

Improved Method of Transfer Close Package for Cloud Resource Clustering in Cloud Computing

Youwei Shao

*School of Applied Electronics, Chongqing College of Electronic Engineering,
Chongqing, China
{Youwei Shao}cqshaoyouwei@126.com*

Abstract

A cloud computing resource clustering framework based on fuzzy theory is constructed. There are five steps: characteristics and requirements set, data standardization process, data normalization process, fuzzy similarity matrix, cluster partition. In the clustering partition, the transitive closure method is improved, and the special elements in the fuzzy similarity matrix are used as the control points to simplify the optimization process and improve the optimization accuracy. The experimental results show that the cloud computing resource based on the improved transitive closure method is fast and the resource utilization rate is high after clustering.

Keywords: *cloud computing, resource clustering, transitive closure method, resource idle rate*

1. Introduction

Cloud computing technology as commercial implementations of the grid computing, distributed computing, utility computing and other traditional technologies has been used by many IT companies and research institutions, and the application throughout all walks of life[1-2]. Various resources in the cluster are available to end users in an on-demand service through Infrastructure as a Service (IaaS), Software as a Service (SaaS), Platform as a Service (PaaS), and other forms [3].

With the cluster scale of cloud computing environment is continually expanding, how to allocate resources and schedule tasks effectively and reasonably have become the emphasis and difficulty of affecting the application efficiency in cloud computing and directly affect the overall performance of the cloud platform and user satisfaction [4-6]. We need to find a suitable way to partition the target resources to narrow the search space. Cluster analysis has been widely used in decision support, data mining, pattern recognition and machine learning and other traditional areas all the time. In recent years, many scholars at home and abroad have applied the method of cluster analysis in grid computing, resource partitioning and task scheduling in the field of cloud computing [7-8].

Cluster analysis is also known as point group analysis, it is a kind of unsupervised learning method, in which according to the relation between individual objects and the degree of similarity to distinguish and classification [9]. The characteristics of the clustering results are that the individuals are similar to each other in the classes, and the individual differences are relatively large among the classes [10]. There are five kinds of clustering methods, which are based on partition, based on density, based on hierarchy, based on grid and based on the model. Among them, the most commonly used clustering method is based on partition, including hard partition and soft partition [11]. Hard partition method, such as k-means and fuzzy c-means (FCM), is based on the objective function, the individual objects are strictly divided into a class, and has the characteristic of "either this or that ". Fuzzy cluster analysis method is a soft partitioning method, which

by constructing the similar relationship between individual pairwise and according to the different similarity to partition the different clustering results, it has the characteristics of "both this and that", and be able to form a dynamic clustering effect [12]. In grid environment, Devi combined the practical application of demand preference and proposed a fuzzy clustering selection algorithm for grid resources based on application preference [13]. Alrokayan studied the scheduling problem in heterogeneous grid environment and the traditional workflow task scheduling algorithm, and proposed a heuristic task graph scheduling algorithm based on fuzzy clustering[14]. The processing unit of the target system is characterized by five resource characteristics, on the basis of this, the grid resources in heterogeneous environment are fuzzy clustering, and use the clustering results in the task scheduling as a reference and basis for the selection of the executive unit, which shortened the completion time of task scheduling and improved the performance of task scheduling. Lifeng proposed a heuristic grid fuzzy clustering resource allocation algorithm which can describe the difference of user's application preferences in large-scale, distributed, autonomous, heterogeneous and dynamic grid computing environment. The method achieved a reasonable pre classified resources, and according to the user's preferences to choose to maximize the user's goal of utility [15].

At the same time, avoid the different tasks assigned to focus on a small number of resources, so that the load balance of the grid environment can be improved. According to the various heterogeneity of service resources in grid environment and combined with the small world theory of grid service resource to model, Moschakis proposed a fuzzy clustering task scheduling algorithm for multidimensional grid resources, which improved the matching efficiency and scheduling performance of resources and tasks[16]. In order to improve the resource utilization and fast response of the complex software and hardware system, Ameen proposes a resource clustering scheduling algorithm based on fuzzy equivalent matrix. The resources are divided into three types of tasks: reading and writing memory tasks, CPU computing tasks and I/O tasks [17]. Misra proposed a new grid resource allocation algorithm based on resource and task mixture clustering, which skillfully allocating appropriate resources to fully meet the needs of the task and effectively foresee the current and future resource needs of the task to avoid the abuse of resources or unreasonable distribution [18,19]. In order to solve the problem of low efficiency and time complexity of the fuzzy equivalent matrix by using the transitive closure algorithm, he used the method of drawing net method to directly classify the fuzzy similar matrix into clusters, which improved the efficiency of computing.

