Available online at www.sciencedirect.com

SciVerse ScienceDirect

AASRI Procedia 1 (2012) 92 – 99

AASRI Procedia

www.elsevier.com/locate/procedia

2012 AASRI Conference on Computational Intelligence and Bioinformatics 2012

Research and Design on Fuzzy-based Cluster Model

Wang Wei[1] , Xu Lihong , Zhou Zhangjun

Tongji University , Shanghai , 200093 , China

Abstract

The task likely fail for the mismatching between the computer's ability and the consume of the task which have been assigned for the computer, along with the sum of the computers in the cluster becoming larger and larger, even though the Internet can share the resources, yet the computers differ greatly. And then it is a tough problem how to apart and management them. The paper gives a way to solve it, whose main purpose is to separate the computers into different groups in which the computers are the most similar by the calculating following the Fuzzy theory. The value of the weighted value is modifiable, which following the characters of the computer can affect the Fuzzy set as you want.

Keywords: Cluster , Fuzzy;

1.The raise of the problem

In reality, most of time, machine is idle, resulting in a significant waste of computing power. When using the Task Manager under Windows or other tools under Linux platforms (such as top, xload) to observe the CPU, we will find the common utilization of the CPU is just among 1 to 2 percent. In fact, if there are more computers, the waste will be intensified. In a department that includes 300 computers, the idle rate of CPU is amazing. However, these sectors still need powerful servers to be used to compile or simulation. Sometimes this situation will be exacerbated, because with the increase of users, more than one even if the 8 CPU servers, nor can the full load duty of landing to another free server because users rarely change their habits to log in another server. If you can take advantage of the existing computing resources and idle CPU utilization, or allow the migration of load on the server smartly, it will be a very happy thing [1] [2].

We are now in the golden stage of the cluster development, generating some different influential cluster

[1] *Corresponding author. Tel.: 13918359871; fax: +420 021 55069192.
E-mail address:wangweiwangwei3@hotmail.com

doi:10.1016/j.aasri.2012.06.017

model, some corresponding to the specific tasks, and some have very good scalability, and some good load balancing. This will gradually form a wholesome cluster standard, and how to divide the cluster which can enable the similar computer based on some features work together, to make better management of the cluster and balance of the load be more scientific, which means to find a way to make cluster divided in a scientific method in accordance with some properties of the computer and characteristics of the computing tasks

In recent years, grid [4] technology with the compatibility of heterogeneous resources become the research focus of the industry, providing a new theoretical and technical support for us to build heterogeneous clusters. The computer's CPU, memory, disk, network parameters vary on INTRANET, so the system formed by connecting them together simply is not a cluster. They must be Clustered and divided into logical computer clusters of different calculation types, and then allocate computing tasks. In order to effectively divide clusters, we proposed the study and implementation of fuzzy -based cluster model (Fuzzy-based Cluster Model) which have practical significance in integration of existing idle computing resources to be a computing device.

2.Mathematical model

2.1.Computer parameters

We only discuss the most common five parameters of the cluster computers, they are: ①CPU: computing speed. (Some use MIPS to measure and some with MHZ.It's OK as long as the dimension unified), ②MEMORY, ③Net Adapter: NIC, ④I / O: the hard disk read and write speeds, ⑤NET: network bandwidth. Of course, there are other very important parameters, the model is not sensitive about the number of parameters by the same treatment. Similarly, the dimension of parameters in this model is not sensitive as long as the dimension of the same parameters can be unified. The impact of the dimensionless can be eliminated through the standardization of data in clustering analysis.

2.2.Parameter weights

The distribution of parameter weights of the computer is different when clustering in order to divide the clusters for classification according to the final calculation of task characteristics(①CPU:K_1,②MEMORY:K_2,③Net Adapter:K_3,④I/O:K_4,⑤NET:K_5, $K_1+K_2+K_3+K_4+K_5=1$.Weights correspond to the number of attribute parameters,when the attribute parameter increases and decreases, the number of weights matched will increase and decrease,but the sum should remain unchanged.

The parallel and serial attribute of computing tasks essentially determines the application efficiency of the cluster. The operating efficiency of each node is the computing essence of the node. Therefore, when fuzzy clustering, the parameters of one or several key attributes corresponding to a task is very important given relatively large weights, so that the impact of the attribute parameters with large weights for similarity is relatively large, the division of cluster can be based on the characteristics of the task, and adapt to the task.

