search results in early convergence.

Table 6 displays the execution time of four different methods over increasing tasks. In workload scheduling, execution time denotes the time the scheduling algorithm takes to assign tasks to resources and create an optimal schedule [31].

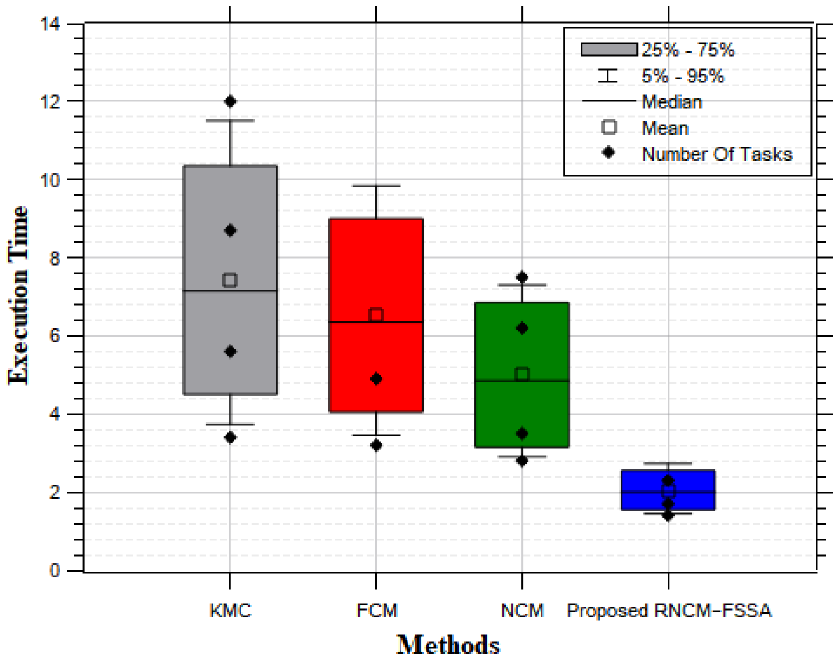

Fig. 10: Evaluation based on Execution Time

Figure 10 proves that the RNCM-FFSA takes relatively little time to accomplish the work schedule compared to the other three cluster-based workload distributions in the cloud. As the number of tasks increases, the execution times for the clustering algorithms rise accordingly. From Figure 10, We can see that, At 250 tasks, RNCM-FSSA exhibits the shortest execution time (1.4s), followed by NCM (2.8s), FCM (3.2s), and KMC (3.4s). However, as the task count reaches 1000, RNCM-FSSA remains the fastest (2.8s), with NCM (7.5s), FCM (10.2s), and KMC (12s) showing longer execution times. Overall, RNCM-FSSA consistently outperforms the other algorithms, offering the most efficient clustering solution as the number of tasks increases. This is because the algorithm strategically selects initial cluster centers using a measure of uncertainty, called membership degree of indeterminacy. This means it's better at handling tasks and resources with unclear or changing requirements in the dynamic cloud environment. The fish school search algorithm, a part of RNCM-FFSA, significantly enhances the clustering process, making it quicker and more efficient compared to other algorithms like KMC, FCM, and NCM.

5 Conclusion

To handle the issue of heterogenous environment-based resource scheduling and load balancing, the clustering technique can able to meet the demand for cloud resources and reduce the screening process-related overhead by forming clusters of both the incoming job and virtual machines based on their capacity in this paper a robust NCM-FSSA algorithm is developed. The indeterminacy of the incoming task request is very challenging for conventional clustering. Hence in this work, the generalization of the uncertainty theories known as neutrosophic logic is used for clustering the outliers in the resource scheduling scheme. The second factor, in FCM, KCM, and NCM, the centroids are selected arbitrarily, and the clustering process begins, which results in early convergence in scheduling, and their performance is directly degraded. To overcome it, this work adopts the metaheuristic model of fish schooling searching, and its searching behavior is utilized for centroid selection. The assessment of the proposed model RNCM-FFSA provides the highest resource utilization with the slightest degree of imbalance and execution time in the workload scheduling of cloud resources.

References:

[1] Gawali, M.B., Shinde, S.K. Task scheduling and resource allocation in cloud computing using a heuristic approach. *J Cloud Comp* 7, 4 (2018).