From the existing research results show that, the classification of cloud resources based on traditional serial fuzzy clustering algorithm will be faced with the problem of excessive scheduling time consuming and excessive scheduling overhead, and may be faced with the usability problems that unable to cluster the large scale cluster resources. All of these can not be effectively applied to the cloud resource allocation and task scheduling in the cloud computing environment, which seriously affect the distribution and scheduling performance of the cloud resources. To this end, this paper will make an improvement on the traditional clustering methods of cloud resources, and put forward the cloud computing resources clustering method based on improved transitive closure method.

2. Fuzzy Clustering of Cloud Computing Resources

2.1. Defining Resource Characteristics and Requirements

Let the cloud cluster node set is $N = \{N_1, N_2, \dots, N_n\}$, then n represents the number of nodes in the set.

According to the characteristics of cluster and the demand of application scenarios, a variety of resource characteristics can be used to characterize the nodes in the cloud cluster and forming the resource feature set R . In this paper, we use seven resource characteristics indicators $r_0, r_1, r_2, r_3, r_4, r_5, r_6$, as the feature set attribute of Hadoop cluster. Each resource attribute in R represents resource situation and capability on each node. The demand of characteristics of resource reflects the user's preference for resources, the weight of resource demand reflects the user's tasks focus on the demand of resources[20, 21].

Let the resource characteristic demand vector $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6\}$, any one of the v_j have three possible values of -1, 0, 1. When $v_j=0$, it means the j characteristic indicator r_j is not interested; when $v_j=-1$, expressed interest in the j characteristic indicator r_j , pointing in the direction of the index is the smaller the better; when $v_j=1$, expressed interested in the j characteristic indicator r_j , pointing in the direction of the index is the bigger the better.

The business demand of cloud cluster resource clustering are differ in different application scenarios, such as computation-intensive, I/O intensive and network bandwidth, etc., The preference for the above seven kinds of resource indicators are different. In order to make the fuzzy clustering more representative, it is necessary to select the resource indexes of different stages in different scenarios to achieve the best clustering results. The resource type and the resource dimension can also be adjusted according to different scenarios, and it can define the resource weight and the satisfaction degree of resource requirement [22,23].

Therefore, there is a resource characteristic vector for each node N_k in the cluster nodes:

$$R(N_k) = (r_{k0}, r_{k1}, \dots, r_{k6}) * \begin{pmatrix} |v_0| \\ |v_1| \\ \vdots \\ |v_6| \end{pmatrix} \quad (11)$$

Where r_{kj} is the characteristic index of the j dimension for the k th nod, $|v_j|$ is the absolute value of the demand vector corresponding to the j th resource characteristic. By taking absolute value of v_j the characteristics of the resources that are not interested can be removed, and only retain the resource characteristics that are interested, the number of resource characteristics $|R| = m = \{v_j | v_j \neq 0\}$, then the original data matrix is $n \times m$ matrix.

2.2. Data Standardization Process

Because the dimension and data level of each dimension resources in the cloud resource data matrix are different, directly calculate the original resource data will lead to a greater impact on the classification of the data level [24, 25]. In order to eliminate the influence of dimension between different characteristic indexes, the r_{kj} is data standardized by the method of translation standard deviation, and the fuzzy matrix elements are obtained:

$$r'_{kj} = \frac{(r_{kj} - \bar{r}_j)}{S_j} \quad (2)$$

In the formula, $\bar{r}_j = \frac{1}{n} \sum_{k=1}^n r_{kj}$ is the mean value of the resource characteristics of the j th dimension. $S_j = \sqrt{\frac{1}{n} \sum_{k=1}^n (r_{kj} - \bar{r}_j)^2}$ is standard deviation of the resource characteristics of the j th dimension.

2.3. Data Normalization Processing

By using the shift range transform method, r'_{kj} is normalized to the interval [0, 1]:

$$r_{kj}^* = \frac{r'_{kj} - r'_{j\min}}{r'_{j\max} - r'_{j\min}} \quad (3)$$

In the formula, $r'_{j\min} = \min(r'_{1j}, r'_{2j}, \dots, r'_{nj})$, $r'_{j\max} = \max(r'_{1j}, r'_{2j}, \dots, r'_{nj})$.

2.4. Data Normalization Processing

The calculation methods for building fuzzy similar matrix are the correlation coefficient method, the distance method, the quantity product method, the arithmetic average minimum method, the exponential similarity coefficient method, the included angle cosine method, the maximum-minimum method, the geometric mean minimum method, the absolute reciprocal method etc.

