International Journal of Control and Automation
Vol. 9, No. 11 (2016), pp.421-428
http//dx.doi.org/10.14257/ijca.2016.9.11.36

Research on Fuzzy Clustering Method for Cloud Computing Task Scheduling

Hao Yuan, Changbing Li and Maokang Du

China Key Lab of Electronic Commerce and Modern Logistics of Chongqing
University of Posts and Telecommunications, Chongqing, China
yln2446@21cn.com

Abstract

In order to achieve scheduling of the cloud computing resource, we propose an FC algorithm based on the fuzzy clustering in this paper. There are 5 steps in executing the FC algorithm, including the requirement eigenvector construction, the data normalization, the data standardization, the fuzzy similarity matrix construction and the clustering analysis. In which we use the transitive closure method in the clustering analysis. In order to improve the clustering speed in the FC algorithm, we make equivalent operation in the fuzzy matrix construction, so to establish the IFC algorithm. As the experimental results show that, the IFC algorithm has the faster execution speed and suitable for the parallel processing, which can effectively solve the clustering scheduling problem of the cloud computing resource.

Keywords*: cloud computing, resource scheduling, clustering analysis, fuzzy matrix*

1. Introduction

In the cloud computing scenario full of heterogeneity, dynamics and fuzziness, how to divide, choose and schedule the resource in the cloud platform has a direct influence on the system resource utilization and the user satisfaction. Therefore, we need to seek a appropriate way to divide the target resource, to narrow the resource searching space .The clustering analysis method has always been widely applied into many traditional fields such as decision support, data mining, pattern recognition and machine learning, *etc* [1].

In recent years, many domestic and foreign scholars have applied the clustering analysis into the resource division and task scheduling in the grid computing and cloud computing. Maluenda proposed a grid resource fuzzy clustering choosing algorithm according to demand preference for practical applications [2]. Bilgaiyan had researched scheduling issue in grid heterogeneous environment and traditional workflow task scheduling algorithm, and proposed a heuristic task graph scheduling algorithm, which uses five resource characteristics index to describe the processing unit of the target system, on that basis to do the fuzzy clustering to the grid resource in the heterogeneous environment, and uses the clustering results as the reference of the choosing execution unit in the task scheduling, which shortens the task scheduling finish time and improves the task scheduling performance[3]. Elabady set a clustering model to the grid service resource combined with small-world theory, according to heterogeneous diversity of the service resource in the grid environment, and proposed a fuzzy clustering task scheduling algorithm facing to multi-dimensional grid resource, improving the matching efficiency and scheduling performance of the resource and task[4].Bartlett proposed a resource clustering scheduling algorithm base on the fuzzy equivalent matrix, dividing the resource into three task types: read/write memory task type,CPU computing task type and I/O task type, to improve resource utilization of complex software and hardware system and

ISSN: 2005-4297 IJCA

the fast response time tasks[5].Singh used the resource clustering division method based on the netting method, the time complexity is $O(n^2)$,calculation amount of establishing the fuzzy similarity matrix is n^2 [6]. Chaudhary used the resource clustering division method based on the transitive closure method, the time complexity is $O(n^3)$,calculation amount of establishing the fuzzy equivalent matrix is $n^3 \sim n^3 \log_2 n$ [7].Morales used the resource clustering division method based on the maximal tree,the time complexity is $O(n^3)$,calculation amount of establishing a maximal tree is no more than $\frac{3n^3}{2}$ [8].

It can be seen that the clustering division method based on the transitive closure method is most used, which is because the formula to solve the square composition operator of the fuzzy equivalent matrix is simple and easy to implement, and when facing to the resource clustering in smaller scale, the clustering time-consuming is nearly the same as the other two methods. It is known from above analysis that,the cloud resource clustering division task based on traditional serial fuzzy clustering algorithm will face the problem of seriously overweight of scheduling time-consuming and large scheduling cost ,or also maybe unable to cluster resource in large scale, all of which is unable to apply into cloud resource allocation and cloud task scheduling in large scale, so to seriously impact the cloud resource allocation and scheduling performance. Therefore, we will improve the fuzzy clustering scheduling algorithm, to increase the scheduling efficiency, decrease the scheduling time, and service better for the cloud computing task scheduling.