[2] Mohammad S Aslanpour, Adel N Toosi, Raj Gaire, and Muhammad Aamir Cheema., Auto-scaling of Web Applications in Clouds:

A Tail Latency Evaluation. In 2020 *IEEE/ACM 13th International Conference on Utility and Cloud Computing*, pp.186-195

[3] Alhaidari F, Balharith TZ, Enhanced Round-Robin Algorithm in the Cloud Computing Environment for Optimal Task Scheduling. *Computers*. 2021; 10(5):63.

[4] Zheng, T., Wan, J, Zhang, J, Deep Reinforcement Learning-Based Workload Scheduling for Edge Computing. *J Cloud Comp* 11, 3 (2022).

[5] Bezdan T., Zivkovic, M., Bacanin, N., Strumberger, I., Tuba, E., Tuba, M, Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm, *Journal of Intelligent and Fuzzy Systems*, Volume 42, Issue 1, 2022, pp.411-423

[6] Amer Al Rahayfeh, Saleh Atiewi, Abdullah Abuhussein, MuderAlmiani, Novel Approach to Task Scheduling and Load Balancing Using the Dominant Sequence Clustering and Mean Shift Clustering Algorithm, *Future Internet*, vol. 11, no. 109, pp.1-15, 2019.

[7] Geetha Megharaj, Dr. Mohan G. Kabadi, Rajani, Deepa M, "FCM-LB: Fuzzy C Means Cluster Based Load Balancing in Cloud", *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 7, Special Issue 6, 2018.

[8] Khallouli, W., Huang J, Cluster resource scheduling in cloud computing: literature review and research challenges. *J Supercomput* 78, pp.6898–6943 (2022).

[9] Hajvali, M., Adabi, S., Rezaee, A, Decentralized and scalable hybrid scheduling-clustering method for real-time applications in volatile and dynamic Fog-Cloud Environments. *J Cloud Comp* 12, 66 (2023).

[10] Allimuthu, U. "BAU FAM: biometric-blacklisting anonymous users using fictitious and adroit manager." *J. Adv. Res. Dyn. Control Syst.*, Volume 9 (2021) No. 12-Special, pp.722-738

[11] R. Ruhin Kouser, T. Manikandan, A novel clustering and optimal resource scheduling in vehicular cloud networks using MKMA and the CM-CSO algorithm, *International Journal of Communication Systems*, Vol. 36, Issue 5, 2023

[12] B. Fei, Elastic Resource Provisioning Using Data Clustering in Cloud Service Platform, *Services Computing*, vol. 15, no. 03, pp.1578-1591, 2022.

[13] N. Akhtar and M. V. Ahmad, "A Modified Fuzzy C Means Clustering Using Neutrosophic Logic," 2015 *Fifth International Conference on Communication Systems and Network Technologies*, Gwalior, India, 2015, pp.1124-1128. doi: 10.1109/CSNT.2015.164

[14] H. A. Nugroho, M. Rahmawaty, Y. Triyani and I. Ardiyanto, "Neutrosophic and fuzzy C-means clustering for breast ultrasound image segmentation," 2017 *9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Phuket, Thailand, 2017, pp.1-5. doi: 10.1109/ICITEED.2017.8250453

[15] E. Rashno, A. Rashno and S. Fadaei, "Fluid Segmentation in Neutrosophic Domain," 2019 *5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, Shahrood, Iran, 2019, pp.1-5.

[16] Gaber, Tarek, Gehad Ismail, Ahmed Anter, Mona Soliman, Mona Ali, Noura Semary, Aboul Ella Hassanien, and Vaclav Snasel. "Thermogram breast cancer prediction approach based on neutrosophic sets and fuzzy c-means algorithm." In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp.4254-4257. Ieee, 2015. doi: 10.1109/EMBC.2015.7319334

[17] X. Lei, X. Yang and F. -X. Wu, "Artificial Fish Swarm Optimization Based Method to Identify Essential Proteins," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 17, no. 2, pp.495-505, 1 March-April 2020. doi: 10.1109/TCBB.2018.2865567

[18] M. Abdel-Basset, G. Manogaran, A. Gamal, F. Smarandache, A group decision making framework based on neutrosophic TOPSIS approach for smart medical device selection. *Journal of medical systems*, vol.43(2) (2019), pp.38-54.