Because the different columns in the original data matrix come from different host nodes, the exponential similarity coefficient method is used to calculate the level of similarity $P(N_i, N_j)$ between N_i and N_j .

$$P(N_i, N_j) = p_{ij} = \frac{1}{m} \sum_{k=1}^m e^{-\frac{3(r'_{ik} - r'_{jk})^2}{4 S_k'^2}} \quad (4)$$

In the formula, $p_{ii} = 1, 0 \leq p_{ij} = p_{ji} \leq 1, S_k'^2 = \frac{1}{n} \sum_{k=1}^n (r'_{kj} - \bar{r}_j)^2$.

At this point, we can get the fuzzy similarity matrix of the nodes set:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \quad (5)$$

According to the above analysis, the fuzzy similarity matrix is reflexive and symmetric.

2.5. Transitive Closure Method Cluster Partition

The cluster partition methods including transitive closure method and direct clustering method (drawing net method, the maximum tree method). Although the form is somewhat

different, the clustering results are the same, the core of the cluster is: N_i and N_j in the classification threshold α level of the same kind, which is equivalent to " the degree of the equivalent relationship between N_i and N_j is not less than α ".

Transitive closure method is also called square method, and the fuzzy similarity matrix can be transformed into a transitive fuzzy equivalent matrix by using the transitive closure.

Based on the fuzzy similarity matrix P , the square method is used to synthesize the operation $p^2 \rightarrow p^4 \rightarrow \dots \rightarrow p^{2^c}$:

$$p_{ij}^* = \vee \{p_{ik} \wedge p_{jk}, 1 \leq k \leq n\} = \text{Max}\{\text{Min}(p_{ik}, p_{jk}), 1 \leq k \leq n\} \quad (6)$$

When $P^* \times P^* = P^*$ occurs for the first time, it indicates that P^* is transitive, and can get the transitive closure of P :

$$P^* = \begin{bmatrix} p_{11}^* & p_{12}^* & \cdots & p_{1n}^* \\ p_{21}^* & p_{22}^* & \cdots & p_{2n}^* \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1}^* & p_{n2}^* & \cdots & p_{nn}^* \end{bmatrix} \quad (7)$$

During the execution of the program, the number of iterations $c \leq \lceil \log_2 n \rceil + 1$ and P^* is symmetric, reflexive and transitive.

$$p_{ij}^{*(\alpha)} = \begin{cases} 1 & p_{ij}^* \geq \alpha \\ 0 & p_{ij}^* < \alpha \end{cases} \quad (8)$$

$$[N_i]_{\alpha} = \{N_j \mid p_{ij}^{*(\alpha)} = 1\} \quad (9)$$

Setting the partition threshold α , the α -truncation matrix $P_{\alpha}^* = (p_{ij}^{*(\alpha)})$ of the fuzzy equivalent matrix $P^* = (p_{ij}^*)$ is obtained. The corresponding nodes that the subscript of elements $p_{ij}^{*(\alpha)}$ in the upper triangular matrix P_{α}^* (or lower triangular matrix) over 0, that is, put the nodes N_i and N_j that meet $p_{ij}^{*(\alpha)} = 1$ in a class to form an equivalent class to aggregate into the same logical subgroup.

By setting different thresholds $\alpha \in [0,1]$, different classifications can be obtained, thus forming a dynamic clustering. α more tends to 0 indicates that the inter cluster similarity degree is low; α more tends to 1 shows that the higher degree of similarity among groups; when α from 1 tends to 0, cluster partitioning from fine to coarse, thus forming different threshold partition interval $\{[1, \alpha_x), [\alpha_x, \alpha_y), \dots, [\alpha_z, 0)\}$.

3. Improved Processing for Transitive Closure Method

The complexity of cloud computing resource clustering algorithm based on transitive closure method is $O(cn^3)$, which is relatively high. To this end, this paper optimizes the step of the transitive closure clustering.

Through the analysis found that both P and P^* are diagonal matrices. and all the elements on the main diagonal are 1. Calculate p_{ij}^* need to first take the small one from the two elements of the corresponding position for the p_{ik} 、 p_{jk} two lines , and get a

length of n row matrix, p_{ij}^* is equal to maximum value of the matrix elements. First taking small later taking large, the computation for a single p_{ij}^* is $(2n - 1)$, the process decomposition as shown in Figure 1.

