Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 2156 – 2169

doi:10.3906/elk-1807-187

A novel resource clustering model to develop an efficient wireless personal cloud environment

Mohan KOWSIGAN[1,*], **Palanisamy BALASUBRAMANIE**[2]
[1]Department of Information Technology, Sri Krishna College of Technology, Coimbatore, India
[2]Department of Computer Science and Engineering, Kongu Engineering College, Erode, India

Received: 18.07.2018 • **Accepted/Published Online:** 18.01.2019 • **Final Version:** 15.05.2019

Abstract: In the current era, cloud computing is the major focus of distributed computing and it helps in satisfying the requirements of the business world. It provides facilities on demand under all the parameters of the computing, such as infrastructure, platform, and software, across the globe. One of the major challenges in the cloud environment is to cluster the resources and schedule the jobs among the resource clusters. Many existing approaches failed to provide an optimal solution for job scheduling due to inefficient clustering of resources. In the proposed system, a novel algorithm called resource differentiation based on equivalence node potential (RDENP) is proposed for clustering the resources in a simulated wireless personal cloud environment. The performance evaluation is done among the existing and proposed approaches; as a result, the proposed RDENP algorithm produces the optimal solution for clustering the resources, which will lead to an efficient scheduling policy in a cloud environment in the future. To take this idea forward, an optimal energy consumption algorithm is to be designed to process the jobs among the resources and to minimize the infrastructure of the cloud environment by clustering the resources virtually.

Key words: Cloud computing, resource clustering, node weight, resource clustering algorithm, RDENP

1. Introduction

Cloud computing is predicted to be a great distributed technology for at least the next ten years in the business sector and economics, as it develops gradually and expands its characteristics across organizations. The future evolution of the Internet of services depends only on the cloud computing, which allows the on-demand supply of application platforms, software, and computing hardware resources [1]. Due to the enormous increase in the usage of mobile devices, cloud computing is also used in wireless mode to enrich the utilization of mobile devices in today's businesses. The structure of cloud-based wireless networks includes mobile clouds, cloud-based radio access networks, reconfigurable networks, and data centers [2]. The wireless cloud network undergoes minimum latency and well-formed communication among various users and acts as a macro relay node in the environment [3]. Many hurdles have been faced to improve the performance of the mobile environment, such as storage, bandwidth, and battery life, and these hurdles are cleared by combining a mobile environment and cloud computing [4].

This paragraph states the opportunities and challenges in cloud computing. There are many opportunities for researchers to do research in various types of resource scheduling in cloud computing such as cost-aware resource scheduling, energy-aware resource scheduling, load balancing-aware resource scheduling, efficiency-

*Correspondence: mkowsigan@gmail.com

aware resource scheduling, QoS-aware resource scheduling, and utilization-aware resource scheduling [5]. The major challenge in the current cloud environment is to build an efficient wireless personal cloud network, where cloud computing acts as the backbone of this environment. Some other challenges in cloud computing include efficiency, creativity, and simplicity in means of the task or job scheduling, VM migration and utilization, work-flow scheduling, QoS, resource optimality, and optimal resource clustering for efficient use of resources [6]. Scheduling the jobs among the resources is one of the critical tasks in a cloud environment. Makespan is one of the parameters that play a vital role in scheduling the jobs among the resources in an effective manner. Makespan is defined as the completion time of overall jobs in a cloud environment. The challenge in attaining minimum makespan, response time, and execution time and increased bandwidth and priority is related to resource clustering. There is a need for an optimal resource clustering algorithm for efficient job scheduling in the cloud environment, but most of the existing scheduling algorithms have no well-defined resource clustering policy and failed to produce an optimal solution due to their limitations such as minimum usage of resources, low-quality communication, or minimum category of services and tasks [7]. In this regard, the proposed algorithm focuses on efficient resource clustering that obtains an optimal solution for scheduling jobs among the resources in the cloud environment.

In this paper, the challenges in building a wireless personal cloud network are resolved. The reason for choosing this problem is that many of small-scale industries around the world face many issues while using their personal cloud networks. The novel algorithm, resource differentiation based on equivalent node potential (RDENP), is proposed to build an efficient wireless personal cloud network by performing optimal resource clustering for the environment. The RDENP algorithm focuses on efficient resource clustering. Here, the resources are clustered based on the weight of the resources. The existing resource clustering algorithms have many complex stages, which will increase the time complexity of the algorithm as well as the makespan of the cloud environment. The proposed RDENP algorithm undergoes various iterations to perform resource clustering in a minimal time. The performance evaluation is done for the proposed RDENP algorithm and also compared with the existing resource clustering algorithms to show the efficiency of the proposed RDENP algorithm. It is observed that the proposed RDENP algorithm performs well and attains the optimal solution for the cloud environment.

