


## Article

# Dynamic Cloud Resource Allocation: A Broker-Based Multi-Criteria Approach for Optimal Task Assignment

Abdulmajeed Aljuhani <sup>1,\*</sup> and Abdulaziz Alhubaishy <sup>2</sup> <sup>1</sup> College of Computer Science and Engineering, Taibah University, Medina 41411, Saudi Arabia<sup>2</sup> College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia; a.alhubaishy@seu.edu.sa

\* Correspondence: abaljuhani@taibahu.edu.sa

**Abstract:** Cloud brokers and service providers are concerned with utilizing available resources to maximize their profits. On the other hand, customers seek the best service provider/resource to provide them with maximum satisfaction. One of the main concerns is the variability of available service providers on the cloud, their capabilities, and the availability of their resources. Furthermore, various criteria influence the effective assignment of a task to a virtual machine (VM) before it is, in turn, submitted to the physical machine (PM). To bring cloud service providers (CSPs) and customers together, this study proposes a broker-based mechanism that measures the tendency of each customer's task. Then, the proposed mechanism assigns all tasks—in prioritized order of importance—to the best available service provider/resource. The model acquires the importance of each task, CSP, or resource by extracting and manipulating the evaluations provided by decision makers and by adopting a multi-criteria decision-making (MCDM) method. Thus, a partial result of the proposed mechanism is a defined and prioritized pool for each of the tasks, CSPs, and resources. Various MCDM methods are examined and compared to validate the proposed model, and experiments show the applicability of the various methods within the model. Furthermore, the results of the experiments verify the suitability and applicability of the proposed model within the cloud environment.

**Keywords:** task scheduling; multi-criteria decision making; task tendency; cloud computing; cloud broker



**Citation:** Aljuhani, A.; Alhubaishy, A. Dynamic Cloud Resource Allocation: A Broker-Based Multi-Criteria Approach for Optimal Task Assignment. *Appl. Sci.* **2024**, *14*, 302. <https://doi.org/10.3390/app14010302>

Academic Editor: Andrea Prati

Received: 14 November 2023

Revised: 17 December 2023

Accepted: 21 December 2023

Published: 29 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The cloud computing environment allows consumers to access and utilize various resources without owning the infrastructure. The cloud environment consists of various features, such as service types, cloud protocols, and activities, as well as cloud deployment models that include public, private, community, and hybrid models. In addition, it contains three service models: software as a service (SaaS), infrastructure as a service (IaaS), and platform as a service (PaaS). Cloud consumers are given access to the resources in accordance with the service level agreement (SLA). Cloud technology enables consumers to gain several benefits, such as minimizing operating costs, improving flexibility, and increasing mobility. However, to ensure a high level of consumer satisfaction, both consumers and cloud service providers (CSPs) should consider the quality of service (QoS). Consumers pay for cloud services, whereas CSPs provide on-demand services. In general, the service contract (or SLA) specifies the cloud service's cost based on the quantity and quality of service. According to SLA standards, identifying an effective solution for handling consumers' requests is required even as the number of cloud consumers grows [1]. Service providers such as Amazon and Google have introduced a variety of cloud services in response to the rapid development of cloud computing, enabling users to manage enormous datasets operating on remote servers without having to install and execute on local devices [2].

Management of the resources required to fulfill consumer needs falls under the purview of the service provider. CSPs schedule consumer tasks and efficiently allocate computing resources by using a variety of scheduling techniques. Various kinds of pre-configured virtual machines (VMs) are used to provide these computing resources. The VM instances can be selected by consumers depending on their application needs and concerns related to cost. In a cloud environment, task-scheduling algorithms aim to support the scheduling of tasks to suitable cloud resources so that optimal performance can be attained. The execution of tasks in a particular order and how resources are allocated to tasks affect the performance of cloud systems. Through effective task scheduling and resource management, service providers seek to boost revenues and fully utilize resources.

Scheduling techniques, such as particle swarm optimization (PSO) [3], the genetic algorithm (GA) [4], and game theory [5], are commonly used to schedule consumers' tasks in a cloud environment. These tasks are managed while considering various criteria, such as the load and size of the task, the execution time, and the availability of resources [6]. To effectively handle the resources, a resource management system that can vigorously schedule and allocate resources is required [7]. For a specific task that will boost performance and must be finished on time, cloud consumers search for the most resources possible. Cloud services provided by different service providers differ in terms of their performance. Due to the various data centers and technologies applied, services provided by the exact service provider can also vary in terms of performance. Globally, data center capacity in different forms is quickly rising. Inadequate load balancing against available resource allocation is the primary cause of rising server interruption rates. The most commonly used criteria for evaluating cloud service performance are the QoS attributes. To address the issue of measuring cloud-based services, the Cloud Services Measurement Initiative Consortium (CSMIC) has suggested a hierarchical Standard Measurement Index (SMI) [8]. Usability, accountability, assurance, agility, security and privacy, financial aspects, and performance are the seven categories that the framework uses to measure cloud-based services. In addition, three or more attributes are included in each category, resulting in a total of 51 sub-attributes. For instance, criteria such as accessibility, suitability, operability, learnability, client personnel requirements, installability, transparency, and understandability are used to further enhance usability.

The consumers of today encounter difficulties when trying to select the most appropriate service provider with respect to various considerations, such as available budgets, operational needs, and performance duration, due to the rapid rise of online cloud services and providers. The most suitable service provider should be determined by a consumer's evaluation of various QoS attributes, such as dependability, security, cost, and performance. The consumer's requirements must be matched with the features of the cloud services provided by different service providers to determine which CSP is most suitable. A thorough investigation and a reliable ranking methodology are needed to select an appropriate CSP. Since this incorporates the inherent linkages among several QoS criteria, this can be classified as a multi-criteria decision-making (MCDM) problem. Recent years have seen a tremendous increase in investigations of CSP selection while relying on the MCDM method. For example, Al-Faifi et al. [9] applied K-means algorithms with a DEMATEL–analytic network process (ANP) to rank CSPs with respect to the interdependence among the performance measurements. Furthermore, Hussain et al. [10] applied a hybrid method by integrating the analytic hierarchy process (AHP) with the Induced OWA (IOWA) operator to assist cloud consumers with selecting the optimal service provider. The Conditional Preference Networks (CP-nets) framework was proposed by Alashaikh and Alanazi [11] to select the optimal CSP while considering a group of attributes with complex interdependencies.