2. Cloud Computing Resource Scheduling Algorithm Based On the Fuzzy Clustering

2.1. Scheduling Processing Designing

In order to achieve designing the cloud computing resource scheduling algorithm,we need to clear the typical resource in cloud computing first.In this paper,we describe the resource in cloud computing in 7 factors,and form the resource index set $Z = \{z_0, z_1, z_2, z_3, z_4, z_5, z_6\}$.

z_0 ——Remaining CPU computing power(CPU basic frequency*CPU cores)*CPU idle rate,directly reflecting the whole node remaining CPU computing power(unit:GHz).

z_1 ——Idle memory amount,idle memory amount(unit:kb).

z_2 ——Remaining HDFS disk space,unused HDFS storage on the nodes(unit:kb).

z_3 —— I/O read rate,physical blocks read from disk to system buffer per second(unit:b/s).

z_4 ——I/O write rate,physical blocks write into disk from system buffer per second average (unit:b/s).

z_5 —— Network upstream bandwidth, data transmission rate from nodes to network(unit:kb/s).

z_6 —— Network downstream bandwidth, transmission rate from network to nodes(unit:kb/s).

Next,we design the clustering steps of cloud computing resource according to the fuzzy clustering ideas:

2.1.1. Define Resource Characteristic and Requirement

Suppose the cloud cluster nodes set $J=\{J_1, J_2, \cdots, J_n\}$, the amount of cluster nodes is n.

According to the cluster characteristic and scenario requirement, we can use multiple kinds of resource characteristic to describe the nodes in the cloud cluster, to form the resource characteristic set Z,in this paper, we use the above 7 resource characteristic index $z_0, z_1, z_2, z_3, z_4, z_5, z_6$ to be the characteristic set attribute of the Hadoop cluster. Each resource attribute in Z indicates the resource condition and resource capacity on each node, the resource characteristic reflects the preference of the user job task to resource, the resource requirement weight reflects requirement focus of the user job task to resource.

Suppose the resource requirement vector is $X=\{x_0, x_1, x_2, x_3, x_4, x_5, x_6\}$,in which $x_i \in \{-1,0,1\}$, $x_i=0$ means no interested in the i -st characteristic index z_i , $v_i=-1$ means interested in the i -st characteristic index z_i ,the less the direction pointing to this index is better, $v_i=1$ means interested in the i -st characteristic index z_i ,the more the direction pointing to this index is better.

The job requirement for cloud resource clustering is different in different scenarios, such as computationally intensive type, I/O intensive type, network bandwidth type *etc*, the preference to the above seven resource index varies, in order to make the fuzzy clustering more typical, we need to choose in different scenarios and different steps, to achieve best clustering result. The resource category, resource dimension can be adjusted according to different scenarios, and define the resource weight and the resource requirement satisfaction.

Therefore, each node J_k in the cloud cluster nodes set has a resource characteristic vector:

$$Z(J_k)=(z_{k0}, z_{k1}, z_{k2}, z_{k3}, z_{k4}, z_{k5}, z_{k6}) * \begin{pmatrix} |x_0| \\ |x_1| \\ |x_2| \\ |x_3| \\ |x_4| \\ |x_5| \\ |x_6| \end{pmatrix} \quad (1)$$

In which, z_{kj} is the j -st dimensional characteristic index of the k -st node, $|x_j|$ is the absolute value of the requirement vector corresponding of the j -st resource characteristic. The uninterested resource characteristic can be removed by take absolute value of x_j ,only remaining interested resource characteristic, the resource characteristic amount is $|Z|=m=\{x_j \mid x_j \neq 0\}$,the original data matrix of cloud cluster $(z_{ij})_{n\times m}$ is $n\times m$.

2.1.2. Data Standardization

As the physical dimension and data grade of each dimensional resource in the cloud resource data matrix are different, calculating the original resource data directly will make the characteristic index in lager data grade impact on the category seriously, in order to diminish the physical dimensional influence between different characteristic indexes, we

need to use translation standard deviation transformation, to data standardize z_{kj} ,and get the fuzzy matrix element:

$$z'_{kj} = \frac{z_{kj} - \bar{z}_j}{S_j} \tag{2}$$

In which, $\bar{z}_j = \frac{1}{n}\sum_{k=1}^{n} z_{kj}$ is the average of the j -st dimensional resource characteristic, $S_j = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(z_{kj} - \bar{z}_j)^2}$ is the standard deviation of the j -st dimensional resource characteristic.

