2016 IEEE 22nd International Conference on Parallel and Distributed Systems

Resource Scheduling Based on Improved FCM Algorithm for Mobile Cloud Computing

Wu Hong-Qiang , Li Xiao-Yong ,Fang Bin-Xing, Wang Yi-Ping
Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education
School of Computer Science and Technology, Beijing University of Posts and Telecommunications
Beijing, China

***Abstract*—With the development of mobile devices, mobile cloud computing is becoming increasingly important. One of the basic questions in mobile cloud computing is how to match user demand with cloud server resources. Based on Improved FCM (IGAFCM) Algorithm, this paper proposes a scheduling scheme which is provided for mobile resources to cluster continuously, so as to reduce the size of the matching requirements during the search. Moreover, Experiments have proved that matching strategy is dynamically adjusted according to the matching score and feedback training.**

Keywords—Mobile cloud computing, resource scheduling, resource clustering, FCM algorithm

I. INTRODUCTION

Cloud computing has a wide range of applications, and ordinary users can buy cloud service from some providers such as Amazon or ALIBABA cloud server, on which they set up their own web sites and applications. With the rapid development of cloud computing, mobile cloud computing (MCC)[1] also has become a new research hotspot.

Existing resource scheduling strategy based on MIN-MIN algorithm is widely used because of the simple and feasible function. However, MIN-MIN algorithm can cause load imbalance easily, and during the process of scheduling, it lacks consideration for the mobile individual needs.

In this paper, the problem of resource scheduling and resource dynamic monitoring in mobile cloud computing is proposed[2]. Therefore, Improved FCM (IGAFCM) Algorithm can be chosen to solve these problems. First, the matching scale is reduced by FCM Algorithm. Then, the individual needs can be classified into different classes, and they will be matched with server resources in the same class. The scheduling server will give a set of the Top-N results by scoring the matches. Finally, combing with a real time calculating in those Top-N results, the scheduling server give the most decided upon answer to the individual requirement.

The main contributions can be summarized as follows:

1) This paper proposes an Improved FCM (IGAFCM) Algorithm to deal with resources scheduling in MCC.

2) A scheduling model is developed, which includes the clustering resources model and user matching model.

3) The IGAFCM is based on these models to reduce the scale of matching by FCM and solute the dynamic changing problem by re-clustering.

The rest of this paper is organized as follows. Scheduling model of mobile cloud computing will be described in Section II. In Section III, this paper proposes the details and work principles of IGAFCM in MCC. In Section IV, experiments are presented and theoretical proof are given. Section V is the conclusion in the paper.

II. MOBILE CLOUD COMPUTING SCHEDULING MODEL

In the dispatching of mobile cloud computing, task scheduling and resource scheduling are faced with the problem of huge amounts of data. The most important problems are resource scheduling and scheduling model with high scalability and throughput [3]. In recent years, many scholars introduce the method of fuzzy clustering to the field of cloud computing. The effect of fuzzy clustering [4] is reducing the scale of resource data during the scheduling process, and it can make user demands in a certain class match the appropriate resources quickly. Simultaneously, through the classification screening, user satisfaction can be improved in first schedule matching.

However, the traditional fuzzy clustering algorithm in cloud computing is only for static clustering in scheduling, and there have no better identification and processing method for small and large scale changing resources. Once re-cluster is beginning, the cloud system efficiency will decline.

Therefore, based on the traditional FCM and GA algorithm, this paper puts forward an Improved FCM(IGAFCM) algorithm to increase monitoring indicators of resources changing [5]. Meanwhile, in resource class reunion, IGAFCM can greatly accelerate the convergence speed of the reunion objective function and improve the efficiency of re-clustering by providing the experience value of clustering points.

This work is supported by the National Nature Science Foundation of China (No. 61370069), Beijing Natural Science Foundation (No. 4162043), Fok Ying Tung Education Foundation (No. 132032) and Program for New Century Excellent Talents in University (No. NCET-12-0794).