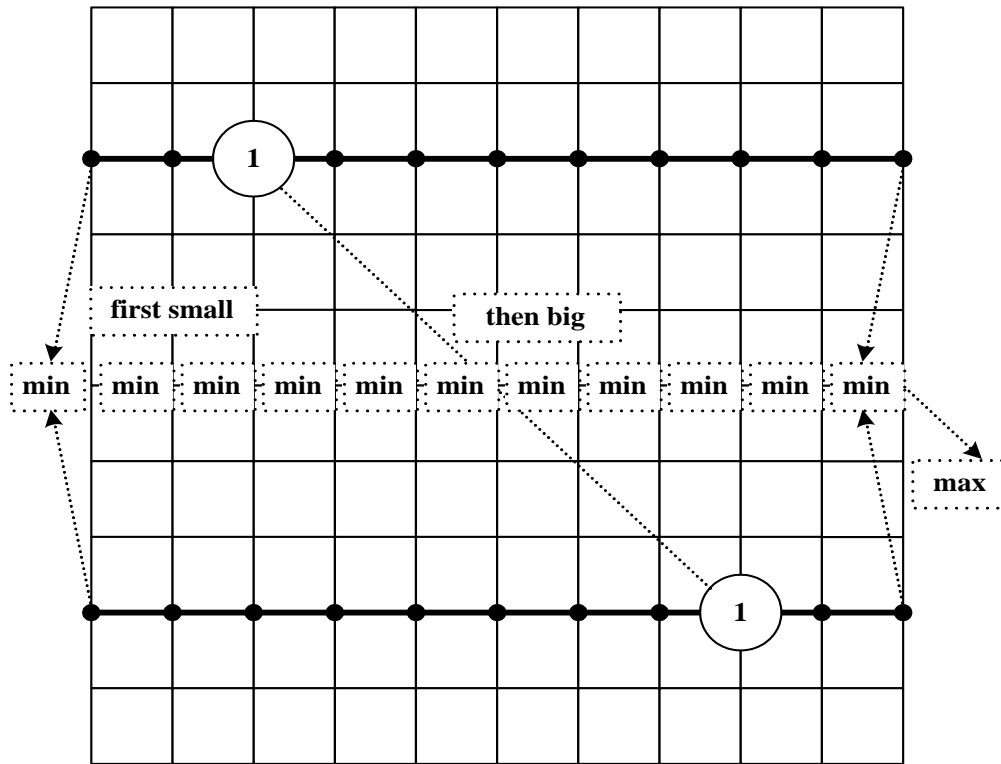


Figure 1. Solution Process Decomposition of p_{ij}^*

If there are two special elements in the matrix P , $p_{ij}(p_{ik}, k = j)$, $p_{ji}(p_{jk}, k = i)$, then:

$$\text{When } k = j, \text{Min}(p_{ik}, p_{jk}) = \text{Min}(p_{ij}, p_{ii}) = \text{Min}(p_{ij}, 1) = p_{ij};$$

$$\text{When } k = i, \text{Min}(p_{ik}, p_{jk}) = \text{Min}(p_{ii}, p_{ji}) = \text{Min}(1, p_{ij}) = p_{ji} = p_{ij};$$

The adopted optimization strategy is as follows, let $\max = p_{ij}$, \max and p_{jk} are compared one by one, after each comparison, a new round of \max will be produced., which may need again compares with p_{ik} , corresponding to the position of p_{jk} , there will be the following two situations:

The first case:

if $\max > p_{jk}$, then

$$\text{When } p_{ik} \geq p_{jk}, \text{Min}(p_{ik}, p_{jk}) = p_{jk}, \text{Max}(\max, p_{jk}) = \max ;$$

$$\text{When } p_{ik} < p_{jk}, \text{Min}(p_{ik}, p_{jk}) = p_{ik}, \text{Max}(\max, p_{ik}) = \max .$$

The second case:

if $\max < p_{jk}$, then

When $p_{ik} \geq p_{jk}$, $Min(p_{ik}, p_{jk}) = p_{jk}$, $Max(max, p_{jk}) = p_{jk}$;

When $p_{ik} < p_{jk}$, $Min(p_{ik}, p_{jk}) = p_{ik}$, $Max(max, p_{ik}) = Max(max, p_{ik})$.

On the basis of the above two cases, we can know that the optimized Max may have three kinds of results:

The first result, if $max \geq p_{jk}$, $max = max$;

The Second result, if $max < p_{jk} \leq p_{ik}$, $max = p_{jk}$;

The third result, if $max < p_{jk}$ and $p_{ik} < p_{jk}$, $max = Max(max, p_{ik})$.