This study explains how to evaluate all parties (CSP, resources, and tasks) by using the MCDM method. Then, the tendencies of tasks based on upper and lower thresholds are defined, and these are calculated based on the results of applying the MCDM method.

Finally, a simple mechanism for optimizing the process of assigning customers' tasks to the best available CSP/resource is adopted.

## 2. Related Work

In a cloud environment, different stakeholders have different interests, which presents a major challenge [12]. Adopting techniques such as the MCDM method can help decision makers overcome these challenges. Additionally, the effectiveness and efficiency of job offloading can be enhanced by adopting MCDM methods, such as AHP [13,14]. Krishankumar et al. [15] stated that the Technique for Order of Preference by Similarity to the Ideal Solution (TOPSIS) and the AHP are the most commonly adopted MCDM techniques in cloud environments. The problem of cloud task scheduling was investigated by Rafieyan et al. [16] based on the integration of an MCDM method—namely, the Best Worst Method (BWM)—and the Vlekriterijumsko KOMPromisno Rangiranje (VIKOR) (or Multi-criteria Optimization and Compromise Solution) method. Task priorities were specified by implementing the VIKOR method and by considering various levels of priorities, such as the resource, scheduling, and task levels. The authors argued that the integration of the BWM and VIKOR enhanced factors such as the waiting time, throughput, and the utilization of VMs. Lai et al. [17] integrated the double-normalization-based multiple aggregation (DNMA) approach into the Z-number environment to overcome limitations in the existing MCDM method for CSP selection. The presented method considered various qualitative and quantitative factors, such as cost and target types, for sustainable service provider development. Lai et al. [18] ranked different CSPs based on a combined compromise solution (CoCoSo) method. The authors introduced a maximum variance optimization framework to ensure consistent results. Neeraj et al. [19] compared different MCDM techniques, such as TOPSIS, the AHP, VIKOR, and the Preference Ranking Organization Method for Enrichment of Evaluations (PROMETHEE) II, to find the most efficient method in terms of time consumption and robustness in a cloud environment. The results showed that the AHP and TOPSIS were equivalent in terms of robustness. Kumar et al. [20] proposed a workflow scheduling algorithm for a cloud environment based on the integration of TOPSIS with the entropy weight method (EWM). The proposed method considered various significant criteria, such as reliability, time, cost, and energy consumption. The main objective of their study was to reduce costs, energy consumption, and execution time. Based on the VIKOR approach, Radi et al. [21] provided a service broker policy for choosing the optimal data center in cloud computing. Through consideration of variables including cost, response time, and data center processing time, the policy sought to maximize user priorities.

A multi-criteria-based resource score heuristic for cloud workflow scheduling was introduced by Chitra [22]. The heuristic was created with the aim of lowering the makespan while taking the probability of the physical resources' temporal availability for the VMs into consideration. Following the creation of an initial schedule based on a priority list of tasks, the task-to-VM mapping was refined in real time to adapt to dynamic changes in the availability of cloud resources. A comparative investigation of MCDM techniques in the backfilling process for executing tasks with deadlines in a cloud environment was proposed by Nayak et al. [23]. In the backfilling algorithm, proper resources were supplied by the backfill tasks. When there were similar tasks, some of the tasks were classified as backfill tasks, resulting in a difficult problem. The backfilling algorithm was simulated alongside the AHP, VIKOR, and TOPSIS to prevent scheduling conflicts among identical tasks. Meesariganda and Ishizaka [24] presented a mapping/scaling technique based on the AHP to solve strategic decision problems in a cloud environment. The proposed model considered six criteria, including risk, financial benefits, core competency, and customer satisfaction. These criteria were weighted to evaluate three alternatives, which were integrated, hardware, and no entry. Achieving a QoS that met customers' requirements within a certain cost range by using the GA was addressed by Devarasetty and Reddy [4]. SELCLOUD is a framework proposed by Jatoth et al. [25] to help with selecting the best CSP based on the integration of the AHP with Grey TOPSIS. An uncertainty-based approach for edge

computing with balanced resource provision and cost regarding energy utilization was developed by Xu et al. [26]. In order to accomplish cost- and energy-effective dynamic resource allocation, multi-objective dynamic allocation with balanced scheduling methods were adopted. Youssef [27] presented a model for selecting the best CSP based on the integration of the BWM with TOPSIS. Various significant criteria, such as response time, security, maintainability, reliability, sustainability, usability, cost, and scalability, were evaluated by using BWM–TOPSIS. Kumar et al. [28] introduced a model for selecting a suitable CSP based on the AHP integrated with TOPSIS. The authors used an AHP importance scale ranging from one to nine to determine the weight of each service provider while considering a number of attributes, such as data privacy, accuracy, assurance of data integrity, and reliability. Sun et al. [29] evaluated the interdependence of criteria with regard to cloud service selection with criterion interactions. The interdependencies of the criteria were evaluated based on a fuzzy measure and the Choquet integral, and the importance of the criteria was determined by using pairwise comparisons. A Fuzzy–Euclidean–Taxicab-distance-based method was proposed by Garg [30] to select the suitable cloud deployment model. The introduced model consisted of four cloud deployment models as alternatives, as well as three decision parameters that included 17 sub-parameters.

Nawaz et al. [31] proposed a cloud broker architecture for selecting the best service provider while considering changes in cloud consumers' preferences. The authors adopted the BWM to rank the available services, and the findings showed that better performance could be achieved when using the BWM rather than other MCDM techniques, such as the AHP. Alhubaishy and Aljuhani [32] proposed a framework for prioritizing cloud tasks while considering consumers' preferences by using the BWM. The proposed framework allowed consumers to evaluate a broad range of criteria with more flexibility in terms of adding or altering criteria to meet their preferences. Nazeri and Khorsand [33] studied the management of energy consumption in a cloud environment when executing applications. The authors integrated the fuzzy AHP with TOPSIS to prioritize various cloud solutions with respect to consumers' requirements. Malhotra et al. [34] introduced the integer multiplication (IM) method based on the AHP as a simplified and effective prioritization technique. IM assisted cloud consumers with making quick decisions with respect to their preferences when selecting a suitable CSP. In addition, the IM method was used to evaluate important decision-making criteria, such as performance, safety, memory, and cost, as well as several sub-criteria, such as availability, reliability, throughput, power consumption, response time, and bandwidth. The authors compared the IM method's results with those of the existing geometric mean (GM) method to prove the power of the IM method over the GM method. Saha et al. [35] used the ANP integrated with VIKOR to determine the best CSP. The ANP is a general version of the AHP, and it allows decision makers to model decision elements in different clusters. The authors used the ANP to rank local service providers and used VIKOR for the global ranking. The ANP network contained criteria such as accountability, agility, flexibility, adaptability, security, performance, and serviceability. These criteria were evaluated with respect to different goals, such as benefits and opportunities, to rank three service providers and select the most suitable one.