[19] Saha, Abhijit, Smarandache, Florentin, Said, Broumi, Neutrosophic Soft Sets Applied on Incomplete Data, *Neutrosophic Sets and Systems*, Vol.32, pp.282-293, 2020.

[20] Bova V, Kuliev, E, Rodzin S, Prediction in Intellectual Assistant Systems Based on Fish School Search Algorithm. *Izv. Sfedu Eng*. Sci, 2, pp.34-47, (2019).

[21] Alhaqbani, Amjaad Kurdi, Heba Hosny, Manar. Fish-Inspired Heuristics: A Survey of the State-of-the-Art Methods. Archives of

Computational Methods in Engineering. 29, (2022).

[22] M. Habibi, A. Ayatollahi, N. Dallalazar and A. Kermani, "Lumen boundary detection using neutrosophic c-means in IVOCT images," 2019 *5th Conference on Knowledge Based Engineering and Innovation (KBEI)*, Tehran, Iran, 2019, pp.599-604. doi: 10.1109/KBEI.2019.8734946

[23] X. Wang and J. Zhao, "A Novel Neutrosophic Image Segmentation Based on Improved Particle Swarm Optimization Fuzzy C-Means Algorithm," 2019 *18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, Wuhan, China, 2019, pp.80-83. doi: 10.1109/DCABES48411.2019.00027

[24] S. Ashour, C. Du, Y. Guo, A. R. Hawas, Y. Lai and F. Smarandache, "A Novel Neutrosophic Subsets Definition for Dermoscopic Image Segmentation," in *IEEE Access*, vol. 7, pp. 151047-151053, 2019. doi: 10.1109/ACCESS.2019.2946762

[25] J. M. Mathew, Surya S.R and P. Simon, "Automatic cloud detection based on neutrosophic set in satellite images," 2013 *International Conference on Control Communication and Computing (ICCC)*, Thiruvananthapuram, India, 2013, pp.210-215. doi: 10.1109/ICCC.2013.6731652

[26] R. Sivanandan and J. Jayakumari, "An Improved Ultrasound Tumor Segmentation Using CNN Activation Map Clustering and Active Contours," 2020 IEEE *5th International Conference on Computing Communication and Automation (ICCCA)*, Greater Noida, India, 2020, pp.263-268. doi: 10.1109/ICCCA49541.2020.9250909

[27] W. F. Halawa, S. M. Darwish and A. A. Elzoghabi, "Cotton Warehousing Improvement for Bale Management System Based on Neutrosophic Classifier," in *IEEE Access*, vol. 9, pp. 159413-159420, 2021. doi: 10.1109/ACCESS.2021.3126790

[28] Y. Zhu and H. Gao, "Improved Binary Artificial Fish Swarm Algorithm and Fast Constraint Processing for Large Scale Unit Commitment," in *IEEE Access*, vol. 8, pp.152081-152092, 2020. doi: 10.1109/ACCESS.2020.3015585

[29] C. Wu, X. Fu, J. Pei and Z. Dong, "A Novel Sparrow Search Algorithm for the Traveling Salesman Problem," in *IEEE Access*, vol. 9, pp.153456-153471, 2021. doi: 10.1109/ACCESS.2021.3128433

[30] C. J. A. Bastos-Filho, W. A. S. Silva and L. R. M. Lira, "Comparing Meta-heuristics for AdaBoost Training Applied to Platelets Detection," in *IEEE Latin America Transactions*, vol. 12, no. 5, pp.942-950, Aug. 2014. doi: 10.1109/TLA.2014.6872910

[31] S. Tuo, H. Chen and H. Liu, "A Survey on Swarm Intelligence Search Methods Dedicated to Detection of High-Order SNP Interactions," in *IEEE Access*, vol. 7, pp. 162229-162244, 2019. doi: 10.1109/ACCESS.2019.2951700.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

S. Yuvaraj Gandhi, carried out the article writing and executing the Optimizing Workload Scheduling in Cloud Paradigm. T. Revathi has implemented and took survey about Workload Scheduling.

S. Yuvaraj Gandhi and T. Revathi has organized and executed the comptre expriments of Section 4.

Hence the The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflicts of interest to declare.