In actual operation, the subjectivity is relatively large; If the weights through a mathematical proof in the strict sense which can be achieved, we need detailed mathematical analysis on the basis of computing tasks(beyond the scope of this article).

2.3.Fuzzy Clustering Analysis

Cluster computer is generally divided into three steps:1 data Standardization,2 establish fuzzy similar matrix,3 clustering. We will discuss in detail in the following .

2.3.1 Data Standardization

Establish a data matrix and remove the dimensional data matrix, and standardized data.

2.3.1 (1) establish a data matrix

Suppose Cluster computers are classified based on the domain U = {X_1, X_2, ... Xn}.The performance of each computer by m attributes parameter(in this paper m = 5), X_i = (x_1, x_2, ... x_m) (i = 1,2, ... n) (in this paper①x_1: CPU ② x_2:MEMORY, ③ x_3: Net Adapter,④x_4:I / O,⑤x_5:NET),so, we get the raw data matrix is:

$$\begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix}$$

2.3.1 (2) standardization of data

Five attribute parameters of computers in clusters have different dimensions.in order to compare different dimensions, you need to do appropriate transformation. Different dimensions according to the requirements of fuzzy matrix to make the data compression to interval [0,1]. This paper discusses only two kinds of transformation (translation standard deviation of the transform, pan range transformation), and more transformations, you can refer to[3].

2.3.1 (2.a) pan - standard deviation of transformation

$$x'_{ik} = \frac{x_{ik} - \overline{x_k}}{s_k} (i = 1, 2, ..., n; k = 1, 2, ..., m)$$

where

$$\overline{x_k} = \frac{1}{n}\sum_{i=1}^{n} x_{ik}, s_k = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ik} - \overline{x_k})^2}$$

In this way, after panning standard differential transform, the mean of each attribute parameter to 0, the standard deviation of 1, and elimination of the influence of the dimensions. The obtained x ik is not necessarily in the interval [0,1], but the classification of the cluster computer is still effective, and the result of classification is still good.

2.3.1 (2.b) translation – the transformation

$$x''_{ik} = \frac{x'_{ik} - \min_{1 \le i \le n}\{x'_{ik}\}}{\max_{1 \le i \le n}\{x'_{ik}\} - \min_{1 \le i \le n}\{x'_{ik}\}}$$

$$(k = 1, 2, ..., m)$$

Obviously, $0 \le x_{ik} \le 1$ and also it can eliminate dimensional influence.

2.3.2 establish fuzzy similar matrix

To classify cluster computer is the domain of U = {X_1, X_2, ... Xn}, each computer is described as X_i = (x_1, x_2, ... x_m) (i = 1,2, ... n) (in this paper m=5). Computers with a similar nature will be classified as a class and similarity S (X_i, X_j) will be Calculated according to a method or principle between any two computer X_i, X_j, denoted by r_{ij}, obviously r_{ii} = 1, and $0 \le r_{ij} = r_{ji} \le 1$. At this time there will be a fuzzy similarity matrix in the domain of U.

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots \\ r_{n1} & \cdots & r_{nn} \end{pmatrix}$$

$$r_{ii} = 1, 0 \le r_{ij} = r_{ji} \le 1, 1 \le i, j \le n$$

There are many ways of Solving fuzzy similar matrix, this paper describes two methods, for more, you can refer to [5].

2.3.2 (a) The arithmetic average and take a small law

$$r_{ij} = \frac{\sum_{k=1}^{m} K_k(x_{ik} \wedge x_{jk})}{\frac{1}{2}\sum_{k=1}^{m} K_k(x_{ik} + x_{jk})}$$

$1 \le i, j \le n$

Take r_{ij} as similarity coefficient of X_i and X_j, this method of $x_{ij} > 0$ (This paper applies) ($1 \le i \le n$, $1 \le k \le m$).

2.3.2 (2) the absolute value of index

$$r_{ij} = e^{-\sum_{k=1}^{m} K_k |x_{ik} - x_{jk}|}$$

There are no restrictions on the use of this method.