Table 1 presents a comparison of several algorithms based on task rankings and when they should be used, including the GA, Ant Colony Optimization (ACO) algorithms, First Come First Service (FCFS), Short Job First Scheduling (SJF), and the BWM.

**Table 1.** Comparison of various algorithms based on how tasks are ranked and when the algorithms should be adopted [36].

	Priority of Tasks	When the Algorithm Is the Best Choice
GA	The advantage of the chromosomal form is that it preserves the sequence in which tasks are to be completed on the designated nodes.	This is a feasible alternative in a large solution search space, but it requires more processing time than other methods.

Table 1. Cont.

	Priority of Tasks	When the Algorithm Is the Best Choice
ACO	The virtual machine is chosen for the subsequent task based on the ant iteration after tasks are arranged in either ascending or descending order. After each iteration, the number of solutions is equal to the number of ants.	For task scheduling, it is the optimum option in soft real-time systems.
SJF	Tasks are prioritized by starting with the least and working up to the greatest based on their durations.	This is suitable in situations in which shorter tasks are more crucial than longer ones.
FCFS	According to the time of their arrival, the tasks are listed in the task list.	This method works well when the solution search space is small.
BWM	The proportional relevance of the tasks determines their order of priority.	This is a suitable option when the focus is on the weight of the tasks.

### 3. Framework

The proposed mechanism seeks to effectively prioritize tasks and efficiently assign them to a VM before the actual allocation of the VM to a PM. Therefore, the proposed mechanism is considered a task scheduling mechanism [37]. Figure 1 shows the proposed multi-criteria task scheduling mechanism. The first component of the framework consists of three sets of inputs to initiate the evaluation and assignment of tasks to the best-matched resources. The first set contains all available service providers (SPs) in the cloud, along with the set of important criteria used to evaluate the SPs. The second set contains all available tasks that the model needs to efficiently assign to the best-matched resources along with the criteria that the customer will use to evaluate the tasks. The last set of inputs contains all available resources in the cloud along with the criteria for evaluating these resources.

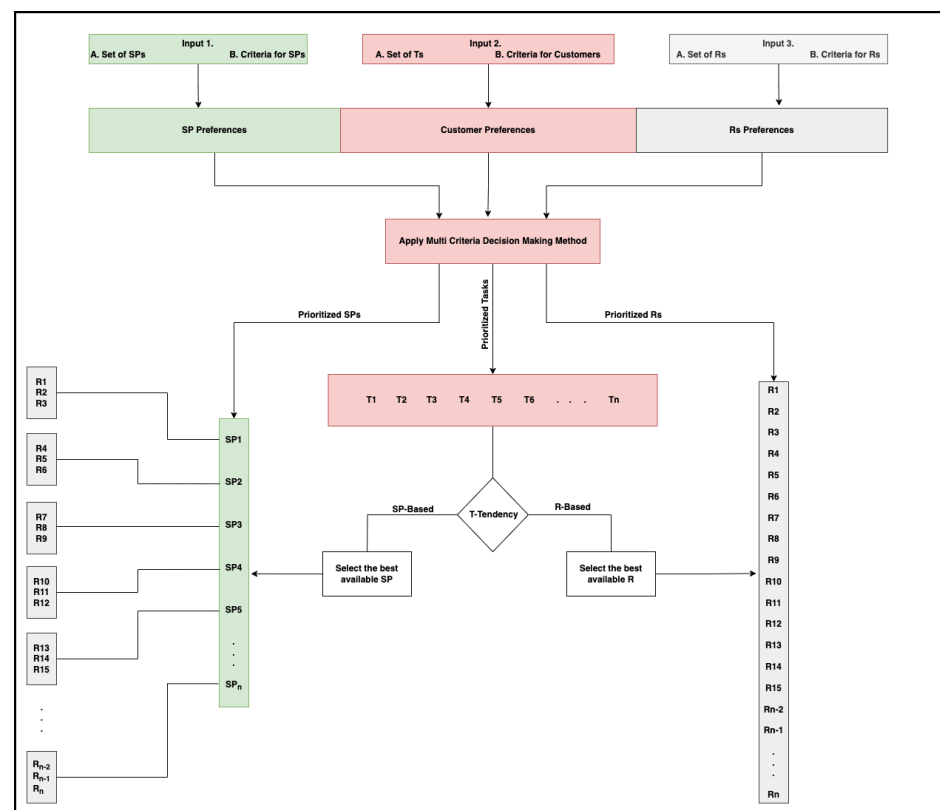


Figure 1. Framework for multi-criteria task scheduling.



The second component of the framework acquires the preferences of the SPs and customers to initiate the prioritization process. These preferences are fed to the MCDM method in the next component of the framework. “Preference” refers what is more important to the SP (or customer) in order to help the framework find the best assignment. For example, one SP is more concerned with reputation than other criteria, while another SP prefers executing tasks with a high price, and so on. Similarly, one customer might prefer the execution of tasks with a low price over those requiring a short time, while another customer is more concerned with the availability of the service than the price, and so on. In the third component of the framework, the actual prioritization process takes place to provide the prioritized list of SPs, customers, and resources.