2. Related works

In this section, a detailed report on the limitations, issues of mobile wireless clouds, scheduling algorithms, and resources clustering is given. In [8], the major limitations of mobile end devices such as task offloading, storage, bandwidth, optimal power and execution time, cost, and security were discussed. To avoid low bandwidth issues two approaches were used, called femtocell and hybrid network, to improve the performance of the system. In [9], the challenges in the design of building mobile cloud applications, such as code/computation offloading, task-oriented mobile services, elasticity, and scalability, were addressed through the existing systems. Mobile cloud computing also suffers in developing an optimal execution framework and in managing heterogeneous computing environments [10]. In [11], existing challenges such as user transparent cloud discovery, unobtrusive application offloading, optimal live VM migration, seamless computational resource hand-off, and solutions for these challenges in seamless application in mobile cloud computing were discussed. The challenges in combining mobile and cloud computing and wired and wireless networks were discussed and addressed with the help of the types of networks such as vertical and horizontal [12]. A quality of service-aware dynamic pricing and scheduling algorithm [13] was proposed to overcome the challenges in the wireless cloud environment and to enrich the gain of the provider by concentrating only on the current state and length of the job queue. In [14], a survey was done of the security issues while integrating two wireless domains such as cloud computing and IoT, and it was also concluded that cloud computing increases the performance of the IoT. Efficient task scheduling [15] was used to attain the optimal solution by clustering the resources by fuzzy logic and it reduced the initialization time of the jobs across the resource clusters. The equivalence partitioning based on recurrence node weight (EPRNW) algorithm [16] was integrated with a neural network approach called the continuous Hopfield neural network to produce the near optimal solution for scheduling jobs in a heterogeneous cloud environment. The resources were clustered by EPRNW algorithm and jobs were scheduled among the resource clusters by the Hopfield neural network. An optimal task scheduling algorithm for cloud computing was obtained by processing the resources using fuzzy clustering. The earliest finish time algorithm [17] was a directed acyclic graph, which was used to schedule jobs among the resource clusters to ensure better resource utilization.

Clusters are formed among the resources by identifying the utilization of resources to improve the performance of the cloud environment. To identify suitable resources for a particular job, the concept of resource discovery was used. High success of resource discovery was achieved by an efficient resource clustering algorithm [18]. In [19], fuzzy-based resource clustering was implemented in cloud computing to raise the potential of the transitive closure method and in the meantime the complexity of the process was reduced to obtain the optimal solution. A modified hierarchal agglomerative clustering algorithm [20] was used to obtain the near optimal solution for resource discovery and a hybrid artificial bee colony and cuckoo search algorithm was used for dynamic resource allocation in a distributed environment.

To defend against DDoS and DoS attacks in the cloud environment, many possible mechanisms were discussed and provided solutions for DDoS attacks [21]. Secure sharing and video transmission was done with the help of the cloud using a genetic algorithm and error correcting codes [22]. In order to ensure the efficiency of job scheduling among the available resources in a heterogeneous cloud environment, a performance evaluation model using a continuous Markov chain and Poisson process was proposed to identify the accurate performance of the cloud environment [23]. A communication-aware directed acyclic graph model [24] was used to process cloud computing applications in order to solve the issues in resource provisioning and to increase the performance of the cloud environment. A hierarchical resource clustering algorithm [25] was proposed to reduce the time complexity of the node selection and to improve the flexibility of the environment.

Many of these works in the literature state the challenges in wireless personal clouds and the importance of scheduling algorithms and resource clustering policies in cloud environments. In real-time cloud applications, efficient resource clustering plays a vital role in scheduling the jobs among the available resources, but most of the existing resource clustering algorithms failed to produce an optimal solution or the time complexity of the algorithms increased during the formation of resource clusters in the cloud environment. Our proposed resource clustering algorithm, RDENP, addresses the existing issues and produces a near optimal solution for resource clustering in the simulated wireless personal cloud environment. A good resource clustering policy will result in an efficient scheduling process, which will be the future way forward of this proposed system.

3. Formulation of the proposed system parameters

The literature presented above clearly indicates the limitations and failure to define and formulate the parameters clearly. To overcome these issues, this section clearly defines and formulates the parameters used for the proposed resource clustering algorithm. The proposed RDENP algorithm mainly depends on the weight (potential) calculation of each and every resource. Potential calculation [14] or weight of the resource can be calculated by using the time and work equation as total work = men $\times$ days. This equation can also be reframed based on the environment as Maximum efficiency (total work) = Number of resources (NR) $\times$ Makespan (MK). By identifying the weight of resources, the algorithm proceeds to cluster the resources. The weights of the resources are calculated by using standard notations such as job and makespan, in that a job can be related to efficiency and makespan can be related to completion time. The purpose of choosing these two parameters is that they will change dynamically according to the situation because real-world data online are always dynamic in nature. In this regard, the performance of the cloud environment is formulated as

$$\text{Maximum Efficiency } = (\sum_{a=1}^{\mathbb{N}} NR_a) \times \sum_{b=1}^{\mathbb{N}} MK_b, \tag{1}$$

where NR represents the number of resources and MK represents the makespan of the cloud environment. An expression for calculating the weights of all the resources is derived from the above expression by reducing the maximum efficiency to overall resource potential by using the concept of reducibility:

$$\text{Overall resource potential of the environment } = (\frac{\text{Efficiency } (E)}{\text{Makespan } (MK)}) \times \text{ Total no. of resources } (NR). \tag{2}$$

A total of 7 resources, $R[P_0, Q_0], R[P_1, Q_1], R[P_2, Q_2], R[P_3, Q_3], R[P_4, Q_4], R[P_5, Q_5]$, and $R[P_6, Q_6]$, are used in our experiment. Weight calculation of the individual resources can be done as:

$$\text{Weight of } R[P_0, Q_0] = (\frac{\text{Maximum Efficiency}}{CT(AR) - CT(AR - R[P_0, Q_0]},) \tag{3}$$

where CT represents completion time and AR represents all resources. The above equation states that the weight of $R[P_0, Q_0]$ can be calculated as maximum efficiency (given) divided by completion time of all the resources including $R[P_0, Q_0]$ subtracted from the completion time of all the resources excluding $R[P_0, Q_0]$. The weight calculation of the remaining resources will be calculated individually in the same manner as shown in the above formula. After completing the weight calculation of the individual resources, a resource transition graph that represents the interdependency and linkage among the resources is created along with the weight flow as shown in Figure 1. This interdependency and linkage will allow the resources to exchange or retain their jobs or tasks according to their service time and resource utilization.