The optimization strategy is analyzed from the point of view of calculation cost:

When p_{jk} appears the first result, the calculation cost required to solve a single p_{ij}^* is $G' = n - 2 = \frac{1}{2}(G - 3)$, at this point the calculation is the most ideal, the amount of computation is reduced by half;

When p_{jk} appears the second or the third result, the calculation cost required to solve a single p_{ij}^* is $G' = 2(n - 2) = G - 3$, at this time the calculation of the situation is the worst.

In practice, the calculation cost required to solve a single p_{ij}^* is between $\left[\frac{1}{2}(G - 3), G - 3 \right]$.

On the basis of the above analysis, we can know that the total computation of the fuzzy equivalent matrix is reduced by the optimized transitive closure method. It can be inferred that the time-consuming of the overall clustering operation will be reduced.

4. Experiment Result and Analysis

In order to verify the improved transitive closure resource clustering method whether can achieve the goal of reducing the time consuming and accelerating the clustering process, the traditional method for the clustering of transitive closure resources is used as a comparison method.

Experimental environment: Using sixteen core node servers, server host with different processing capability (dual-core, quad-core, 8 core, 16 core) is formed by dynamic opening and closing a plurality of cores, other hardware and software configurations and data are not described in detail here.

In experiment, we build a cloud with 16 resource nodes, the resource allocations are shown in Table 1.

Table 1. Resource Allocations of the Cloud Nodes in Experiment

	CPU	Memory	Bandwidth
Node 1	Dual-core 2.0GHz	2.0GB	100M/s
Node 2	Dual-core 2.0GHz	1.0GB	100M/s
Node 3	Dual-core 1.0GHz	2.0GB	100M/s
Node 4	Dual-core 1.0GHz	1.0GB	100M/s
Node 5	quad-core 2.0GHz	2.0GB	100M/s

Node 6	quad-core 2.0GHz	1.0GB	100M/s
Node 7	quad-core 1.0GHz	2.0GB	100M/s
Node 8	quad-core 1.0GHz	2.0GB	100M/s
Node 9	8 core 1.0GHz	1.0GB	100M/s
Node 10	8 core 1.0GHz	1.0GB	100M/s
Node 11	8 core 2.0GHz	2.0GB	100M/s
Node 12	8 core 2.0GHz	1.0GB	100M/s
Node 13	16 core 1.0GHz	2.0GB	100M/s
Node 14	16 core 1.0GHz	2.0GB	100M/s
Node 15	16 core 1.0GHz	1.0GB	100M/s
Node 16	16 core 1.0GHz	1.0GB	100M/s

Resource data of different scale clusters on the server host with different cores respectively perform the traditional transfer closure method resource clustering and improved transfer closure method resource clustering, clustering time by 5 times the average value to calculate, the results are as shown in Figure 2