As mentioned in the literature review, various MCDM methods, such as the AHP ([37,38]), the ANP [35], and the BWM [16], have proven their ability in cloud task scheduling. Each MCDM method has its advantages and disadvantages. Regardless of the underlying MCDM, which can certainly influence the effectiveness of the framework, we consider the selection of a suitable MCDM method that occurs during the implementation phase of the proposed framework. When implementing the framework, it is important to select the MCDM method during the early stages. However, differences regarding the type of information needed from the decision maker, SP, or customer should be considered when selecting the MCDM method. For example, the AHP and ANP require the decision maker to give pairwise comparisons between each pair of criteria, while the BWM requires the decision maker to select the best criterion and the worst criterion and give pairwise comparisons between only the best and worst criteria and all other criteria. Applying an MCDM method results in prioritized sets of SPs, tasks, and resources. The main reason to have separately prioritized SPs and prioritized resources is to enable task execution based on their tendency. “Task tendency” means that if the customer’s preference shows more of a tendency toward the execution of his/her tasks, the tasks will be assigned to the best available resources among the prioritized resources, regardless of the SPs that the resources belong to. On the other hand, if the customer is more concerned with criteria related to the overall evaluation of SPs, his/her tasks will be assigned to the resources of the best available SPs. To illustrate the task tendency, suppose that the framework has three criteria for evaluating tasks: completion time, availability, and reputation. Suppose that the customer’s evaluation shows that reputation is the most important criterion (80%), followed by availability (15%) and completion time (5%). In this case, we say that the task tends to the SPs more than the resources; therefore, we assign it to the best available SPs. Suppose that the results show the completion time (70%) to be the most important, followed by reputation (20%) and availability (10%). In this case, we say that the task tends to the resources more than the SPs; therefore, we assign it to the best available resource. We propose that the threshold for measuring the tendency of the tasks be based on (1) the number of SP-based/resource-based criteria and (2) the overall weight of the criteria resulting from the MCDM method. Finally, it is important to have a suitable mechanism for managing the load among the available resources; many mechanisms have been provided, as mentioned in the literature. However, the load fairness assignment is beyond the scope of this study; therefore, we examine the proposed method without considering this issue. Algorithms 1 and 2 show the steps of prioritizing and scheduling tasks.

**Algorithm 1:** Prioritization of Tasks, Service Providers, and Resources**Input:** 1. Sets of SPs, tasks, and resources.

2. Sets of criteria for SPs, tasks, and resources.

**Output:** Prioritized SPs, tasks, and resources

1. Select the MCDM method.
2. Each SP evaluates its overall capability, and preferences are acquired.
3. Each SP evaluates tasks, and preferences are acquired.
4. Each SP evaluates the capability/performance of its resources, and preferences are acquired.
5. Apply the selected MCDM method to each evaluation.
6. Calculate the overall weight of each set of criteria.
7. Prioritize SPs.
8. Prioritize tasks.
9. Prioritize resources.

**Algorithm 2:** Scheduling of Tasks Based on Their Tendencies**Input:** Prioritized SPs, tasks, and resources.**Output:** Assignment of tasks to the best available resources.

1. Calculate the tendency of each task based on a certain threshold.
2. Based on the task tendency, store each task in either the “Ts-tendency-to-SPs” list or “Ts-tendency-to-resources” list.
3. Calculate the overall ranking of tasks for the “Ts-tendency-to-SPs” list.
4. Calculate the overall ranking of tasks for the “Ts-tendency-to-resources” list.
5. Assign each task in the “Ts-tendency-to-SPs” list to a resource in the best available SPs.
6. Assign each task in the “Ts-tendency-to-resources” list to the resource in the best available resources regardless of its SPs.

**4. Experimental Results and Discussion**

In the beginning, the criteria for evaluating SPs, cloud resources, and tasks should be determined. As stated in the introduction, the hierarchical SMI framework proposed by the CSMIC includes seven categories, and each category consists of several sub-attributes. Based on the SMI framework, criteria such as provider business stability, provider certifications, provider personnel requirements, profit or cost sharing, reputation, and trust can be used to evaluate various CSPs. In addition, criteria such as adaptability, portability, recoverability, reliability, performance, and cost can be used to evaluate cloud resources. However, in this study, four criteria were selected to measure the tendency of each task. These criteria were reputation, trust, performance, and cost.

In experiment 1 and 2, the AHP was applied to rank a set of SPs, resources, and tasks. The AHP method was introduced by Saaty [39] as a theory of measurement to solve complex decision problems. A pairwise comparison matrix was used to derive the ratio scale in the AHP approach. The structure of the AHP consists of three levels, which are the goal, criterion, and alternative levels. At the goal level, the objective of applying the AHP or the decision problem that needs to be solved is identified. The criterion level contains all possible criteria related to the decision problem, and this affects the alternatives. In addition, the criterion level may include sub-criteria if they are needed during the structuring of the decision problem. The alternative level includes all possible solutions that are evaluated with respect to the selected criteria to find the optimal solution. However, in order to maintain consistency of judgment, the number of alternatives should be reasonably small; therefore, the AHP was adopted in Experiments 1 and 2 to consider a suitable number of

alternatives. After structuring the decision problem, consumers provide their preferences for each AHP level and assess each element in the model. The preferences are entered into the pairwise comparisons in the AHP using the fundamental scale of comparisons. The Super Decision software (Version 2.10) [40] was used to count the aggregation judgments for the AHP and ANP.

In Experiment 3, the ANP was applied; this is a generalized version of the AHP. The ANP's structure is based on a network that contains several nodes or clusters. Each node consists of several elements that are needed for the decision problem. These nodes represent various criteria and their sub-criteria. In addition, the ANP network should include an alternative node that contains the proposed solutions. The main difference between the ANP and the AHP is that in the ANP, the interdependence among the network elements is taken into consideration and feedback is allowed within the network structure. In the ANP, feedback means that high-node elements can be evaluated with respect to the low-node elements, and this cannot be achieved in a hierarchical structure. Similarly to the AHP, consumers' preferences are entered into the pairwise comparisons of the ANP using the fundamental scale of comparisons. The ANP allows decision makers to maintain more consistency with a large number of elements in the network with respect to its structure; therefore, it was adopted with large numbers in Experiment 3; however, the ANP method consumes more time than the AHP due to its complicated structure.

The BWM was applied in Experiment 4 to rank several SPs, resources, and tasks. The BWM was introduced by Rezaei [41] to overcome the heavy calculations of the AHP, which makes the BWM a light pairwise decision method. The BWM proved its power and reliability in structuring decision problems and achieving consistent results when compared with other MCDM techniques. In comparison with the AHP and ANP, the BWM allows decision makers to include large numbers of decision elements while maintaining consistency and simplicity by reducing the number of computations needed. Therefore, the BWM was adopted in Experiment 4 with a larger number of elements than in the other experiments, and the Solver Linear BWM was used to calculate the judgments for the BWM adoption. The idea of the BWM is to compare the best attribute with all other attributes and to compare the worst attribute with all other attributes. Thus, the decision makers should determine the best and worst attributes at the beginning. The optimal weights of all attributes, including the best and worst attributes, are calculated based on the preferences that were entered by using the fundamental scale.