2.1.3. Data Normalization

Normalize z'_{kj} to range [0, 1] with translation standard deviation transformation, to get:

$$z'_{kj} = \frac{z'_{kj} - z'_{j\min}}{z_{j\max} - z'_{j\min}} \tag{3}$$

In which, $z'_{j\min} = \min(z'_{1j}, z'_{2j}, \cdots z'_{nj})$, $z'_{j\max} = \max(z'_{1j}, z'_{2j}, \cdots z'_{nj})$

2.1.4. Establish Fuzzy Similarity Matrix

The calculation to establish fuzzy similarity matrix includes the correlation coefficient method, distance method, quantity product method, arithmetic average minimum method, exponential similarity coefficient method, angle cosine method, maximum and minimum method, geometric mean minimum method, reciprocal of absolute value method, *etc*. As different arrays in the original data matrix are from different host nodes, we need to calculate the similarity $S(J_i, J_j)$ between node J_i and J_j with the exponential similarity coefficient method.

$$S(J_i, J_j) = p_{ij} = \frac{1}{m}\sum_{k=1}^{m} e^{-\frac{3}{4}\bullet\frac{(z'_{ik} - z'_{jk})^2}{(S'_k)^2}} \tag{4}$$

Finally, we can get the ultimate fuzzy similarity matrix:

$$S = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1n} \\ S_{21} & S_{22} & \cdots & S_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ S_{n1} & S_{n2} & \cdots & S_{nn} \end{bmatrix} \tag{5}$$

2.1.5. Clustering Division

We need to use the transitive closure method,which can transform the fuzzy similarity matrix to the transitive fuzzy equivalent matrix.

Synthesis operate $S^2 \to S^4 \to \cdots \to S^{2c}$ with transitive closure from the fuzzy similarity matrix S :

$$\begin{aligned} S_{ij}^* &= \vee\{s_{ik} \wedge s_{jk}, 1 \le k \le n\} \\ &= Max\{Min(s_{ik}, s_{jk}), 1 \le k \le n\} \end{aligned} \tag{6}$$

When $S^* \times S^* = S^*$ appears for the first time,it means S^* is transitive,so we can get the transitive closure of S.

$$S^* = \begin{bmatrix} S_{11}^* & S_{12}^* & \cdots & S_{1n}^* \\ S_{21}^* & S_{22}^* & \cdots & S_{n2}^* \\ \cdots & \cdots & \cdots & \cdots \\ S_{n1}^* & S_{n2}^* & \cdots & S_{nn}^* \end{bmatrix} \tag{7}$$

S^* has the symmetry, reflexivity and transitivity.

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

$$[J_i]_\alpha = \left\{ J_j \mid S_{ij}^{*(\alpha)} = 1 \right\} \tag{9}$$

Put together the nodes which are corresponding with subscript of $S_{ij}^{*(\alpha)}$ constructed with the element more than zero in the upper triangle(or lower triangle) region in S_α^*,which also means the nodes J_i and J_j meeting the condition $S_{ij}^{*(\alpha)}$,to constitute equivalence classes, and cluster a logical subgroup.

2.2. Fuzzy Clustering Optimization

The feature of the transitive closure method is that, iterate for $c \leq \lceil \log_2 n \rceil + 1$ times to get the transitive closure, and we need to calculate $\frac{n(n-1)}{2}$ new elements in each iteration, and then clustering divide the fuzzy equivalent matrix S^* with division threshold. The time complication of this method is $O(c \cdot n^3)$.

In the transitive closure method, the fuzzy equivalent processing costs almost 90% times of the overall clustering time-consuming, when facing the cloud resource clustering division of more than thousands of nodes, the time-consuming cost of getting scheduling decision basis can not be guaranteed. In this paper, the ideal clustering time-consuming of cloud resource division according to the fuzzy clustering theory is a minute or even shorter. Therefore, we need to decrease the calculation of the fuzzy clustering algorithm, to shorten the resource clustering time-consuming and break unavailability of the clustering scale problem, to accomplish clustering division to obtain division result in short time as possible. We can never continue researching the problems such as the cloud resource allocation and task scheduling until guaranteeing the extra cost of resource scheduling in a rational range.

In the transitive closure method, S and S^* are both diagonal matrixes, and the elements on the main diagonal are all 1,however,calculating S_{ij}^* needs to take smaller one between each two element in the corresponding position in the rows S_{ik} and S_{jk},to get a row matrix with the length n, then S_{ij}^* is equal to the maximum value of the element in the row matrix. For this reason, we take the following optimization strategies:

Make $\max = S_{ij}$, and compare $\max$ and S_{jk} one by one, to generate a new $\max$ after each comparison, in which we maybe need to compare it and S_{ik} on the corresponding position of S_{jk} again, the following two situations may happen:

The first one:if $\max > S_{ij}$,the following two possibilities may happen:

(1)If $S_{ik} \geq S_{jk}$, then $Min(S_{ik}, S_{jk}) = S_{jk}$, $Max(\max, S_{jk}) = \max$;

(2)If $S_{ik} < S_{jk}$, then $Min(S_{ik}, S_{jk}) = S_{ik}$, $Max(\max, S_{ik}) = \max$.