The weights of the resources are represented as 0s and 1s, which simplifies the relationship between the resources in the transition graph. The ultimate goal of representing the weight of the resources in 0s and 1s is to make the values simple, which makes it easy for the system's algorithm to process the weight of the resources for the purpose of clustering the resources in the wireless personal cloud environment. The weights of the resources represent the potential of the resources, according to which clusters are formed among the resources in the simulated wireless personal cloud environment. The actual weights represented in the resource transition graph are shown in Table 1. The potential or weights of the resources are calculated based on parameters such as resource utilization and completion time of the job (represented in milliseconds).

3.1. The proposed resource clustering algorithm: resource differentiation based on equivalent node potential algorithm (RDENP)

The RDENP algorithm forms clusters among the resources with the help of the resource transition graph, which identifies the ruler resource among the resources that has high potential to lead other resources. In our scenario,

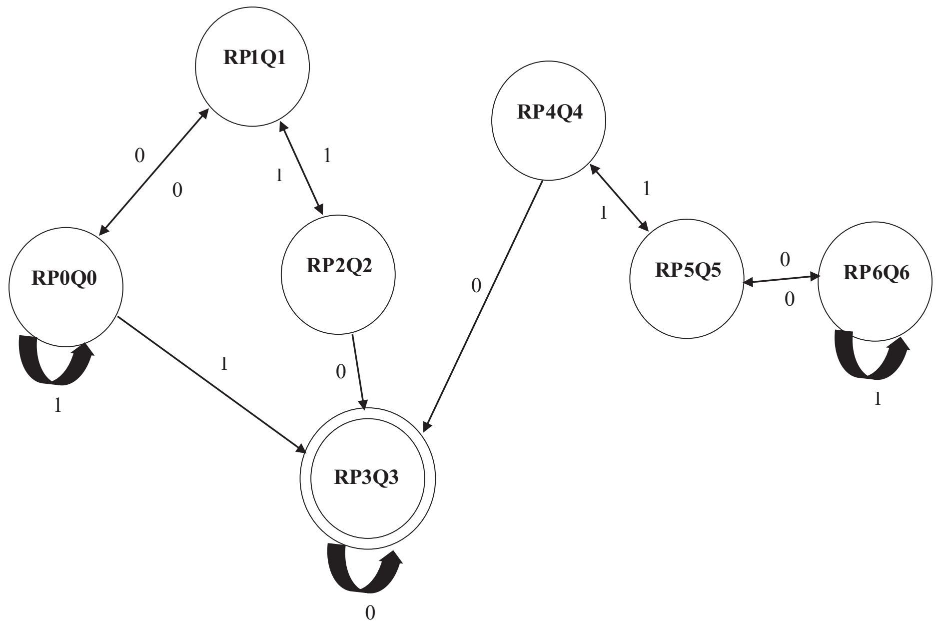

Figure 1. Resource transition graph.

Table 1. Weight flow and representations among the resources.

Resource transitions	Weight representation in 0 or 1	Cumulative weight or potential
$R_{p_0q_0} - R_{p_1q_1}$	0	0.01
$R_{p_0q_0} - R_{p_0q_0}$	1	0.15
$R_{p_0q_0} - R_{p_3q_3}$	1	0.18
$R_{p_1q_1} - R_{p_0q_0}$	0	0.030
$R_{p_1q_1} - R_{p_2q_2}$	1	0.035
$R_{p_2q_2} - R_{p_1q_1}$	1	0.213
$R_{p_2q_2} - R_{p_3q_3}$	0	0.269
$R_{p_3q_3} - R_{p_3q_3}$	0	0.033
$R_{p_4q_4} - R_{p_5q_5}$	1	0.238

three ruler resources $([R_{p_1}, R_{q_1}], [R_{p_3}, R_{q_3}], [R_{p_5}, R_{q_5}])$ are identified and clustered among themselves. This is done by comparing all the resources (R_p, R_q) in the simulated wireless personal cloud environment. The weights of the resources perform the transitions between the resources in the transition graph to identify the formation of clusters among the ruler resources and other resources. The proposed RDENP algorithm helps in identifying the equivalence among the resources, which helps in the formation of clusters. The initial step in the RDENP algorithm is to check whether the resources obey the differentiation property and identify the resources that are distinguishable from other resources. After that, the distinguishable resources are compared with the ruler resources along with their weights to identify the equivalence among the resources using the following steps.

Step 1: The resource (R_{p_0}, R_{q_0}) is checked for being distinguishable from other states.

$Transition([(R_{p_0}, R_{q_0}), (R_{p_1}, R_{q_1})], 1) = [(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})]$

$Transition([(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})], 0) = [(R_{p_1}, R_{q_1}), (R_{p_3}, R_{q_3})]$

$Transition([(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})], 1) = [(R_{p_0}, R_{q_0}), (R_{p_1}, R_{q_1})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})], 0) = [(R_{p_1}, R_{q_1}), (R_{p_3}, R_{q_3})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})], 1) = [(R_{p_0}, R_{q_0}), (R_{p_5}, R_{q_5})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_5}, R_{q_5})], 0) = [(R_{p_1}, R_{q_1}), (R_{p_6}, R_{q_6})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_5}, R_{q_5})], 1) = [(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_6}, R_{q_6})], 0) = [(R_{p_1}, R_{q_1}), (R_{p_5}, R_{q_5})]$
$Transition([(R_{p_0}, R_{q_0}), (R_{p_6}, R_{q_6})], 1) = [(R_{p_0}, R_{q_0}), (R_{p_6}, R_{q_6})]$

One of the ruler resources, (R_{p_3}, R_{q_3}), occurs in the transition of resources $[(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})]$ along with the resource (R_{p_1}, R_{q_1}). This indicates that the resources $(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})$ are distinguished or differentiated. The resources $(R_{p_0}, R_{q_0}), (R_{p_1}, R_{q_1})$ are also distinguished due to the occurrence of the resources $(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})$ in their transition, i.e. $(R_{p_0}, R_{q_0}), (R_{p_2}, R_{q_2})$ are already distinguished resources. The ruler resource (R_{p_3}, R_{q_3}) occurs in the transition of resources $[(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})]$ along with the resource (R_{p_1}, R_{q_1}). Now $(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})$ are distinguished. The resources $(R_{p_0}, R_{q_0}), (R_{p_5}, R_{q_5})$ are also distinguished due to the occurrence of the resources $(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})$ in their transition, i.e. $(R_{p_0}, R_{q_0}), (R_{p_4}, R_{q_4})$ are already distinguished resources.