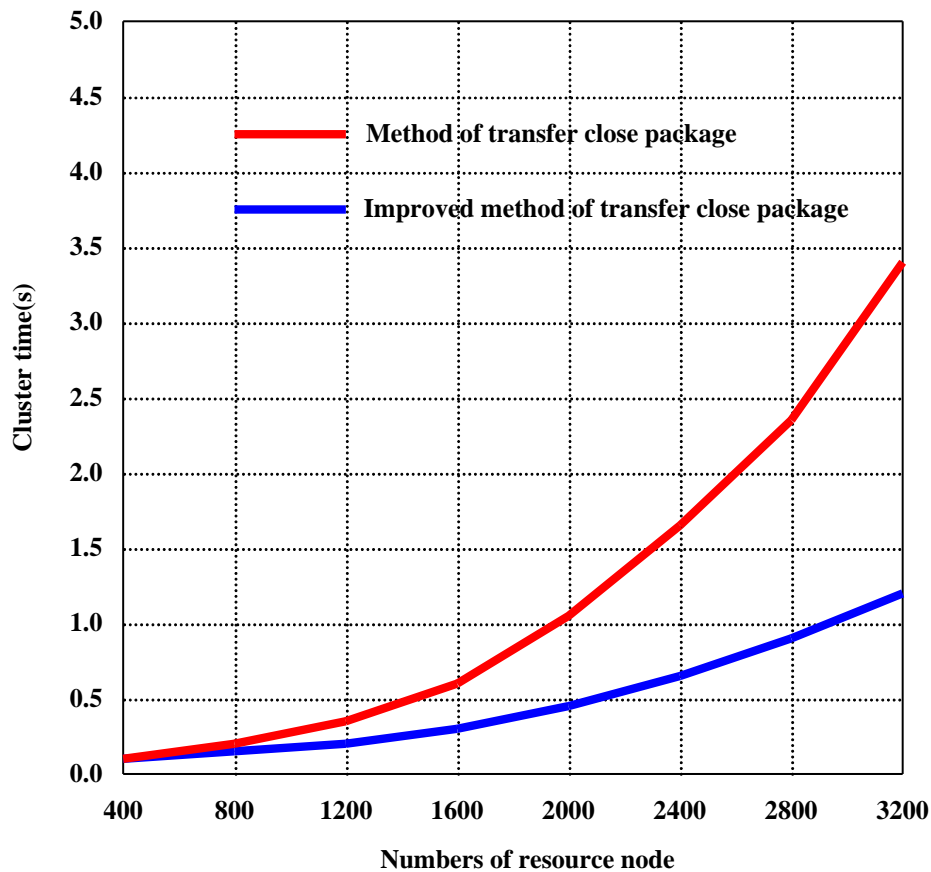


Figure 2. Ratio of Clustering Time of the Two Methods

From the results of Figure 2 can be seen: the horizontal axis represents the number of node resources, start from 400, with 400 as incremental, and gradually increased to 3200

nodes; the vertical axis represents the clustering time, in seconds; with the increase of the number of resource nodes, the clustering time is also increasing; traditional transitive closure method increases rapidly, when it added to 3200 resource nodes, the clustering time is 3.4 seconds; the clustering time of the improved transitive closure method increases slowly, and increases to 3200 resource nodes, and the clustering time is only 1.2 seconds.

The results in Figure 2 are fully illustrated that because of the decrease of computational complexity, the improved transitive closure method performs the same number of nodes in the cluster, and the clustering time is greatly reduced.

At the same time, to further investigate the CPU, memory, bandwidth and other hardware resources idle rate after the completion for the clustering of the two methods, the results shown in figure 3.