2.3.3 Clustering

With the fuzzy similarity matrix (3.3.2) is not enough, it is because there is no transitivity, such as the similarity between computer is greater than or equal to α counted as a class, then when $r_{ij} \ge \alpha$ and $r_{jk} \ge \alpha$ may not be there $r_{ik} \ge \alpha$, that is, when X_i, X_j similar and X_j, X_k similar, X_i and X_k are not necessarily similar. so, calculating the equivalent of closure R(denoted as R*) on the basis of (2.3.2)will be useful. Solving the Transitive Closure based on the fuzzy similarity matrix clustering is just one clustering method, and other more clustering methods can be found in [3].

$$R^* = \begin{pmatrix} r_{11}^* & \cdots & r_{1n}^* \\ \vdots & \ddots & \vdots \\ r_{n1}^* & \cdots & r_{nn}^* \end{pmatrix}$$

$$r_{ii}^* = 1, 0 \le r_{ij}^* = r_{ji}^* \le 1, 1 \le i, j \le n$$

2.3.3 (1) closure clustering method

Do cycle of R as the following:

$$r_{ij}^{'} = t_{ij} = \bigvee_{k=1}^{n} (r_{ik} \wedge r_{jk})$$

$$= \max_{1 \le k \le n} \{r_{ik} \wedge r_{jk}\}$$

Until rij remain unchanged, at this time rij = rij*, ie R = R*.

The fuzzy relation matrix R based on five (or more) property parameters between computers is often a fuzzy similarity matrix, and not necessarily a fuzzy equivalent matrix. Therefore, computers are classified by the equivalent matrix, and we usually take transitive closure method to construct the transitive closure R* = t(R), and then clustering on the basis of the R* (this method). After a series of non-identity transform in the process of construct R *, the inevitable increase of some factors that do not belong to R results in the clustering distortion. The more times of the non-identity transform, the greater the distortion of likelihood .In this approach, in order to pursue the transformation, we use R* instead of R to do clustering. Although the principle is not perfect, the method is simple, and tests confirmed the classification results are better, so this method can be used in practice.

2.3.3 (2) the determination of the optimal threshold

In fuzzy clustering analysis, different α ($0 \le \alpha \le 1$) can be a different classification, to form a dynamic clustering map giving a comprehensive understanding of computers in a cluster which is relatively intuitive and visual. However, when clustering according to the characteristics of the task (beyond the scope of this article), we want to determine a threshold based on the characteristics of specific tasks. The similarity in the computer within this threshold can facilitate management or regard as same node in the distribution of

computing tasks that can be relatively easy to achieve load balancing and other issues. It needs detailed mathematical analysis in computing tasks to get the corresponding threshold.

We use F statistic to determine the value of α which can adapt to a general computing tasks, and also a good method to determine α. The larger the F value, the greater the difference between classes. the greater the distance between classes, the better the classification. for more discussion, refer to[3].

3.The validity of the model to prove

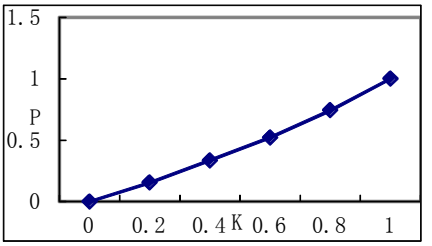
3.1.The validity of the weights to prove

In order to facilitate the discussion of the weight Ki (in this paper i=5) for the clustering results, we introduce the variable P. The definition of P as follows:

$$P_i = \frac{k_i(x_{ik} \wedge x_{jk})}{\frac{1}{2}\sum_{k=1}^{m} k_k(x_{ik} + x_{jk})} \over r_{ij}$$

$$\Rightarrow P_i = \frac{k_i(x_{ik} \wedge x_{jk})}{\sum_{k=1}^{m} k_k(x_{ik} \wedge x_{jk})}$$

According to the definition of P, Ki-Pi image is as follows (assuming other weights are evenly reduced;the experimental data from No.2 and No.16 sample computers).