Step 2: Repeat the same procedure for all the resources to identify the distinguishable resources.

Step 3: The resources that obey the differentiation or distinguishable property are said to be nonequivalent with other resources.

Step 4: Finally the clusters are formed among the equivalent resources. Here, in our simulated wireless personal cloud environment, three clusters are formed.

3.2. Pseudocode of resource differentiation based on equivalent node potential algorithm (RDENP)

Input: Resources, Weight (wg) of the resources
Output: Resource Clusters (R_{P_1} to $_n, R_{Q_1}$ to $_n$)*
Begin Clustering
Compute $n * n$ $(R_{P,Q})$; Suspecting $R_P \neq R_Q$,
i.e. R_P is differentiated with R_Q.
if $([R_P, R_Q], wg) = [a, b]$
/*a, b represents any resources in the simulated wireless personal cloud environment*/
if $[a, b] = X$ then
/* X denotes the property of differentiating the resources*/
Differentiate $[R_P, R_Q]$.
/*Cluster cannot be formed; Checked among all the P and Q resource nodes*/
else $R_P \neq R_Q$
/*Here also Cluster cannot be formed; Differentiate $[R_P, R_Q]$*/
else $R_P = R_Q$
Resources are equivalent $[R_P, R_Q]$
/*Cluster can be formed but $[R_p, R_Q]$ not differentiated*/
end if
end Clustering

Begin Differentiating the resources (Separation)
Name each and every node for its identification purpose.
/*Resources from $n[R_{P_1}, R_{P_2}, R_{P_3}$ to $R_{P_N} - R_{Q_1}, R_{Q_2}, R_{Q_3}$ to $R_{Q_N}]$*/
Estimate the weight or potential for all the resources to identify the capacity of each and every resource.
From this choose any one of the resources as the Ruler resource (X), which has high potential to lead the remaining resources.
/*Ruler resource is differentiated from other resources*/
Identify the equivalence of the resources in the cloud environment by comparing each resource with other resources including the Ruler resource
if $[(R_{P_1\ to\ n}, R_{Q_1\ to\ n}), wg] = [X(R_P, R_Q)$, any one of the resources$(R_{P_1\ to\ n}, R_{Q_1\ to\ n})]$
then $[R_{P_1\ to\ n} \neq R_{Q_1\ to\ n}]$.
/*some of the resources $[R_P, R_Q]$ are not equivalent and clusters are not formed*/
/*The resources are differentiated*/
else if $[X(Rp, RQ)$, resources $(R_{P_1\ to\ n},\ P_{Q_1\ to\ n}), wg] = [(R_{P_1\ to\ n},\ R_{Q_1\ to\ n})]$
/* Ruler Resource is excluded in $(R_{P_1\ to\ n}, R_{Q_1\ to\ n})$*/
then
$n[RP \neq RQ]$.
/*some of the resources $[Rp, RQ]$ are not equivalent and clusters are not formed*/
/*The resources are differentiated*/
end if
else $[R_{P_1\ to\ n} = R_{Q_1\ to\ n}]$.
/*some of the resources $[R_P, R_Q]$ are equivalent and clusters are formed, but the resources are not differentiated due to the equivalence property*/
end if
end Differentiating the resources (Separation)

Figure 2 specifies the working of the proposed RDENP algorithm in a flowchart. The objective of the algorithm is to cluster the suitable resources by identifying equivalence and difference among the resources. The proposed RDENP algorithm concludes by forming the suitable resource clusters to produce optimal task scheduling in the cloud environment.

4. Results and discussion

The experimental analysis is done among the various existing approaches and compared with the proposed RDENP algorithm in regard to the resource clustering in the cloud environment. Here the resources $R[P_0, Q_0]$, $R[P_1, Q_1]$, $R[P_2, Q_2]$, $R[P_3, Q_3]$, $R[P_4, Q_4]$, $R[P_5, Q_5]$, and $R[P_6, Q_6]$ are represented as 0, 1, 2, 3, 4, 5, and 6 and one cluster is formed among the ruler resources and two other clusters are formed among the equivalent resources, as shown in Figure 3.

4.1. Experimental setup

A real cloud testbed named the SKI cloud testbed has been used for this research. This testbed is subject to an educational institution called Sri Krishna Institutions (SKI). This testbed is shared among multiple servers with the ability to access a large number of wireless devices. Sensor events like GPS, telephony, and many other input devices are used to sense the inputs from the wireless devices. A blade server system with the