#### 4.1. Experiment 1

In the first experiment, eight tasks were prioritized and assigned to the best-matched of two SPs and four resources. SPs and resources were prioritized based on the preferences of SPs, while tasks were prioritized based on the customers' preferences. The underlying method for prioritization was the AHP. Table 2 shows all prioritized SPs and their resources along with their weights and ranks.

**Table 2.** Preference-based ranking of SPs and their resources (Experiment 1).

SP	Weight	Rank	R	Weight	Rank
1	0.67	1	1	0.09	4
			2	0.33	1
2	0.33	2	3	0.34	2
			4	0.24	3

The four criteria were then prioritized and ranked based on the customers' preferences to prioritize all of the customers' tasks. The overall results of prioritizing the criteria were found to be: reputation (0.32%), trust (0.44%), performance (0.15%), and cost (0.09%). Thus, Table 3 shows all customers' tasks, along with their weights and ranks.



**Table 3.** Preference-based ranking of tasks (Experiment 1).

T	Weight	Rank
1	0.10	6
2	0.16	2
3	0.08	7
4	0.07	8
5	0.12	4
6	0.13	5
7	0.18	1
8	0.15	3

To assign each task to either an SP or a resource in the cloud, we first needed to compute the thresholds to identify the tendency of each task. The upper threshold was considered to be the sum of all SP-based criteria weights, and the lower threshold was considered to be the sum of all R-based criteria weights. The sum of all SP-based criteria weights was 0.76 (the sum of reputation weights was 0.32, and that of trust weights was 0.44), and the sum of all R-based criteria weights was 0.24 (the sum of performance weights was 0.15, and the sum of cost weights was 0.09). Therefore, we determined all tasks for which the sum of the weights was less than the upper threshold, and we identified their tendency toward SPs. The weights of the rest of the tasks were greater than or equal to the lower threshold; therefore, their tendency was set toward resources. Note that the sum of weights considers the sum of the prioritized tasks; therefore, the rank and the threshold play important roles in determining which task tends to an SP and which does not. The results of the assignment are shown in Table 4.

**Table 4.** Categorization and assignment of tasks (Experiment 1).

Tendency	T	SP/Resource
SP-Tendency	2	SP1
	5	SP2
	6	SP2
	7	SP1
	8	SP1
	1	R3
R-Tendency	3	R2
	4	R4

After classifying all tasks' tendencies as either an SP-tendency or an R-tendency, we assigned each task to the best match with either the prioritized SPs or the prioritized resources. Any of the techniques mentioned earlier could be used for this assignment process. For simplicity, we assigned each task as follows.

For each SP,

$$N_{Ts} = \left\lfloor N_{SP-tendency} * SP_w \right\rfloor \quad (1)$$

where  $N_{Ts}$  is the number of tasks assigned to each SP,  $N_{SP-tendency}$  is the number of tasks in the SP tendency category, and  $SP_w$  is the weight of the SP.

By applying Equation (1), we calculated the number of tasks for each SP. It is important to mention that the assignment process should start from the pool of the prioritized SPs, i.e., the highest-ranked SP down to the lowest-ranked SP, in order to obtain efficient assignment. Similarly, tasks with the highest ranks should have the priority to be assigned to the best match among the available SPs.

The assignment process for the second category—concerning the R-tendency—was conducted in a similar manner but by considering the pool of prioritized resources instead. Thus, we assigned each task to a resource as follows.

For each  $R$ ,

$$N_{Ts} = \left\lfloor N_{R\text{-tendency}} * R_w \right\rfloor \quad (2)$$

where  $N_{Ts}$  is the number of tasks assigned to each  $R$ ,  $N_{R\text{-tendency}}$  is the number of tasks in the R-tendency category, and  $R_w$  is the weight of the  $R$ .

By applying Equation (2), we calculated the number of tasks for each  $R$  until we assigned all of the remaining tasks. Table 4 shows the results, where each task was classified and assigned to a resource of a certain  $SP$  or any other resource that could accommodate the customer's preferences.

#### 4.2. Experiment 2

The second experiment adopted the same underlying prioritization method, the AHP, but the difference was the increased number of  $SP$ s, resources, and tasks. The main reason for this experiment was to assess the applicability of the proposed model with a larger number of resources and tasks. An experiment was conducted to assign 12 tasks to 8 resources belonging to 5  $SP$ s (see Table 5).

**Table 5.** Preference-based ranking of  $SP$ s and their resources (Experiment 2).

SP	Weight	Rank	R	Weight	Rank
1	0.28	1	1	0.03	8
2	0.24	3	2	0.07	6
3	0.26	2	3	0.22	1
			4	0.14	5
4	0.002	5	5	0.22	2
			6	0.14	4
			7	0.04	7
5	0.21	4	8	0.15	3

The prioritization of tasks was acquired after prioritizing the four criteria. The results in this experiment differed and showed the following weights: reputation, 0.25%; trust, 0.40%; performance, 0.20%; cost, 0.15%. Table 6 shows the final weights and prioritization of the 12 tasks.

**Table 6.** Preference-based ranking of tasks (Experiment 2).

T	Weight	Rank
1	0.13	3
2	0.06	8
3	0.11	4
4	0.03	12
5	0.13	2
6	0.15	1
7	0.05	9
8	0.03	11
9	0.10	6
10	0.04	10
11	0.08	7
12	0.10	5

The upper threshold was 0.65 and the lower threshold was 0.35, which resulted in the first five tasks tending to the SP-tendency category and seven tasks tending to the R-tendency category. Similarly, Equations (1) and (2) were adopted for the prioritized tasks, and they were matched with the best option in either the pool of prioritized SPs or the pool of prioritized resources based on each task's tendency. Table 7 shows the final assignment of each task with the best match.

