

Based on PSO cloud computing server points location searching

Chen Hua-sha

Information Technology Center
Shanghai International Studies University
Shanghai, china, 13916415603
E-mail: huasha@shisu.edu.cn

Shen jiajie

Computer center of School of Information Science &
Technology
East china normal university
Shanghai, china, 18817519704
E-mail: 51111211010@ecnu.cn

Abstract: Aiming to problem of how to optimize the cloud computing location, an improved PSO algorithm is designed to handle this problem, using the improved PSO algorithm servers location and communication cost can be optimized under cloud computing situation. Though theoretical derivation, the correctness of improved PSO algorithm is proofed. The correctness of theoretical derivation is also verified by the experiment.

Keyword: cloud computing, particle swarm optimization, communication cost

I Introduction

With the development of the cloud computing, and more and more application is upload to the cloud side, however the how to locate the cloud computing server node remain a question to debate.

In the paper, an improved PSO algorithm is presented to handle this problem. Though the theoretical deviation, the correctness of the algorithm is proofed. The correctness of the theoretical deviation is verified by the experiment.

The paper is organized as follow: In second section of paper, the related work will introduced. In the third section of paper improved PSO algorithm will be presented. The experiment and its result will appear in forth section. The fifth section is the result.

II related work

2.1 cloud computing

Cloud computing^[1] is hot topic both research and application development. Many paper discuss about this topic^{[2]-[12]}, for example some paper are more focus to the definition of the cloud computing other are application establish and the model. It is true that the cloud computing will become a computer organize form of next generation. The cloud computing also cost vary problem, for example how to find the resource in optimization cost

and how to make the friendly inference the in the cloud situation.

2.2 particle swarm optimization

Standard PSO algorithm is first presented by Kennedy, Eberhart in 1995^[13]. After standard PSO algorithm, there are a lot of improvement, and several application and case have already been solved by the PSO algorithm.

2.2.1 the function of the standard PSO algorithm

The position and velocity of swarm as follow:

$$\begin{cases} X_i = (x_{i1}, x_{i2}, \dots, x_{id}) \\ V_i = (v_{i1}, v_{i2}, \dots, v_{id}) \end{cases} (1 \leq i \leq n) \quad (1)$$

The optimization value of the particle i is marked as:

$$p_i = (x_{i1}, x_{i2}, \dots, x_{in}) \quad (2)$$

The optimization value of the swarm is marked as:

$$p_{gi} = (x_{gi1}, x_{gi2}, \dots, x_{gin}) \quad (3)$$

Each particle will change position and velocity the every step based on the follow function:

$$\begin{aligned} v_{id}^{k+1} &= \omega v_{id}^k + c_1 \xi (p_{id}^k - x_{id}^k) + c_2 \eta (p_{gd}^k - x_{id}^k) \\ x_{id}^{k+1} &= x_{id}^k + v_{id}^{k+1} \end{aligned} \quad (4)$$

2.2.2 The description of the standard PSO algorithm

The step of standard PSO algorithm as follow:

Step 1: initialize the swarm include the position and velocity

Step 2: using the PSO algorithm to find the optimization value.

Step 2.1: find the optimization value of each particle(p_i) and optimization value of swarm(p_{ig}).

Step 2.2: update the position and velocity based on function 4.

Step 2.3: calculate the target function value of each particle.

Step 2.4: if step number bigger than maximum step number or the optimization value is reached go to step 3, else go to step 2.1.

Step 3: Print the optimization value

The improved PSO algorithm have been presented by vary papers^{[14]-[25]}, for example how to combine the PSO algorithm with SVM algorithm with optimization parameters and how to use different kind of improved PSO algorithm to solve vary problems.

III Improved PSO algorithm and algorithm's characters

3.1 The assumption and definition of PSO algorithm

Assumption 1: The communication cost is increase with the distance between the nodes.

Assumption 2: One cloud server only can locate in several different place which is close to each other.

Assumption 3: The center data base can collect all agents' data after communicated with all agents, and only after it communicate with all agents.

Assumption 4: One city can have several different server nodes.

Definition 1: the number of server which communicate with center base each city node is called as the communication number.

Definition 2: the city server node which is closest to the center data base is called as closest distance between city i and particle k marked as

$$sn_{ijk} = \arg \min(\|n_{ij} - p_k\|^2) \quad (5)$$

n_{ij} is position of city i node j. p_k is particle k.

Definition 3 the variable range of the particle i is called as the particle range i. marked as RP_i .

Note: the particle range should include all the server node of one city.

Definition 4: the distance between the city i server j and the center data base is called as the communication length between the city i server j and the center data base k, marked as:

$$dc_{ijk} = \sqrt{\|n_{ij} - sd_k\|^2} \quad (6)$$

n_{ij} is the city i server j, sd_k the center data base k.