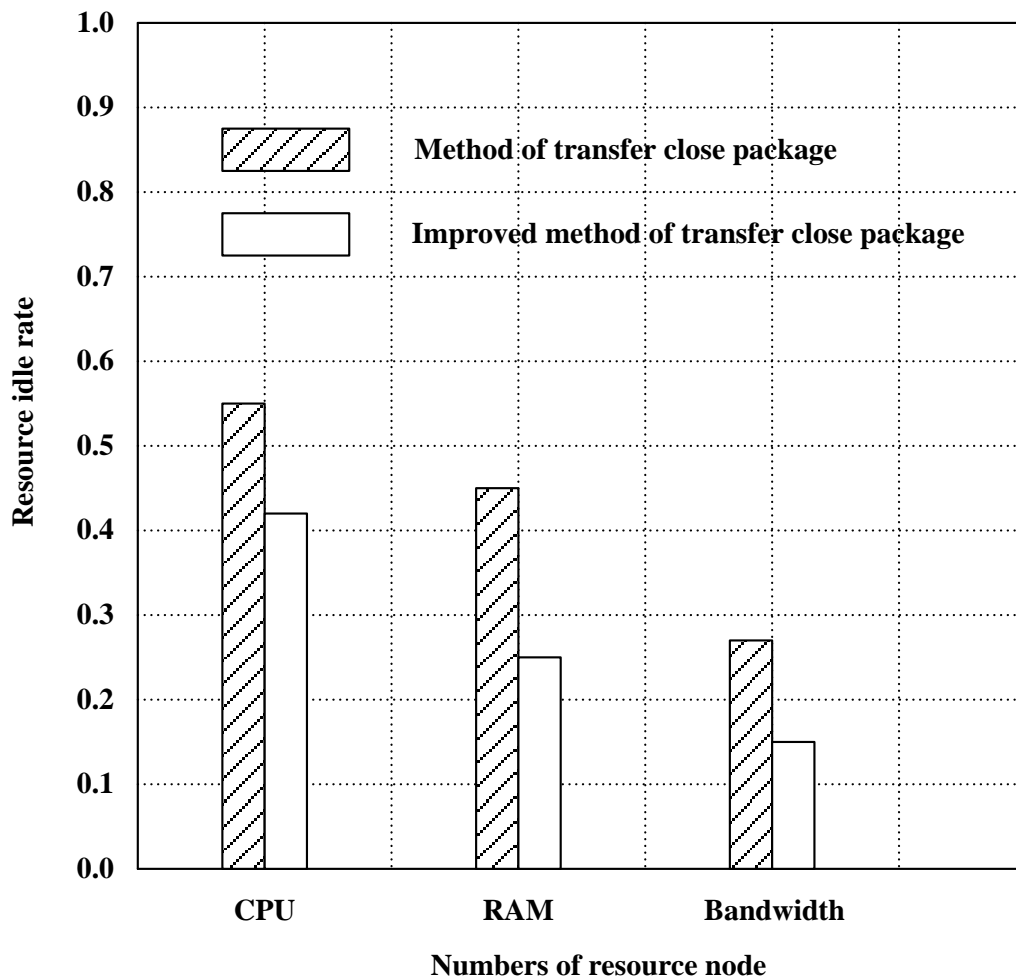


Figure 3. Comparison of Resource Idle Rate in Two Methods

The comparison results in Figure 3 can be seen that the idle rate of hardware resources after the completion of the traditional method is obviously higher than that of the improved transitive closure method, in which the idle rate of CPU has the minimum difference, the idle rate of RAM has the maximum difference. The results of the comparison show that after completing the configuration of the transfer closure method, the use of hardware resources is more reasonable, and there is not too much waste of resources.

Based on the experimental results of the two groups, we can know that after the improvement of this paper, cloud computing resource clustering based on transitive closure method can not only improve the speed, but also improve the efficiency of the configuration.

5. Conclusions

Aiming at the problem of cloud computing resource clustering, the framework of resource clustering is constructed. This framework is divided into five steps: first, definition of cloud computing resources features and user requirements; second, the relevant data were standardized processing, and unify the dimension of resources in each dimension; third, further normalizing the data after standardized, adjust the data on each dimension to a unified range; fourth, the fuzzy similarity matrix is established according to the resource attributes; fifth relying on the transitive closure method to complete the resource clustering.

Transitive closure method is of high complexity and low clustering efficiency in resource clustering. To this end, the traditional method of transitive closure was optimized. In the optimization process, the special elements in the fuzzy matrix are set to the control point, which not only simplifies the process of extremum seeking, but also improves the accuracy of the results.

The experimental results show that compared with the traditional transitive closure method, the improved method has higher clustering efficiency and higher utilization of hardware resources.

Appendix

This paper is a revised and expanded version of a paper entitled [Research on Cloud Resource Clustering Based on Improved Method of Transfer Close Package] presented at The 5th International Conference on Information Science and Industrial Applications (ISI 2016), 19-20 August 2016, Harbin, China.

Acknowledgments

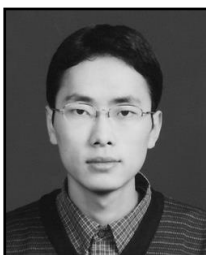
The authors thank all the reviewers and editors for their valuable comments and work. The research is supported by Chongqing Municipal Education Commission science and technology research project (KJ132202).

References

- [1] E. Mazloumi, M. Mesbah, A. Ceder, S. Moridpour and G. Currie, "Efficient transit schedule design of timing points: a comparison of ant colony and genetic algorithms", *Transportation Research Part B: Methodological*, vol. 46, no.1, (2012), pp. 217-234.
- [2] K. Dinesh and R. Zahid, "A PSO based VM resource scheduling model for cloud computing", *Proceedings IEEE International Conference on Computational Intelligence and Communication Technology, CICT*, (2015).
- [3] M. Mirabi, "Ant colony optimization technique for the sequence-dependent flowshop scheduling problem", *International Journal of Advanced Manufacturing Technology*, vol. 55, no. 1, (2011), pp. 317-326.
- [4] M. P. Sousa, A. Kumar, R. F. Lopes, T.A. Waslon and M. S. Alencar, "Cooperative Space-Time Block Codes for Wireless Video Sensor Networks", *Wireless Personal Communications*, (2012), pp.1-15.
- [5] Fang, Yadong., Wang, Fang and Wang, Hui, "Research of multi-objective optimization study for job shop scheduling problem based on grey ant colony algorithm", *Advanced Materials Research*, (2011), pp.1033-1036.
- [6] D. Wang and X. Liu, "A resource scheduling strategy for cloud computing platform of power system simulation based on dynamic migration of virtual machine", *Dianli Xitong Zidong hua/Automation of Electric Power Systems*, vol. 39, no.12, (2015), pp. 97-105.