**Table 7.** Categorization and assignment of tasks (Experiment 2).

Tendency	T	SP/Resource
SP-Tendency	6	SP1
	5	SP1
	1	SP1
	3	SP3
	12	SP3
R-Tendency	9	R5
	11	R4
	2	R5
	7	R6
	10	R3
	8	R8
	4	R3

#### 4.3. Experiment 3

In the third experiment, we proposed the ANP as the underlying prioritization method to prioritize SPs, resources, and tasks. The experiment was conducted on 8 SPs, 12 resources, and 16 tasks. Table 8 depicts the weights and ranks after conducting the prioritization process for the SPs' preferences.

**Table 8.** Preference-based ranking of SPs and their resources (Experiment 3).

SP	Weight	Rank	R	Weight	Rank
1	0.23	1	1	0.16	1
			2	0.08	7
2	0.05	2	3	0.12	4
3	0.20	2	4	0.02	11
			5	0.11	5
4	0.17	3	6	0.06	10
5	0.11	5	7	0.14	2
			8	0.00	12
6	0.06	7	9	0.13	3
			10	0.06	9
7	0.06	6	11	0.07	8
8	0.12	4	12	0.10	6

The calculated weights of the four prioritized criteria based on customers' preferences were as follows: reputation, 0.51%; trust, 0.32%; performance, 0.12%; cost, 0.05%. Table 9 shows the final weights and prioritization of the 16 tasks.

**Table 9.** Preference-based ranking of tasks (Experiment 3).

T	Weight	Rank
1	0.08	5
2	0.12	2
3	0.12	1
4	0.03	14
5	0.04	11

**Table 9.** *Cont.*

T	Weight	Rank
6	0.04	13
7	0.09	4
8	0.01	16
9	0.06	9
10	0.07	8
11	0.01	15
12	0.08	7
13	0.09	3
14	0.05	10
15	0.08	6
16	0.04	12

The upper threshold in this experiment was 0.83 and the lower threshold was 0.12, which, as expected, allowed more tasks to be assigned to an SP regardless of the characteristics of its resources. The results shown in Table 10 illustrate the final assignment of all 16 tasks after applying Equations (1) and (2) to the prioritized SPs and resources.

**Table 10.** Categorization and assignment of tasks (Experiment 3).

Tendency	T	SP/Resource
SP-Tendency	3	SP1
	2	SP1
	13	SP3
	7	SP3
	1	SP4
	15	SP4
	12	SP8
	10	SP5
	9	SP7
R-Tendency	14	R1
	5	R7
	16	R9
	6	R3
	4	R5
	11	R12
	8	R2

#### 4.4. Experiment 4

In the last experiment, we used the BWM as the underlying prioritization method to acquire the SPs' and customers' preferences and to prioritize 10 SPs, 16 resources, and 20 tasks. The results of adopting the BWM are shown in Table 11.

**Table 11.** Preference-based ranking of SPs and their resources (Experiment 4).

SP	Weight	Rank	R	Weight	Rank
1	0.06	7	1	0.10	1
2	0.14	4	2	0.09	3
3	0.17	3	3	0.04	13
4	0.18	3	4	0.08	7
			5	0.07	8
5	0.10	5	6	0.05	11
			7	0.08	6
6	0.01	10	8	0.09	2

**Table 11.** *Cont.*

SP	Weight	Rank	R	Weight	Rank
			9	0.06	10
7	0.05	8	10 11	0.04 0.08	14 5
8	0.17	2	12	0.02	15
9	0.04	9	13 14	0.04 0.08	12 4
10	0.07	6	15 16	0.07 0.02	9 16

Similarly to the previous experiments, we calculated the weights of the four criteria to prioritize the tasks accordingly. However, the result of this experiment showed different weights, which affected each task's tendency. The weights were as follows: reputation, 0.11%; trust, 0.23%; performance, 0.38%; cost, 0.28%. Table 12 shows the final weights and prioritization of the 20 tasks.

**Table 12.** Preference-based ranking of tasks (Experiment 4).

T	Weight	Rank
1	0.01	16
2	0.05	11
3	0.025	13
4	0.10	1
5	0.02	14
6	0.09	3
7	0.00	17
8	0.00	18
9	0.01	15
10	0.09	4
11	0.00	19
12	0.09	9
13	0.06	10
14	0.10	2
15	0.08	7
16	0.08	6
17	0.08	8
18	0.00	20
19	0.04	12
20	0.09	5

The upper threshold in this experiment was 0.34 and the lower threshold was 0.66, which, as expected, allowed more tasks to be assigned to a resource regardless of its SP's characteristics. The results in Table 13 illustrate the final assignment of all 16 tasks after applying Equations (1) and (2) to the prioritized SPs and resources.



**Table 13.** Categorization and assignment of tasks (Experiment 4).

Tendency	T	SP/Resource
SP-Tendency	3	SP4
	2	SP8
	13	SP3
R-Tendency	7	R1
	1	R1
	15	R8
	12	R8
	10	R2
	9	R2
	14	R14
	5	R11
	16	R7
	6	R4
	4	R5
	11	R15
	11	R9
	11	R6
	11	R13
	11	R3
	8	R10

#### 4.5. Discussion

The results of the four experiments showed that the two algorithms were effective and correct. Regardless of the various numbers of inputs in each experiment, the main influence in the model was the underlying MCDM. Based on the conducted experiments, it is evident that each selected underlying prioritization method had a different structure for ranking and evaluating the SP and resource criteria. The AHP used in Experiments 1 and 2 used a hierarchical structure with fewer pairwise comparisons than the ANP, which was used in Experiment 3. However, the ANP allowed decision makers to represent more real problems by considering the interdependence among network elements. The time consumed by the ANP due to its complex calculations makes it less commonly used than the AHP and other ranking MCDM methods. The Super Decision software is an automated tool that can be used to facilitate calculations in both the AHP and ANP. In Experiment 4, the BWM was used as an underlying prioritization method, and the main concept of this method is to overcome the limitations of both the AHP and ANP by allowing decision makers to solve decision problems with less time and effort. The BWM depends on determining the best and worst criteria and then comparing these two over the other criteria within the decision problem with no need to evaluate each criterion with respect to all others. The maximum number of pairwise comparisons needed in the BWM is  $2n - 3$  comparisons, while the AHP, for example, requires  $n(n - 1)/2$  comparisons. In this way, the number of pairwise comparisons is minimized, which makes the BWM more suitable for use with large numbers of criteria and alternatives.