1521-9097/16 $31.00 © 2016 IEEE
DOI 10.1109/ICPADS.2016.24

A. Mobile Resource Clustering Model

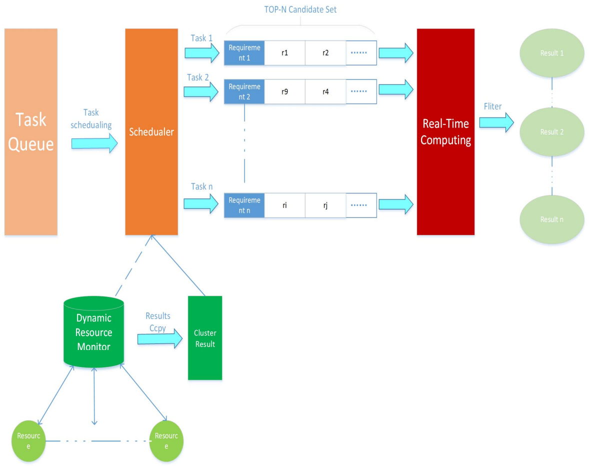

Fig. 1. Mobile Resource Clustering Model

To avoid the influence of clustering process on the system performance, the clustering model adopts the method of hierarchical processing. As shown in Fig.1, clustering results are stored in the scheduling server, and mobile user demands come into the task queue to wait for schedule dispatching. When resources are changed, the scheduling server monitor the changing class. Once the cluster center C_i and average degree of membership U_{ij} occur large drift, scheduling server will choose different strategy in order to deal with different changing situations: 1) Increase the number of re-cluster categories for added resources; 2) Reduce the number of re-cluster categories for decreased resources; 3)Just re-cluster for resources changing, while the number of resources is not changing.

To increase the re-cluster speed, the central point of the last clustering is used as an empirical point to the new clustering process. These points can help improve the convergence rate of the objective function and the efficiency of the class reunion.

In the above steps, the server scheduling results are according to the cluster queue before, and update the queue until re-cluster is completed. Although re-clustering steps need extra processing overhead, the task scheduling is normal.

B. Resource Model

In MCC, resource properties can be divided into three categories: calculation type, storage type and tele-communication type [6], and each category contains various small properties. In order to reduce the dimensions of resource clustering, it is reasonable to use fuzzy processing methods to normalize these small properties. For example, a cloud service provider has hardware servers in Beijing and Hangzhou, service resources can be divided according to regions. CPU and GPU are different attributes, and the weighted average is a good way to unify the calculation ability. By normalizing the similar attributes, it can greatly reduce the dimension of data processing in cloud computing. As shown in the following formula.

$prop_i$ is the ith property, and M_i represents its weight.

$$\overline{prop} = \frac{\sum_{i=1}^{n} prop_i * M_i}{\sum_{i=1}^{n} M_i} \tag{1}$$

C. User Matching Model

Although fuzzy clustering method can quickly provide qualified resources, the results are not optimal, which because the precision of properties has been blurred [7]. In order to improve the service quality and customer satisfaction [8], in the matching model, filtered results are added to the queue of candidate set by score calculating, and the candidate resources can be sorted through setting the scheduling algorithm in a queue. Top-N values in the queue are putted into real-time calculation, and users will obtain the most satisfied results. Meanwhile, the process can be assigned to the resources of their own calculation to get better results through the concurrent processing. The User Matching Model is shown as Fig. 2.

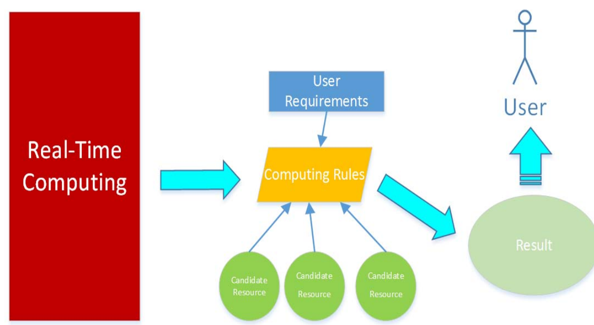

Fig. 2. User Matching Model

D. Traditional FCM algorithm

FCM algorithm divides L-dimension objects into fuzzy sets. N represents the number of objects, and C represents the number of sets. Minimize the objective function to cluster data by the iterative update of objects membership and cluster centers. The objective function is as follows :

$$J(U,V) = \sum_{i=1}^{N} \sum_{c=1}^{C} u_{ic}^{m} d^2(x_i, v_c) \tag{2}$$

Membership degree u_{ic} meets the constraint condition：

$$\sum_{c=1}^{C} u_{ic} = 1, \forall i \tag{3}$$

So the calculating formulas of membership degree and cluster center are as follows:

$$u_{ic} = \frac{1}{\sum_{c=1}^{C} \left(\frac{d(x_i,v_c)}{d(x_i,v_c)} \right)^{m\text{-}1}} \tag{4}$$

$$v_c = \frac{\sum_{i=1}^{N} u_{ic}^{m} x_i}{\sum_{i=1}^{N} u_{ic}^{m}} \tag{5}$$

III. IGAFCM Algorithm Improvement

IGAFCM Algorithm mainly contains three parts: 1) Resource Clustering. A large number of resources are decomposed, which reduces screening data volume. 2) Resource Scheduling. This step is to choose the candidate Top

-N results by algorithm of fast searching. 3) Score Matching. Considering the actual user demands and resource performance, the dispatcher chooses the best results from the candidate queue and return to the user.