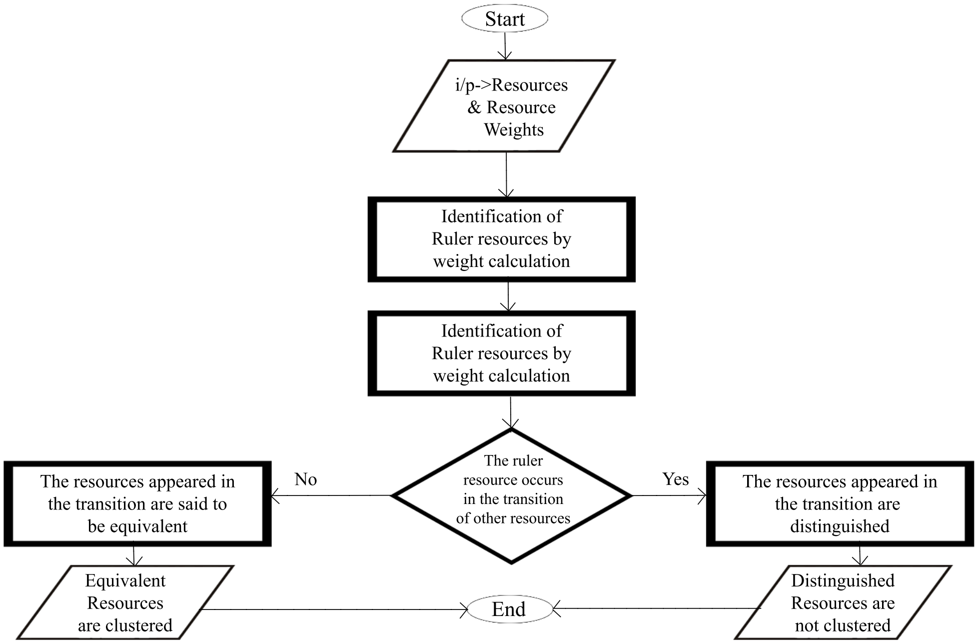

Figure 2. Working of the proposed RDENP algorithm.

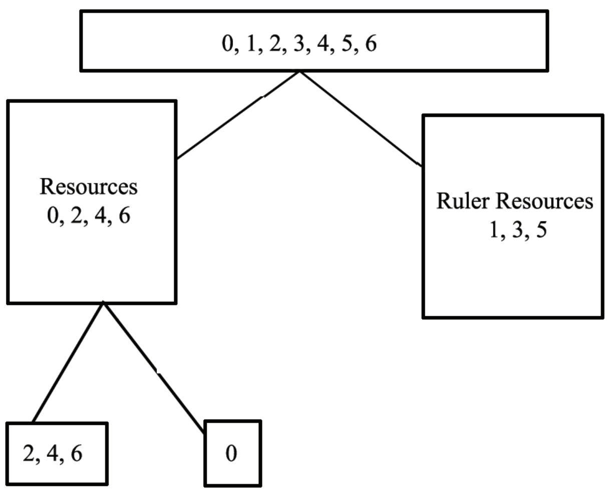

Figure 3. Ruler resources segregated from other resources.

configuration of 34x Dell PowerEdge M610 Blades with Intel Xeon 5645 processors, 36 Gb memory, 5 Tb SSD, direct attached storage (DAS) network equipment, and nearly 350 heterogeneous resources is used. Android and Cent OS are used in individual blades along with the SSH server, which acts as an administrator to maintain its client resources.

4.2. Comparison of results with the existing systems

Many of the existing approaches used in this paper are based on the hybrid fuzzy approach, which has large number of iterations in the computation, and it is also a tedious job for an algorithm to compute these approaches in the cloud environment. The fuzzy-based existing approaches used here are the index method, min–max method, correlation method, cosine method, arithmetic method, and Euclidean distance method. EPRNW was another existing approach also used for our experimental analysis and compared with the proposed RDENP approach. The resultant chart of all the existing approaches and performance comparison table of all the existing approaches with the proposed approach are shown below. For this simulated cloud environment a total of 350 resources were used. The resource clusters formed among the resources by the index method gives a percentage of 35.86271 to form a cluster containing the probability of 126.483 resource collection with an average resource weight of 3.04605. The percentages of other resource cluster formations are also shown in the resultant chart and the performance comparison table of the index method in Figure 4 and Table 2.

Table 2. Comparison table of resource cluster formation between index and min–max method.

Average resource potential	Index method	Min–max method
3.04605	35.86271	36.20277
2.02188	64.13838	63.51653
2.99137	84.06854	83.72945
3.06162	95.85297	95.62962

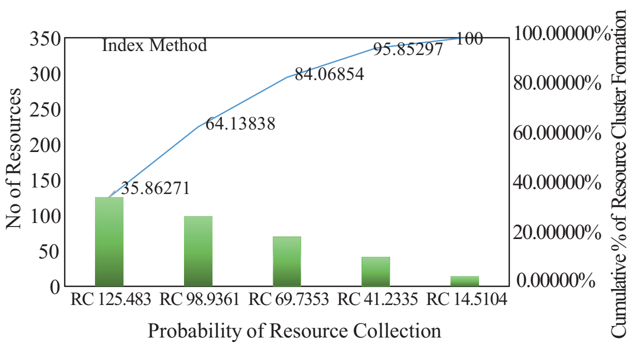

Figure 4. Resource clustering using the index method.

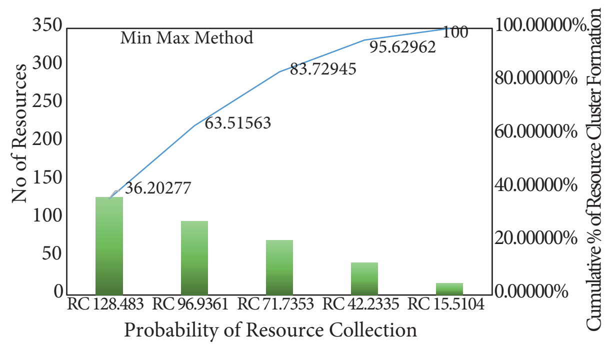

Figure 5. Resource clustering using the min–max method.