It is important to mention that the model lacks the ability to infer useful information and prioritize the SPs, customers, and tasks based on historical data. Instead, the model was only tested and validated by creating sets of inputs; this could be the main threat to its validity. Therefore, validating the model with more real datasets that incorporate historical data and enabling the model to process these data to prioritize SPs, customers, and tasks can lead to an enhancement of the efficiency and the effectiveness of the model.

#### 5. Conclusions and Future Work

This study proposed a task scheduling mechanism for properly prioritizing tasks and assigning them to various VMs before allocating them to PMs. The evaluation and task assignment process in the framework begins with three sets of inputs. All of the cloud-based SPs are included in the first set, along with a list of crucial attributes for assessing SP quality. Furthermore, the second set includes all available tasks and the criteria according to which the tasks should be evaluated by a cloud consumer. The final set

of inputs includes all available cloud resources as well as the attributes used to assess them. Moreover, the proposed framework adopts an MCDM technique to prioritize different preferences and provide a prioritized list of SPs, tasks, and resources. Different MCDM techniques can be adopted in the presented framework. For example, in Experiments 1 and 2, the AHP was adopted as an MCDM, while the ANP and BWM were adopted in Experiments 3 and 4, respectively. The AHP allows decision makers to use a top-to-bottom structure, while the ANP structure depends on a network consisting of several nodes while considering elements' interdependencies. On the other hand, the BWM structure depends on identifying the most-desired criterion and the least-desired criterion and then making a pairwise comparison with respect to these two criteria with no need to compare all of the model's items. This reduces the time needed to evaluate each SP, resource, and task when using the BWM. The primary objective of having separately prioritized SPs and prioritized resources is to make it possible to execute tasks in accordance with their tendencies.

Future work can be carried out to extend the current model so that it includes various types of inputs. For example, a broker can accommodate historical data about the performance and other criteria of SPs and their resources to give a realistic evaluation and prioritization of all available SPs. Furthermore, the model can utilize/combine other MCDM methods to expedite the task scheduling mechanism. Finally, the model should be supplemented with a robust mechanism that assures fairness when assigning tasks, and validating this supplement with real data is important.

**Author Contributions:** Conceptualization, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Investigation, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Methodology, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Project administration, A.A. (Abdulmajeed Aljuhani); Resources, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Supervision, A.A. (Abdulaziz Alhubaishy); Validation, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Writing—original draft, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy); Writing—review and editing, A.A. (Abdulmajeed Aljuhani) and A.A. (Abdulaziz Alhubaishy). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are contained within the article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

MCDM	Multi-Criteria Decision Making
AHP	Analytic Hierarchy Process
ANP	Analytic Network Process
BWM	Best Worst Method
CSP	Cloud Service Provider
VM	Virtual Machine

## References

1. Manikandan, M.; Subramanian, R.; Kavitha, M.; Karthik, S. Cost Effective Optimal Task Scheduling Model in Hybrid Cloud Environment. *Comput. Syst. Sci. Eng.* **2022**, *42*, 935–948. [[CrossRef](#)]
2. Varghese, B.; Buyya, R. Next generation cloud computing: New trends and research directions. *Future Gener. Comput. Syst.* **2018**, *79*, 849–861. [[CrossRef](#)]
3. Wang, X.; Gu, H.; Yue, Y. The optimization of virtual resource allocation in cloud computing based on RBPSO. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5113. [[CrossRef](#)]
4. Devarasetty, P.; Reddy, S. Genetic algorithm for quality of service based resource allocation in cloud computing. *Evol. Intell.* **2021**, *14*, 381–387. [[CrossRef](#)]