- [7] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem", *IEEE Transactions on Evolutionary Computation*, vol.1, no. 1, (1997), pp. 53-66.
- [8] R. Paoli, F. J. Fernández-Luque, G. Doménech, F. Martínez and J. Zapata, R. Ruiz, "A system for ubiquitous fall monitoring at home via a wireless sensor network and a wearable mote", *Expert Systems with Applications*, vol. 39, no. 5, (2012), pp. 5566-5575.
- [9] R. K. Uskenbayeva, A. A. Kuandykov, Y. I. Cho and Zh. B. Kalpeyeva, "Tasks scheduling and resource allocation in distributed cloud environments", *International Conference on Control, Automation and Systems*, (2014).
- [10] F. Guo, L. Yu, S. Tian and J. Yu, "A workflow task scheduling algorithm based on the resources fuzzy clustering in cloud computing environment", *International Journal of Communication Systems*, vol. 28, no. 6, (2015), pp.1053-1067.
- [11] X. Song, L. Gao and J. Wang, "Job scheduling based on ant colony optimization in cloud computing", *International Conference on Computer Science and Service System, CSSS*, (2011), pp. 3309-3312.
- [12] I. A. Moschakis and H. D. Karatza, "Evaluation of gang scheduling performance and cost in a cloud computing system", *Journal of Supercomputing*, vol. 59, no. 2, (2012), pp. 975-992.
- [13] D. K. Vimala and K. M. Mehata, "Modelling resource estimation and scheduling for processing in cloud computing", *5th International Conference on Advanced Computing, ICoAC*, (2014).
- [14] A. Mohammed, V. D. Amir and B. Rajkumar, "SLA-aware provisioning and scheduling of cloud resources for big data analytics", *IEEE International Conference on Cloud Computing in Emerging Markets*, (2015).
- [15] L. Ai, M. Tang and C. Fidge, "Resource allocation and scheduling of multiple composite web services in cloud computing using cooperative co-evolution genetic algorithm", *Lecture Notes in Computer Science*, (2011), pp. 258-267.
- [16] I. A. Moschakis and H. D. Karatza, "Evaluation of gang scheduling performance and cost in a cloud computing system", *Journal of Supercomputing*, vol. 59, no. 2, (2012), pp. 975-992.
- [17] A. Ameen, M. Nasrullah, Hasan and Rakib, "The mechanisms to decide on caching a packet on its way of transmission to a faulty node in wireless sensor networks based on the analytical models and mathematical evaluations", *Proceedings of the 3rd International Conference on Sensing Technology, ICST*, (2008).
- [18] S. Misra, S. D. Hong, G. Xue and J. Tang, "Constrained relay node placement in wireless sensor networks: Formulation and approximations", *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, (2010), pp. 434-447.
- [19] B. Xu, Z. Peng, F. Xiao, A. M. Gates and J.-P. Yu, "Dynamic deployment of virtual machines in cloud computing using multi-objective optimization", *Soft Computing*, vol. 19, no. 8, (2015), pp. 2265-2273.
- [20] Z. Peng, B. Xu, A. M. Gates, D. Cui and W. Lin, "The Feasibility and Properties of Dividing Virtual Machine Resources using the Virtual Machine Cluster as the Unit in Cloud Computing", *KSII Transactions on Internet and Information Systems*, vol. 9, no. 7, (2015), pp. 2649 – 2666.
- [21] I. A. Moschakis and H. D. Karatza, "Evaluation of gang scheduling performance and cost in a cloud computing system", *Journal of Supercomputing*, vol. 59, (2012), pp. 975-992.
- [22] X. Song, L. Gao and J. Wang, "Job scheduling based on ant colony optimization in cloud computing", *International Conference on Computer Science and Service System, CSSS*, (2011).
- [23] R. K. Uskenbayeva, A. A. Kuandykov, Y. I. Cho and Zh. B. Kalpeyeva, "Tasks scheduling and resource allocation in distributed cloud environments", *International Conference on Control, Automation and Systems*, (2014).
- [24] P. Florin, D. Ciprian, C. Valentin and B. Nik, "Deadline scheduling for aperiodic tasks in inter-cloud environments: a new approach to resource management", *Journal of Super Computing*, vol. 71, no. 5, (2015), pp. 1754-1765.
- [25] B. B. Kadda, B. Farid, S. Faouzi and M. Mhamed, "New tasks scheduling strategy for resources allocation in cloud computing environment", *6th International Conference on Modeling, Simulation, and Applied Optimization, ICMSAO Dedicated to the memory of late Ibrahim EI-Sadek*, (2015).

Author



Youwei. Shao, he is an Associate Professor, Chongqing College of Electronic Engineering. Born in 1979, Mr Shao graduated and got master degree from Chongqing University, and his main research interests are computational intelligence and Cloud Computing.