The resource clusters formed among the resources by the min–max method give a percentage of 36.20277 to form a cluster containing the probability of 128.483 resource collection with an average resource weight of 3.04605. Compared to the index method, the percentage of resource cluster formation in the min–max method was increased. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the min–max method in Figure 5 and Table 2.

The resource clusters formed among the resources by the correlation method give a percentage of 36.24961 to form a cluster containing the probability of 129.483 resource collection with an average resource weight of 3.04605. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the correlation method in Figure 6 and Table 3.

The resource clusters formed among the resources by the cosine method give a percentage of 36.42961 to form a cluster containing the probability of 126.483 resource collection with an average resource weight of 3.04605. Compared to the correlation, index, and min–max methods, the percentage of resource cluster formation in the cosine method was increased. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the cosine method in Figure 7 and Table 3.

Table 3. Comparison table of resource cluster formation between correlation and cosine method.

Average resource potential	Correlation method	Cosine method
3.04605	36.24961	36.42961
2.02188	63.66746	63.77309
2.99137	83.77823	84.17507
3.06162	95.62977	95.79191

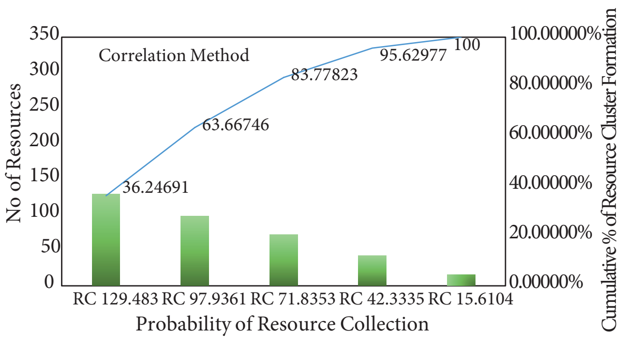

Figure 6. Resource clustering using the correlation method.

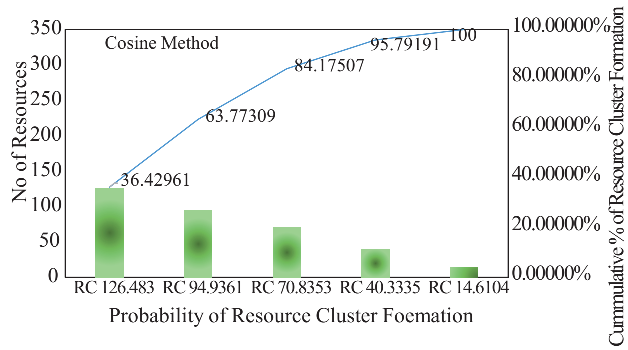

Figure 7. Resource clustering using cosine method.

The resource clusters formed among the resources by the arithmetic mean method give a percentage of 35.7492 to form a cluster containing the probability of 129.483 resource collection with an average resource weight of 3.04605. As far as the correlation, min–max, and cosine methods are concerned, the percentage of resource cluster formation in arithmetic mean method is decreased. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the arithmetic mean method in Figure 8 and Table 4.

Table 4. Comparison table of resource cluster formation between arithmetic mean and Euclidean distance method.

Average resource potential	Arithmetic mean method	Euclidean distance method
3.04605	35.7492	35.9261
2.02188	63.34074	63.16635
2.99137	83.72607	83.22021
3.06162	95.6901	95.42663

The resource clusters formed among the resources by the Euclidean distance method give a percentage of 35.9261 to form a cluster containing the probability of 130.483 resource collection with an average resource weight of 3.02605. Compared to the correlation, cosine, and min–max methods, the percentage of resource cluster formation in Euclidean distance method is decreased. Here the percentage decreases due to the calculation of the average resource weight and the probability of resource collection varies compared to other approaches. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the Euclidean distance method in Figure 9 and Table 4.

The resource clusters formed among the resources by the EPRNW method give a percentage of 39.2481 to form a cluster containing the probability of 120.471 resource collection with an average resource weight of 3.02605. Compared to the correlation, min–max, index, arithmetic mean, Euclidean distance, and cosine methods, the percentage of resource cluster formation in the EPRNW method is increased. Due to the calculation of the average resource weight and variation in the probability of resource collection, EPRNW performs well compared to other approaches. The percentages of other resource cluster formations are also shown in the resultant chart and performance comparison table of the arithmetic mean method in Figure 10 and Table 5.

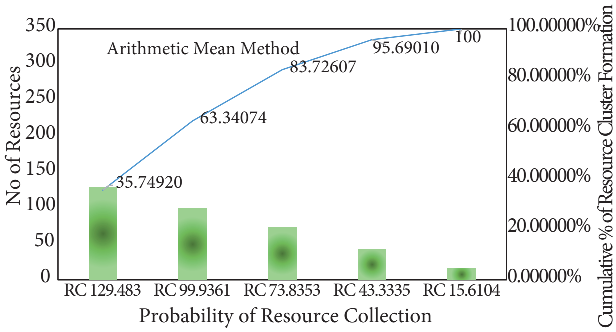

Figure 8. Resource clustering using arithmetic mean.

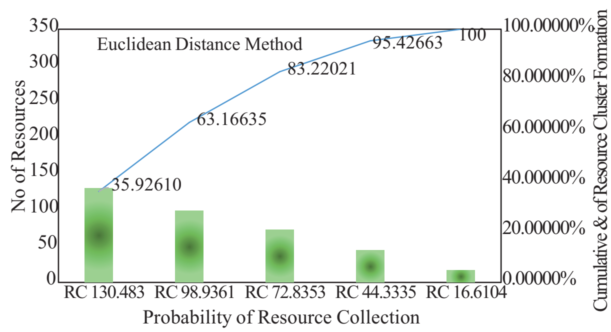

Figure 9. Resource clustering using Euclidean distance method.