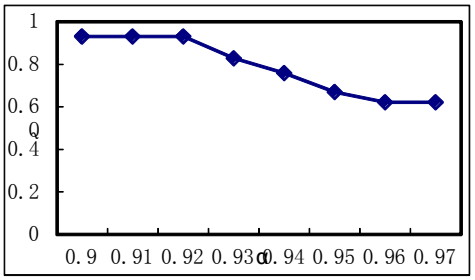
It is not difficult to come from the definition formula, P denotes the percentage of K_i (weights) corresponding to the attributes (X_i) in the two computer similarity (R_{ij}) (as opposed to other attributes). K_i (weights) corresponding to the contribution rate (percentage) of the attribute (X_i) to similarity (R_{ij}).when K_i (weights) increases, the P value increases, X_i (attributes) for contribution to the similarity increases (for the sake of discussion, assume that the weights of other attributes reduce by an average.) Clustering similarity (R_{ij}) is the clustering standard, so the impact of the X_i (attributes) is great. To the contrary, K_i (weights) decreases, ie, P value decreases, the clustering results also by X_i attribute impact the smaller. In summary, P denotes the contribution rate of X_i (attribute) to similarity (R_{ij}) so that to affect the clustering results, so we can use P values to denote X_i (property) for the clustering effect. K_i effectively affect the clustering results, and the clustering result is the same monotonous with K_i. By adjusting the value of K_i to make the clustering more inclined to a few properties that are most concerned about. For instance, when clustering is more concerned about the CPU and memory, the weight corresponds to them increase, while the other weights decrease. when the others are reduced to 0, it is equivalent to no this property. when a weights of these properties is equal to 1, it indicates that ignore the other properties only according to the selected of these properties to cluster.

Proof of 3.2 clustering threshold clustering

In order to facilitate the discussion of threshold for the clustering results, we introduce the variable Q, Q-definition is as follows:

$$Q = 1 - \frac{n-1}{N-1}$$

(n: the number of clusters at the end; the initial total number of the computers.) Computer experiments based on 30 samples; Q-alpha image is as follows:



(Where $k_1 = k_2 = k_3 = k_4 = k_5 = 0.2$, if the similarity is too small, it does not make sense, so the coordinate origin moved to 0.9, the specific move to where depending on the application, here according to the values of the last 30 sample computers).

It can be drawn from the definition formula that Q represents the degree of the final clustering results. when α increases, the clustering degree of (Q) value decreases; when α decreases, the value of the clustering degree of (Q) increases. Threshold (α) is equivalent to the harsh requirements for clustering, When the threshold is increased, it indicates that the harsh degree of clustering. The higher the similarity requirements, the lower the degree of clustering.

Because the computer can reach a higher threshold number of smaller relative to the lower threshold of the number of computers, naturally, high threshold will form a larger cluster (low cluster extent), but the degree of similarity within each cluster is higher.

4.Model algorithm

The following is clustering results according to different weights and different clustering threshold obtained by 30 sample computers at the author's work place. The classification results are denoted using the computer's serial number.

	CPU(G)	MEM(G)	NA(M/S)	I/O(M/S)	NET(M/S)
1	2.4	1	100	300	100
2-3	1.6	1	100	300	100
4	2.2	1	100	300	100
5-6	4.2	1	100	300	100
7	1.6	0.5	100	300	100
8-9	1.8	0.5	100	300	100
10	4.2	1	100	300	100
11-12	7.2	3.6	100	300	100
13-14	4.4	1	100	300	100
15	4.4	0.5	1000	300	100
16-18	4.6	1	1000	150	100
19	1.6	0.5	100	150	100
20	4.2	1	1000	150	100
21	4	0.25	100	150	100
22-23	3	0.25	100	150	100
24	2.8	0.5	100	150	100
25	4.6	1	100	150	100

26-27	3	0.5	100	150	100
28	2.8	1	100	150	100
29-30	3	0.5	100	150	100

4.1.Different weights(α = 0.96)

(1) k1=0.8,k2=k3=k4=k5=0.05
(2) k1=0.6,k2=k3=k4=k5=0.1
(3) k1=0.4,k2=k3=k4=k5=0.15
(4) k1=0.2,k2=k3=k4=k5=0.2
(5) k1=0,k2=k3=k4=k5=0.25