A. Resource Clustering

MIN-MIN algorithm [9] completes the task by assigning the minimum priority task to the resources which take the minimum time, but in the cloud resource scheduling, it costs higher because of global solution. On the contrary, IGAFCM algorithm is used to classify the resources, and the number of requirements to solve the problem is greatly reduced. Moreover, IGAFCM can re-cluster resources based on the dynamic changes, so that the clustering results can be maintained in dynamic update. Implementation steps are as follows:

First, manually set the first clustering classification number C, stall threshold E and weighted coefficient M. Although the unsupervised FCM can automatically generate the above parameters, clustering results are often unsatisfactory. Thus, semi supervised FCM results combined with the artificial setting is the best choice.

In the process of setting the classification number C, the target of resource clustering is to reduce the number of related resources, so the value should be as far as possible to meet the actual demands. Such as clustering 100 resources into four categories ideally, and each average number is 25. Then this data scale is relatively modest.

Next, for the sake of making the fast convergence of clustering function, the genetic algorithm is used for fast global search, which generates the initial cluster centers, and then FCM clustering (GAFCM) is performed according to the initial cluster centers. Experiments show that the GAFCM algorithm has the characteristics of fast convergence, and it can improve the clustering efficiency.

At last, for changing resources, it mainly divided into the following two aspects to discuss.

(1) The same number of resources, and resource property changes.

When some resources change, re-calculate the membership degree $U_{i,j}$ in their categories. Owing to the FCM algorithm, the calculation of clustering center C_i depends on the membership degree, whose changes will reaction in the cluster center. When variation of cluster centers exceeds the threshold range, re-cluster can be carried out. At this moment, due to the unchanged number of resources, the number of clusters should not be changed. At the same time, the central value of the last clustering result is the initial center of the new process, which improves the clustering efficiency greatly.

$$u'_i = \frac{d(x_i, v_c)^m}{d(x'_i, v_c)^m} \quad (6)$$

x_i' represents the points after changing in the cluster, and x_i represents the points before changing in the cluster.

$$\overline{V_c} = \frac{\sum_{i=1}^{N} u_{ic}^{m} x_i u'_i}{\sum_{i=1}^{N} u_{ic}^{m}} \quad (7)$$

Take advantage of $\|\overline{V_C} - V_C\| > \omega$, and ω is the offset value to arrange manually.When detected large deviations, the reunion methods will be triggered.

(2) Resource number has changed.

Changes of resource number will directly affect the clustering results. Only when the decimal point changes, a certain class can be adjusted. Such as adding points or remove the class directly, while updating membership degree and cluster centers C_i in the class, and reallocate resources in the next cluster. When the number of resources occur large-scale changes, the number of classification needs to be considered during the re-clustering. By measuring the ratio of current number scale to previous one, reset the number of classes, and use the last cluster center as the new cluster center to speed up the efficiency of FCM clustering.

Here is the pseudo code.

TABLE I. IGAFCM ALGORITHM

Algorithm 1 IGAFCM cluster algorithm

```
IGAFCM(U,K,ε,m)
// U : matrix of user data
// K : the total number of clustering groups
// ε : threshold of clustering
// m : weighted coefficient
Using GA algorithm, compute the global optimal classification center C
GA(U,K)→initCenter[C]
D = calcDistance(U ,C)
A = calcA(m,D);
error = 1;
while error > ε do
   A′ =A;
   G = calcGroupCenter(U ,K );
      D = calcDistance(U ,C);
      A = calcProbabilityMatrix (m,D);
  error = // A − A′ // ;
end while
for k=1toK do
   for i=1 to M do
   if a_ki ≥ θ then
        u_i →FP[k];
     end if
   end for
end for

Situation 1 Uc,j change , number of sourcedata not change:
using the before clustering center for fast clustering,
FCM(ResultCenter[C],newsouredata) → ClusterResult,ResultCenter[C]

Situation 2 Uc,j change, number of sourcedata change
Caculate the new center of change resource C[n]
FCM(C[n],newsouredata) → ClusterResult,ResultCenter[C]
```

Through the above steps, IGAFCM algorithm only needs to specify the parameters of the first clustering scheme, and the following clustering can automatically set the parameters according to past results. The experiment shows that the algorithm has better dynamic adaptability.