The resource clusters formed among the resources by the proposed RDENP algorithm give a percentage of 39.47624 to form a cluster containing the probability of 120.122 resource collection with an average resource weight of 3.02605.

Table 5. Comparison table of resource cluster formation between EPRNW and proposed RDENP method.

Average resource potential	EPRNW method	Proposed RDENP method
3.04605	39.24811	39.47624
2.02188	68.23371	68.577
2.99137	88.25135	88.40428
3.06162	98.21286	98.30351

Compared to the correlation, cosine, index, Euclidean distance, arithmetic mean, min–max, and EPRNW methods, the percentage of resource cluster formation in the proposed RDENP method is increased. Here, the percentage increases due to the calculation of the average resource weight and variations in the probability of resource collection, and the proposed RDENP method performs well when compared to all other existing approaches. The percentages of other resources cluster formations are also shown in the resultant chart and performance comparison table of the proposed RDENP method in Figure 11 and Table 5.

From the above results, it is clearly understood that the proposed RDENP approach performs well when compared to other existing resource clustering approaches, including the EPRNW algorithm.

The probability of resource collection calculation is simple in the RDENP algorithm and the collection is formed by identifying the characteristics of the resources. The percentage of formation of resource clustering is high compared to other existing approaches. Here, the resource clustering is done by identifying the weight or potential of each and every resource and the calculation is simple and more secure when compared to other existing approaches. The performances of all the approaches are shown in Figure 12 and the proposed approach of RDENP produces the optimal solution for the resource clustering in the simulated wireless personal cloud environment.

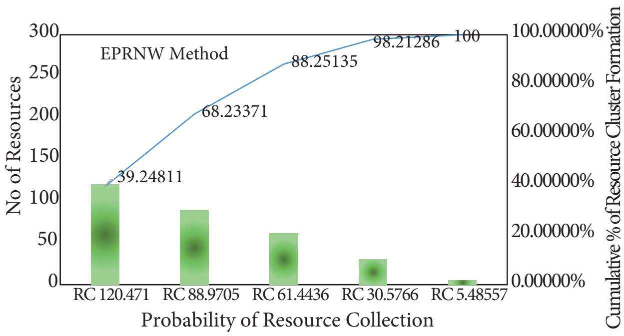

Figure 10. Resource clustering using EPRNW method.

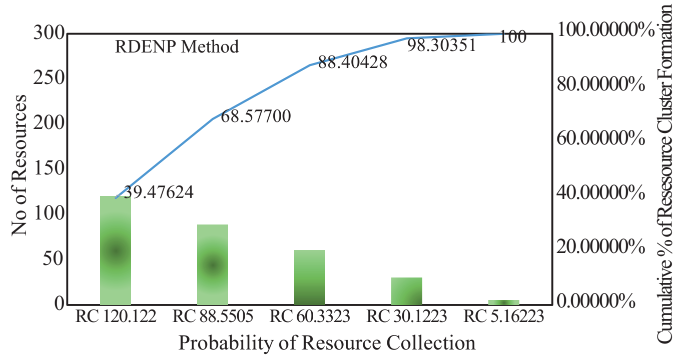

Figure 11. Resource clustering using proposed RDENP method.

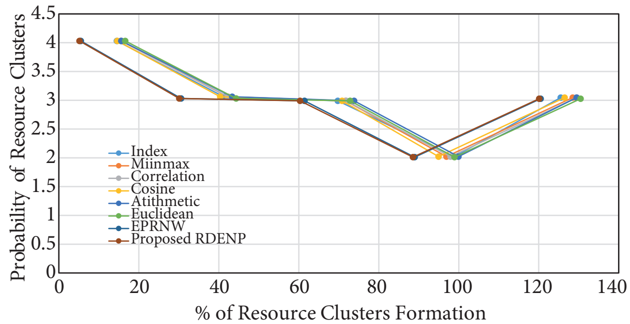

Figure 12. Comparison graph containing the percentages of resource cluster formation by the proposed RDENP approach and all the existing approaches.

5. Conclusion

The ultimate goal of this research paper is to produce an efficient resource clustering algorithm in order to obtain the optimal solution for scheduling jobs in a cloud environment. The proposed RDENP resource clustering algorithm is compared with many existing approaches and it is observed that the algorithm obtains the optimal solution for clustering the resources in the cloud environment. The cumulative percentage of resource cluster formation of the existing approaches lies between 35.7492 percent and 39.24811 percent for an average collection of resources from 120 to 129 in the cloud environment. By using the proposed RDENP algorithm, the cumulative percentage of resource cluster formation is increased to 39.47624 percent for an average collection of resources from 120 to 129 in the cloud environment. The performance comparison graph shows that the proposed RDENP algorithm forms the number of clusters among the pool of resources compared to the existing approaches. The proposed RDENP algorithm is developed in a simulated wireless personal cloud environment and it performs well. In the future, this work will be extended to minimize the infrastructure of the cloud environment by clustering the resources virtually and to develop an optimal energy consumption algorithm for processing the jobs among the resources.

References

[1] Moreno VR, Montero RS, Llorente IM. Key challenges in cloud computing: enabling the future internet of services. IEEE Internet Computing 2013; 17 (4): 18-25. doi: 10.1109/MIC.2012.69

[2] Chen M, Zhang Y, Hu L, Taleb T, Sheng Z. The role of cloud computing architecture in big data. Cloud-based wireless network: virtualized, reconfigurable, smart wireless network to enable 5G technologies. Mobile Networks and Applications 2015; 20 (6): 704-712. doi: 10.1007/s11036-015-0590-7

[3] Parichehreh A, Ramantas K, Spagnolini U, Vardakas JS. Scheduling of the super-dense wireless cloud networks. In: 2015 IEEE International Conference on Communication Workshop; 8–12 June 2015; London, UK. pp. 2263-2268.