Group	(1)	(2)	(3)	(4)	(5)
1	1,4	1,4	{1-4}{7-9}	{1-6}{10,13}{14}	{1-6}{10,13}{14}
2	{2,3}{7-9}{19}	{2,3}{7-9}{19}	{5,6}{10,13}{14,25}	7-9	7-9
3	{5,6}{10,13}{14,21}{25}	{5,6}{10,13}{14,25}	11,12	11,12	11,12
4	11,12	11,12	{16-18}{20}	{16-18}{20}	{16-18}{20}
5	{15-18}{20}	{15-18}{20}	{22-24}{26-30}	{19}{21-24}{26,27}{29,30}	{19}{21-24}{26,27}{29,30}
6	{22-24}{26-30}	{22-24}{26-30}			25,28
self-group		21	15,19,21	15,25,28	15

(Computer using the serial number, self-group: the computer serial number to be a group alone)According to the classification results, we can easily find, with the weight of CPU gradually reduce, degree of CPU concern is getting smaller and smaller. for example, in the first category, when the CPU's weight 0.8, only 1,4, when it reduced to 0.4, 2,3,7,8,9 is added in. It can be seen even if there are some differences on the CPU, but considering the other factors, it is quite similar. When it reduce to 0, without considering the CPU, the addition to the number of the computer which other aspects are similar will be more, and CPU is somewhat similar, but the other four areas are not very similar is excluded. So that by adjusting the weights will be able to better control the properties of interest.

4.2.Different threshold (k1=k2=k3=k4=k5=0.2)

α, Group	0.9-0.92	0.93	0.94	0.95	0.96-0.97
1	{1-10}{13,14}{19}{21-30}	{1-10}{13,14}	{1-6}{10,13}{14}	1-4	1-4
2	11,12	11,12	7-9	{5,6}{10,13}{14}	{5,6}{10,13}{14}
3	{15-18}{20}	{16-18}{20}	11,12	7-9	7-9
4		{19}{21-24}{26,27}{29,30}	{16-18}{20}	11,12	11,12
5		25,28	{19}{21-24}{26,27}{29,30}	{16-18}{20}	{16-18}{20}
6				{21-24}{26,27}{29,30}	{22,23}{26}
7					{24,27}{29,30}
self-group		15	{15,25}{28}	{15,19}{25,28}	{15,19}{21,25}{28}

(Computer using the serial number, self-group: the computer serial number to be a group alone)It can be seen that the cluster will be increased when clustering threshold increases, and the number of self-group is also increased, So that the characteristics of each computer in the cluster will be even more distinctive, and the internal similarity of each cluster will be more closer, the granularity of the cluster smaller that forms very similar cluster. Small clusters formed with a larger threshold can not be separated when clustering with a smaller threshold. experiments verify the theoretical results.

The experiments show that the weight value of the property can well control the concerned clustering properties, and threshold choice can also adjust the degree of clustering.

5.Conclusion

Currently how to divide the cluster is a difficult subject; this paper gives a method of division, and the experiments prove that the computers in the cluster can be effectively divided by the method.

There are a lot of research space for the division according to the specific tasks. how to decompose the task, how to make the task be quantified and can be measured by the computer's parameters, so that the division based on tasks has the theoretical basis and experimental validation leading the future research directions.

References

[1] A. Rajkumar Buyya. High Performance Cluster Computing: Architectures and Systems, Volume1,2.Zheng Weimin, Shi Wei, Wang Dongsheng. Beijing: Electronic Industry Press, 2002.
[2] Andrew S. Tanenbaum. Structured Computer Organization, Fifth Edition. Liu Weidong, Song Jiaxing, Xu Ke. Beijing: People's Posts & Telecom Press, 2006.
[3] Wang Guojun. Computational Intelligence (Volume II) - the word computing and fuzzy sets. Beijing: Higher Education Press, 2006.
[4] Chen Qingkui,Na Lichun.Data parallel computational grid model based dynamic redundancy. Journal on Communications.2005,12.
[5] Chen Qingkui,Na Lichun.Communication Model for Grid Based on Clusters, Computer Engineering and Application, 2006,42 (27).
[6] Zhou Bing,Feng Zhonghui, Wang Hexing.the Parallel Clustering Algorithm of the Cluster Environment. Computer Science, 2007,10 (15).
[7] Xiasheng Ping,Lv Xiajun,Liu Jianjun. Based Parallel Distributed Clustering and its Application. Zhengzhou University (Natural Science Edition), 2006,4 (23).