B. Resource Scheduling

Resource scheduling means for scheduling needs to find a suitable category, while matching category resources and demands. Pick Top-N values of higher conventional matching degree as the candidate points, which are calculated in the next step with the user attributes, and then the highest score will be returned to the user as the scheduling results. Scoring rules for the algorithm is an adjustable part, and the easiest matching is weighted matching [8]. For the attributes that users are concerned about, giving them higher weights makes the results come in line with the user demands. Here is the formula of simple weighted matching score.

$$Score = \frac{\sum \|req_i\text{-}pro_i\| m_i}{\sum m_i} \tag{8}$$

req_i represents property $\boldsymbol{i}$ of user demands, and pro_i represents the resource property $\boldsymbol{i}$ and m_i represents the weight of property $\boldsymbol{i}$.

C. Score Matching

Preliminary results obtained by the previous step can get a candidate resources queue to meet the user demands. However, in the mobile cloud computing, resource is in a state of dynamic change, and it is necessary to calculate the candidate results in real time. Calculations can be distributed to independent resource servers, and parallel computing will return the fastest response resource as the final result to users.

IV. Experiments Analysis

This paper uses MATLAB simulation test for IGAFCM application in the mobile computing process of resource allocation. Resource node attributes in the experiment are divided into three categories: CPU, memory and bandwidth.

A. Comparison of IGAFCM Algorithm and MIN-MIN Algorithm

In order to verify the effectiveness of IGAFCM algorithm in the resources allocation, this paper compares and tests the improved algorithm with MIN-MIN algorithm firstly. The internal mechanism of the algorithm is to quickly find relevant taxonomic needs by clustering, thereby reducing the number of values to solve. Experimental results are shown in Fig. 3.

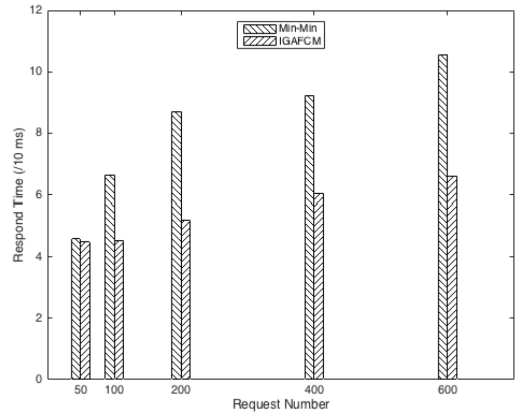

Fig. 3. The algorithm efficiency for IGAFCM and Min-Min

When there are 50, 100, 200, 400, 600 user requests, the chart shows that the comparison of the solution efficiency with IGAFCM algorithm and MIN-MIN algorithm. Because the size of the resource sample is 400, it is divided into 5 categories. It is proved that the IGAFCM algorithm solution is random, so in some cases the efficiency change of 50 or 100 requirement solution is not too large. Nevertheless, in accordance with the MIN-MIN algorithm, with the increase of demand, the time required will increase. The experimental results demonstrate that the IGAFCM algorithm can increase the demand hit efficiency in the scheduling process.

B. Semi-supervised type of IGAFCM algorithm

To solve the problem of demand change which caused changes in the classification, this paper puts forward the improved algorithm of semi-supervised model. By analyzing the results of changes in a small or large number of data points, this method has good adaptability and maneuverability.