[4] Dinh HT, Lee C, Niyato D, Wang P. A survey of mobile cloud computing: architecture, applications and approaches. Wireless Communication & Mobile Computing 2013; 13 (18): 1587-1611. doi: 10.1002/wcm.1203

[5] Syed HHM, Muhammad SAL, Yahaya C, Shafi MA. Resource scheduling for infrastructure as a service (IaaS) in cloud computing: challenges and opportunities. Journal of Network and Computer Applications 2016; 68: 173-200. doi: 10.1016/j.jnca.2016.04.016

[6] Abdulhamid SIM, Latiff MSA, Bashir MB. Scheduling techniques in on-demand grid as a service cloud: a review. Journal of Theoretical and Applied Information Technology 2014; 63 (1): 1-15.

[7] Qi H, Gani A. Research on mobile cloud computing: review, trend and perspectives. In: 2012 IEEE Second International Conference on Digital Information and Communication Technology and Its Applications; 16–18 May 2012; Bangkok, Thailand. pp. 195-202.

[8] Rahimi MR, Ren J, Liu CH, Vasilokas AV, Venkatasubramanian N. Mobile cloud computing: a survey, state of art and future directions. Mobile Networks and Applications 2014; 19 (2): 133-143. doi: 10.1007/s11036-013-0477-4

[9] Wang Y, Chen IR, Wang DC. A survey of mobile cloud computing applications: Perspectives and challenges. Wireless Personal Communications 2014; 80 (4): 1607-1623. doi: 10.1007/s11277-014-2102-7

[10] Ahmed E, Gani A, Sookhak M, Hamid S, Xia F. Application optimization in mobile cloud computing: motivation, taxonomies, and open challenges. Journal of Network and Computer Applications 2015; 52: 52-68. doi: 10.1016/j.jnca.2015.02.003

[11] Ahmed E, Gani A, Khan M, Buyya R, Khan US. Seamless application execution in mobile cloud computing: motivation, taxonomy, and open challenges. Journal of Network and Computer Applications 2015; 52: 154-172. doi: 10.1016/j.jnca.2015.03.001

[12] Sanaei Z, Abolfazli S, Gani A, Buyya R. Heterogeneity in mobile cloud computing: taxonomy and open challenges. IEEE Communications Surveys & Tutorials 2014; 16 (1): 369-392. doi: 10.1109/SURV.2013.050113.00090

[13] Wang Z, Wu J, Wu Y, Deng S, Huang H. QoS aware dynamic pricing and scheduling in wireless cloud computing. In: 2017 IEEE 56th Annual Conference on Decision and control; 12–15 December 2017; Melbourne, Australia. pp. 3702-3707.

[14] Christos S, Kostas EP, Byung GK, Brij G. Secure integration of IoT and cloud computing. Future Generation Computer Systems 2018; 78: 964-975. doi: 10.1016/j.future.2016.11.031

[15] Guo F, Yu L, Tian S, Yu J. A workflow task scheduling algorithm based on the resources fuzzy clustering in cloud computing environment. International Journal of Communication Systems 2014; 28 (6): 1053-1067. doi: 10.1002/dac.2743

[16] Kowsigan M, Balasubramanie P. An improved job scheduling in cloud environment using auto-associative memory network. Asian Journal of Research in Social Sciences and Humanities 2016; 6 (12): 390-410. doi: 10.5958/2249-7315.2016.01299.5

[17] Liu Z, Qu W, Liu W, Li Z, Xu Y. Resource preprocessing and optimal task scheduling in cloud computing environments. Concurrency and Computation-Practice and Experience 2015; 27 (13): 3461-3482. doi: 10.1002/cpe.3204

[18] Malathy G, Somasundaram RM, Duraiswamy K. Performance improvement in cloud computing using resource clustering. Journal of Computer Science 2013; 9 (6): 671-677. doi: 10.3844/jcssp.2013.671.677

[19] Youwei S. Research on cloud resource clustering based on improved method of transfer close package. Advanced Science and Technology Letters 2016; 138: 168-172. doi: 10.14257/astl.2016.138.34

[20] Durgadevi P, Srinivasan S. Optimal resource discovery and dynamic resource allocation using modified hierarchal agglomerative clustering algorithm and bi-Objective hybrid optimization algorithm. Asian Journal of Information Technology 2016; 15 (22): 4464-4474. doi: 10.3923/ajit.2016.4464.4474

[21] Gupta BB, Badve, OP. Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a cloud computing environment. Neural Computing and Applications; 2017; 28 (12): 3655–3682. doi: 10.1007/s00521-016-2317-5

[22] Shamim HM, Ghulam M, Wadood A, Biao S, Gupta BB. Cloud-assisted secure video transmission and sharing framework for smart cities. Future Generation Computer Systems 2018; 83: 596-606. doi: 10.1016/j.future.2017.03.029

[23] Kowsigan M, Balasubramanie P. An efficient performance evaluation model for the resource clusters in cloud environment using continuous time Markov chain and Poisson process. Cluster Computing 2018; 2018: 1-9. doi: 10.1007/s10586-017-1640-7

[24] Kliazovich D, Pecero JE, Tchernykh A. CA-DAG: Modeling communication-aware applications for scheduling in cloud computing. Journal of Grid Computing 2016; 14 (1): 23-39. doi: 10.1007/s10723-015-9337-8

[25] Zhao Y, Liu J, Fang X, Xu L, Du C et al. A strategy for improving netclust server placement for multicloud environments. Turkish Journal of Electrical Engineering & Computer Sciences 2018; 26 (1): 115-124. doi: 10.3906/elk-1704-206