The second one: if $\max < S_{ik}$,the following two possibilities may happen:

(1)If $S_{ik} \geq S_{jk}$, then $Min(S_{ik}, S_{jk}) = S_{jk}$, $Max(\max, S_{jk}) = \max$;

(2)If $S_{ik} < S_{jk}$, then $Min(S_{ik}, S_{jk}) = S_{ik}$, $Max(\max, S_{ik}) = Max(\max, S_{ik})$.

3. Experiment and Analysis

In order to test the effectiveness of the clustering analysis method and optimization strategy proposed in this paper,we take the following researches.

The experimental environment, AMAX sever with 16 cores, open or close several cores dynamically, to form severs with different CPU processing capacity(dual cores, quad cores, eight cores ,sixteen cores).FC(fuzzy clustering) means the fuzzy clustering method constructed initially. IFC means the optimized fuzzy clustering method.

Execute the FC algorithm and IFC algorithm with cloud computing resource in different Hadoop scales and in severs with different CPU cores. As nodes amount in the cluster increases, the clustering time of FC algorithm and IFC algorithm are as shown in Figure 1.

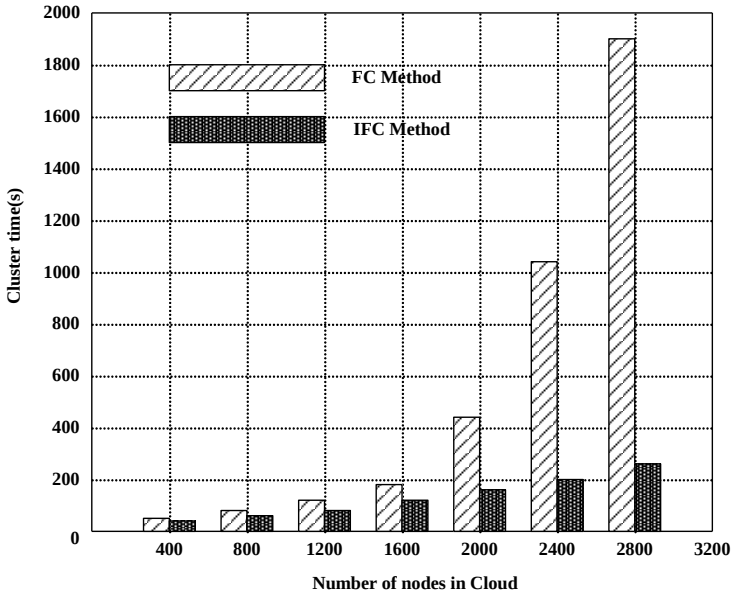

Figure 1. Cloud Computing Resource Clustering Time Comparison of the Two Algorithms

In Figure 1, FC and IFC are both calculating in the sever with quad cores. As the result shows that, the two algorithms can both accomplish the cloud computing resource clustering task. Meanwhile, the time-consuming of IFC reduces significantly to 20%. This is because the total calculation of solving the fuzzy equivalent matrix decreases after optimized with the transitive closure method. After multiple iterations, the time-consuming reduction of solving the fuzzy equivalent matrix will influence the whole clustering.

Next, we test the executive efficiency of the IFC in parallel strategy further. In which, we use a speed-up ratio parameter η to analyze the effects of IFC in different clustering resource data sets. The calculation of the parameter is as follows:

$$\eta = \frac{t_s}{t_m}$$

In which, t_s indicates the serial execution time of IFC on single core CPU; t_m indicates the parallel execution time of IFC on multiple cores CPU.

In the experimental environment set in this paper, the experimental result of parallel execution IFC algorithm is shown in Figure 2.