5. Yang, J.; Jiang, B.; Lv, Z.; Choo, K.K.R. A task scheduling algorithm considering game theory designed for energy management in cloud computing. *Future Gener. Comput. Syst.* **2020**, *105*, 985–992. [\[CrossRef\]](#)
6. Gawali, M.B.; Shinde, S.K. Task scheduling and resource allocation in cloud computing using a heuristic approach. *J. Cloud Comput.* **2018**, *7*, 4. [\[CrossRef\]](#)
7. Chauhan, N.; Agrawal, R. Probabilistic Optimized Kernel Naive Bayesian Cloud Resource Allocation System. *Wirel. Pers. Commun.* **2022**, *124*, 2853–2872. [\[CrossRef\]](#)
8. Siegel, J.; Perdue, J. Cloud services measures for global use: The service measurement index (SMI). In Proceedings of the 2012 Annual SRII Global Conference, San Jose, CA, USA, 24–27 July 2012; pp. 411–415.
9. Al-Faifi, A.; Song, B.; Hassan, M.M.; Alamri, A.; Gumaei, A. A hybrid multi criteria decision method for cloud service selection from Smart data. *Future Gener. Comput. Syst.* **2019**, *93*, 43–57. [\[CrossRef\]](#)
10. Hussain, W.; Merigo, J.M.; Gao, H.; Alkalbani, A.M.; Rabhi, F.A. Integrated AHP-IOWA, POWA framework for ideal cloud provider selection and optimum resource management. *IEEE Trans. Serv. Comput.* **2021**, *16*, 370–382. [\[CrossRef\]](#)
11. Alashaikh, A.; Alanazi, E. Conditional Preference Networks for Cloud Service Selection and Ranking With Many Irrelevant Attributes. *IEEE Access* **2021**, *9*, 131214–131222. [\[CrossRef\]](#)
12. Liu, Y.; Wang, L.; Wang, X.V.; Xu, X.; Zhang, L. Scheduling in cloud manufacturing: State-of-the-art and research challenges. *Int. J. Prod. Res.* **2019**, *57*, 4854–4879. [\[CrossRef\]](#)
13. Wu, H. Multi-objective decision-making for mobile cloud offloading: A survey. *IEEE Access* **2018**, *6*, 3962–3976. [\[CrossRef\]](#)
14. Bangui, H.; Ge, M.; Buhnova, B.; Rakrak, S.; Raghay, S.; Pitner, T. Multi-criteria decision analysis methods in the mobile cloud offloading paradigm. *J. Sens. Actuator Netw.* **2017**, *6*, 25. [\[CrossRef\]](#)
15. Krishankumar, R.; Ravichandran, K.S.; Tyagi, S.K. Solving cloud vendor selection problem using intuitionistic fuzzy decision framework. *Neural Comput. Appl.* **2020**, *32*, 589–602. [\[CrossRef\]](#)
16. Rafieyan, E.; Khorsand, R.; Ramezanpour, M. An adaptive scheduling approach based on integrated best-worst and VIKOR for cloud computing. *Comput. Ind. Eng.* **2020**, *140*, 106272. [\[CrossRef\]](#)
17. Lai, H.; Liao, H.; Šaparauskas, J.; Banaitis, A.; Ferreira, F.A.; Al-Barakati, A. Sustainable cloud service provider development by a Z-number-based DNMA method with Gini-coefficient-based weight determination. *Sustainability* **2020**, *12*, 3410. [\[CrossRef\]](#)
18. Lai, H.; Liao, H.; Wen, Z.; Zavadskas, E.K.; Al-Barakati, A. An improved CoCoSo method with a maximum variance optimization model for cloud service provider selection. *Eng. Econ.* **2020**, *31*, 411–424. [\[CrossRef\]](#)
19. Neeraj; Goraya, M.S.; Singh, D. A comparative analysis of prominently used MCDM methods in cloud environment. *J. Supercomput.* **2021**, *77*, 3422–3449.
20. Kumar, M.S.; Tomar, A.; Jana, P.K. Multi-objective workflow scheduling scheme: A multi-criteria decision making approach. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 10789–10808. [\[CrossRef\]](#)
21. Radi, M.; Alwan, A.; Abualkishik, A.; Marks, A.; Gulzar, Y. Efficient and Cost-effective Service Broker Policy Based on User Priority in VIKOR for Cloud Computing. *Sci. J. King Faisal Univ. Basic Appl. Sci.* **2021**, *22*, 1–8. [\[CrossRef\]](#)
22. Chitra, S. Multi criteria based resource score heuristic for cloud workflow scheduling. *Procedia Comput. Sci.* **2019**, *165*, 136–142. [\[CrossRef\]](#)
23. Nayak, S.C.; Parida, S.; Tripathy, C.; Pati, B.; Panigrahi, C.R. Multicriteria decision-making techniques for avoiding similar task scheduling conflict in cloud computing. *Int. J. Commun. Syst.* **2020**, *33*, e4126. [\[CrossRef\]](#)
24. Meesariganda, B.R.; Ishizaka, A. Mapping verbal AHP scale to numerical scale for cloud computing strategy selection. *Appl. Soft Comput.* **2017**, *53*, 111–118. [\[CrossRef\]](#)
25. Jatoth, C.; Gangadharan, G.; Fiore, U.; Buyya, R. SELCLOUD: A hybrid multi-criteria decision-making model for selection of cloud services. *Soft Comput.* **2019**, *23*, 4701–4715. [\[CrossRef\]](#)
26. Xu, X.; Cao, H.; Geng, Q.; Liu, X.; Dai, F.; Wang, C. Dynamic resource provisioning for workflow scheduling under uncertainty in edge computing environment. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e5674. [\[CrossRef\]](#)
27. Youssef, A.E. An integrated MCDM approach for cloud service selection based on TOPSIS and BWM. *IEEE Access* **2020**, *8*, 71851–71865. [\[CrossRef\]](#)
28. Kumar, R.R.; Kumar, C. A multi criteria decision making method for cloud service selection and ranking. *Int. J. Ambient. Comput. Intell. (IJACI)* **2018**, *9*, 1–14. [\[CrossRef\]](#)
29. Sun, L.; Dong, H.; Hussain, O.K.; Hussain, F.K.; Liu, A.X. A framework of cloud service selection with criteria interactions. *Future Gener. Comput. Syst.* **2019**, *94*, 749–764. [\[CrossRef\]](#)
30. Garg, R. MCDM-based parametric selection of cloud deployment models for an academic organization. *IEEE Trans. Cloud Comput.* **2020**, *10*, 863–871. [\[CrossRef\]](#)
31. Nawaz, F.; Asadabadi, M.R.; Janjua, N.K.; Hussain, O.K.; Chang, E.; Saberi, M. An MCDM method for cloud service selection using a Markov chain and the best-worst method. *Knowl. Based Syst.* **2018**, *159*, 120–131. [\[CrossRef\]](#)
32. Alhubaishy, A.; Aljuhani, A. The best-worst method for resource allocation and task scheduling in cloud computing. In Proceedings of the 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 19–21 March 2020; pp. 1–6.
33. Nazeri, M.; Khorsand, R. Energy Aware Resource Provisioning for Multi-Criteria Scheduling in Cloud Computing. *Cybern. Syst.* **2022**, 1–30. [\[CrossRef\]](#)

34. Malhotra, A.; Dhurandher, S.K.; Gupta, M.; Kumar, B. Integer multiplication ranking method for cloud services selection. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 2003–2017. [[CrossRef](#)]
35. Saha, M.; Panda, S.K.; Panigrahi, S. A hybrid multi-criteria decision making algorithm for cloud service selection. *Int. J. Inf. Technol.* **2021**, *13*, 1417–1422. [[CrossRef](#)]
36. Aljuhani, A.; Alhubaishy, A.; Rahmani, M.K.I.; Alzahrani, A.A. Light-Weighted Decision Support Framework for Selecting Cloud Service Providers. *Comput. Mater. Contin.* **2023**, *74*. [[CrossRef](#)]
37. Zhang, L.M.; Li, K.; Zhang, Y.Q. Green task scheduling algorithms with speeds optimization on heterogeneous cloud servers. In Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing, Hangzhou, China, 18–20 December 2010; pp. 76–80.
38. Jaybhaye, S.; Attar, V. Heterogeneous Resource Provisioning for Workflow-Based Applications Using AHP in Cloud Computing. In Proceedings of the International Conference on Communication and Computational Technologies, Virtual, 27–28 February 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 453–465.
39. Saaty, T.L. Decision making with the analytic hierarchy process. *Int. J. Serv. Sci.* **2008**, *1*, 83–98. [[CrossRef](#)]
40. Saaty, R.W.; Saaty, R. *Decision Making in Complex Environments*; Ashgate Publishing, Ltd.: London, UK, 2007.
41. Rezaei, J. Best-worst multi-criteria decision-making method: Some properties and a linear model. *Omega* **2016**, *64*, 126–130. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.