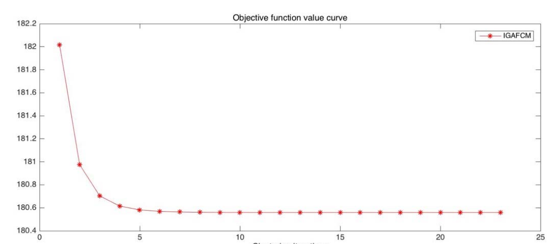

Fig. 4. Objective function value curve of IGAFCM

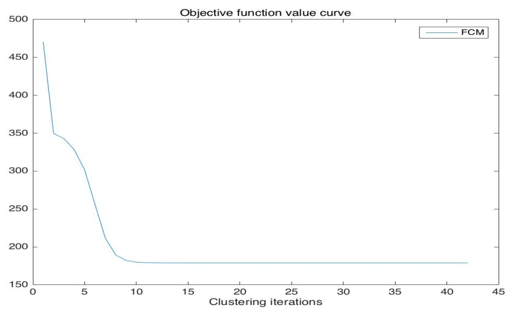

Fig. 5. Objective function value curve of FCM

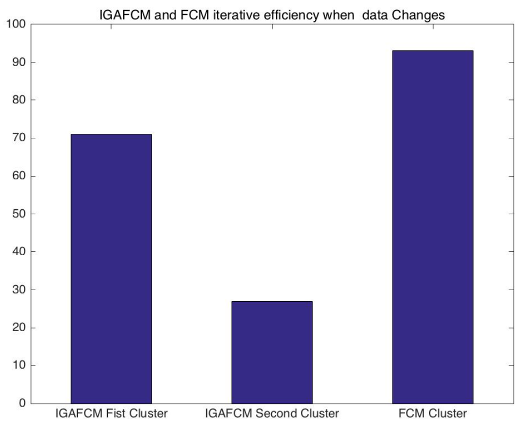

Fig. 6. IGAFCM and FCM iterative efficiency when data Change

The above experiment result shows that the IGAFCM algorithm has good convergence. When resources need to be re-clustered, the IGAFCM algorithm can accelerate the iteration speed. Moreover, compared with traditional FCM algorithm, the result of a single time IGAFCM re-clustering has faster iteration speed. This is because the genetic algorithm can quickly find the approximate cluster center for global search to avoid the possible formation of local optimal solution which reduces efficiency situation of traditional FCM algorithm.

C. Customer satisfaction measurement

In view of the user demand analysis, IGAFCM algorithm can guarantee the efficiency and user satisfaction. Formula of measuring customer satisfaction is as follows:

$$\overline{U_{score}} = \ln\left(\sum {}^{\rho\left(t_{request}\right)}\Big/_{r_{csource}}\right) \qquad (9)$$

$\rho\left(t_{request}\right)$ represents the attribute value of user request, and $r_{csource}$ represents attribute value of certain resource.

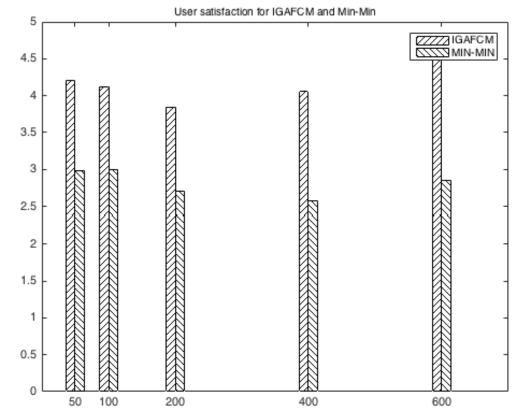

Fig. 7. User satisfaction of IGAFCM and Min-Min

V. CONCLUSION

In this paper, an improved FCM (IGAFCM) algorithm is proposed in the process of mobile resource scheduling. IGAFCM algorithm can reduce the scale of resource scheduling and screen candidate results in real time by customized rating mechanism to provide users with satisfied results. Besides, the IGAFCM algorithm have a certain dynamic resource monitoring capabilities. Under normal circumstances, it can re-clustering quickly to maintain a good operating efficiency and ensure the good user satisfaction.

REFERENCES

[1] Dinh H T, Lee C, Niyato D, et al. A survey of mobile cloud computing: architecture, applications, and approaches[J]. Wireless communications and mobile computing, 2013, 13(18): 1587-1611

[2] Li W J, Zhang Q F, Ping L D, et al. Cloud scheduling algorithm based on fuzzy clustering[J]. Journal of China Institute of Communications, 2012, 33(3): 146-154.

[3] GUO F, YU L, TIAN S, et al. Workflow task scheduling algorithm based on resource clustering in cloud computing environment[J]. Journal of Computer Applications, 2013, 8: 020.

[4] Bezdek J C, Ehrlich R, Full W. FCM: The fuzzy c-means clustering algorithm[J]. Computers & Geosciences, 1984, 10(2-3): 191-203.

[5] Li J F, Peng J. Task scheduling algorithm based on improved genetic algorithm in cloud computing environment[J]. Jisuanji Yingyong/ Journal of Computer Applications, 2011, 31(1): 184-186.

[6] Zhou B, Dastjerdi A V, Calheiros R N, et al. A context sensitive offloading scheme for mobile cloud computing service[C]//Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on. IEEE, 2015: 869-876.

[7] Tang J, Zhang G, Wang Y, et al. A hybrid approach to integrate fuzzy C-means based imputation method with genetic algorithm for missing traffic volume data estimation[J]. Transportation Research Part C: Emerging Technologies, 2015, 51: 29-40.

[8] Lee B G, Park J H, Pu C C, et al. Mobile-based kernel-fuzzy-c-means-wavelet for driver fatigue prediction with cloud computing[C]//SENSORS, 2014 IEEE. IEEE, 2014: 1236-1239.

[9] Etminani K, Naghibzadeh M. A min-min max-min selective algorihtm for grid task scheduling[C]//Internet, 2007. ICI 2007. 3rd IEEE/IFIP International Conference in Central Asia on. IEEE, 2007: 1-7.