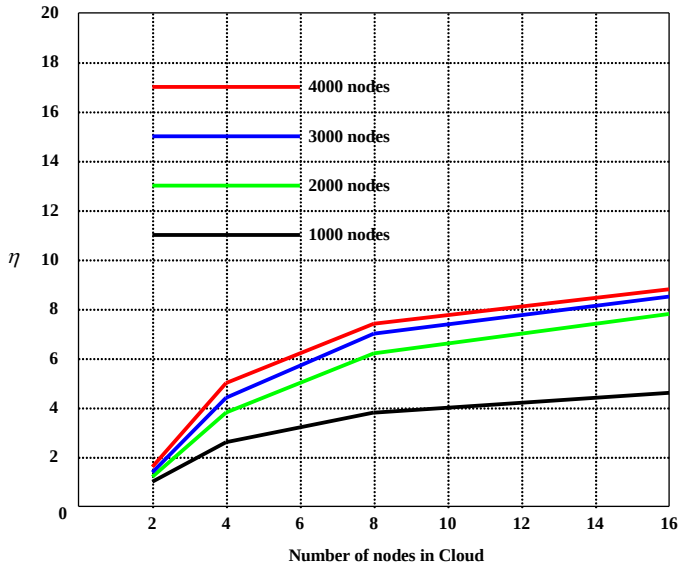

Figure 2. Speed-Up Ratio of Parallel Execution with IFC

The speed-up ratio is shown in Figure 2, as the CPU cores increases, the speed-up ratio of clustering algorithm is bigger and bigger, slowly approaching to linear speed-up ratio, which indicates the optimized clustering algorithm of multiple processes parallel has a preferable executive efficiency.

4. Conclusions

Pointing at the clustering problem in cloud computing resource, we design a FC algorithm in this paper, and optimize it to form a IFC algorithm. FC algorithm establishes a fuzzy similarity matrix for clustering analysis by constructing the resource requirement eigenvector, and executing data normalization and standardization. In the final phase of clustering analysis, we use the transitive closure method. To reduce the executive time of FC algorithm, we take a equivalent algorithm in constructing the fuzzy matrix, which reduces the calculation significantly, finally forming the IFC algorithm. As the experimental results show that, the IFC algorithm not only takes performance of rapid clustering, but also especially suitable for parallel processing.

References

[1] K.A Saranu and J. Suresh, "Intensified scheduling algorithm for virtual machine tasks in cloud computing", Advances in Intelligent Systems and Computing, vol. 3, no. 25, **(2015)**, pp. 283-290.
[2] D. Maluenda, A. Carnicer, R. Martnez-Herrero, I. Juvells and B. Javidi, "Optical encryption using photon-counting polarimetric imaging", Optics Express, vol. 23, no. 2, **(2015)**, pp. 655-666.
[3] S. Bilgaiyan, S. Sagnika and M. Das, "A multi-objective cat swarm optimization algorithm for workflow scheduling in cloud computing environment", Advances in Intelligent Systems and Computing, vol. 1, no. 11, **(2015)**, pp. 73-84.

[4] N. F. Elabady, H.M. Abdalkader, M.I. Moussa and S.F. Sabbeh, " Image encryption based on new one-dimensional chaotic map", 2014 International Conference on Engineering and Technology (ICET), Cairo, **(2014)**.
[5] A.C. Bartlett, A.A. Andales, M. Arabi and T.A. Bauder, "A smartphone app to extend use of a cloud-based irrigation scheduling tool", Computers and Electronics in Aghriculture, vol. 1, no. 11, **(2015)**, pp. 127-130.
[6] H. Singh, A.K Yadav, S. Vashisth and K. Singh, "Double phase-image encryption using gyrator transforms, and structured phase mask in the frequency plane", Optics and Lasers in Engineering, vol. 6, no. 7, **(2015)**, pp. 145-156.
[7] D. Chaudhary and B. Kumar, "An analysis of the load scheduling algorithms in the cloud computing environment: A survey", 9th International Conference on Industrial and Information Systems, Gwalior, **(2014)**.
[8] Y. Morales, L. Daz, C. Torres and Radia "Hilbert transform in terms of the fourier transform applied to image encryption", Journal of Physics Conference Series, vol. 1, no. 2, **(2015)**, pp. 582-590.

Authors

Hao Yuan, he is an Associate Professor, Chongqing University of Posts and Telecommunications. Born in 1970, Mr Yuan graduated and got master degree from Chongqing University, and his main research interests are computational intelligence and Cloud Computing.

Changbing Li, he is an Professor, Chongqing University of Posts and Telecommunications. Born in 1970, Mr Li graduated and got doctor degree from Chongqing University, and his main research interests are computational intelligence and Cloud Computing.

Maokang Du, he is an Professor, Chongqing University of Posts and Telecommunications. Born in 1969, Mr Du graduated and got bachelor degree from Xinan University, and his main research interests are computational intelligence and Cloud Computing.