The minimum the communication length between the city i server j and the center data base is called the minimum communication length between city i and data base k, marked as

$$mdc_{ik} = \min_{j=1}^n \sqrt{\|n_{ij} - sd_k\|^2} \quad (7)$$

n_{ij} is the city i server j, sd_k the center data base k.

Definition 5: the total communication length is called as the total communication length, marked as

$$TD_{id} = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l dc_{ijk} \quad (8)$$

TD_{id} is the total distance, dc_{ijk} is the communication length between the city i server j and the center data base k.

The minimum value of total communication length is called as the minimum communication length. marked as

$$MTD = \min_{id} TD_{id} \quad (9)$$

3.2 The description of improved PSO algorithm

The description of improved PSO algorithm as follow:

Step 1: initialize the swarm include the position and velocity.

Step 2: using PSO to find the minimum sum distance of the

Step 2.1: find the optimization value of each particle (p_i) and optimization value of swarm (p_{ig}).

Step 2.2: update the position and velocity based on function 4.

Step 2.3: check all particle location, and if the particle i outside of region fix it.

Step 2.4: calculate the target function value of each particle.

Step2.5 if the step number is bigger than maximum step number or the optimization value go to step 3, else go

to step 2.1.

Step2.6:search the point which is inside of side which

Step 3: print the result server node

From step 2.1 to step 2.4 , the process of improvement PSO algorithm is as same as the standard PSO algorithm. The main difference between the improvement PSO

3.3 The character of improved PSO algorithm

Theorem 1: if every selected node is closest to the center data base the total communication will become the minimum communication length.

Proof:

If the total communication minimum communication length, according to the function 9 and function 8:

$$MTD = \min_{id} TD_{id} = \min_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l dc_{ijk}$$

and because all the dc_{ijk} independent with each other,

the minimum value of function equal to:

$$\min_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l dc_{ijk} = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l \min dc_{ijk}$$

So if and only if all the the communication length between the city i server j and the center data base k is minimum, the optimization value will be reached.

Inference 1: if the improvement PSO can find the close point in every city the minimum total communication length will be found.

Proof:

As the theorem 1 if all the minimum communication length between city i and data base k is reached, the minimum total communication length will be found, so if all the minimum communication length is found by PSO algorithm, the total communication length will be minimum total communication length.

Theorem 2: if the particle range is small enough, the total communication length will change a little between different rounds of experiment.

Proof:

If the the particle range is small enough that one city look like one point comparing with the distance between

the citys is very small and look like a point. So the distance different between the different experiment round will be close.

Inference 2: if the particle range is small enough, the total communication length increase with communication number, it is close to direct ratio.

Proof:

According to the theorem 2, if the particle range is small enough that the different between the experiment round will be very small. So if the communication number is n , it is equal to n time experiment of search n closest city agent point to the center data base.

$$\begin{aligned} TD_{id} &= \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l \sum_{w=1}^g dc_{ijkw} \\ &= \sum_{w=1}^g \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^l dc_{ijkw} \\ &= \sum_{w=1}^g TD_{idw} \end{aligned}$$

TD_{idw} is the w round of the total communication

length, dc_{ijkw} is communication length between the city i server j and the center data base k in round w and

$$TD_{id1} \approx \dots \approx TD_{idg}$$

The TD_{idg} is g round total communication length.

So the total length of the communication length is close to the sum of n time of the total length of communication length when communication number is 1.

$$TD_{id} \approx \sum_{w=1}^g TD_{id1} \approx \dots \approx \sum_{w=1}^g TD_{idg}$$

and

$$TD_{id} \approx nTD_{id1} \approx \dots \approx nTD_{idg}$$

So the theorem is proofed.

IV Experiment

4.1 The experiment data

The data material of this experiment is come from <http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/tsp/> ,

and the experiment data is established as follows:

Step1: select several different several data-sets of TSP from the TSP database, and use one point represents a server node of the cloud. The data-set represents one city.

Step2: each node plus a offset which far bigger than the coordinate of dataset's node coordinate, and make sure the same dataset node having same offset.

Step3: make sure all dataset is far away from each other, or go to step2.

Step 4: set the location of center data base.

The selected dataset and offset as follows:

Cloud id	Dataset name	Coordinate offset	The TSP length
1	ei51	(0,2000)	426
2	ei51	(2000,0)	426
3	ei76	(1000,1000)	538
4	ei101	(0,500)	629
5	ei51	(2000,2000)	426
6	ei51	(500,0)	426

As table 1 shows the all city is created by 3 different data sets, and offset is far larger than TSP length.

The coordinate of the center data base is location is (1600, 1200) and (600, 700).

4.2 The experiment result

Figure 1 shows the node of city and center base location

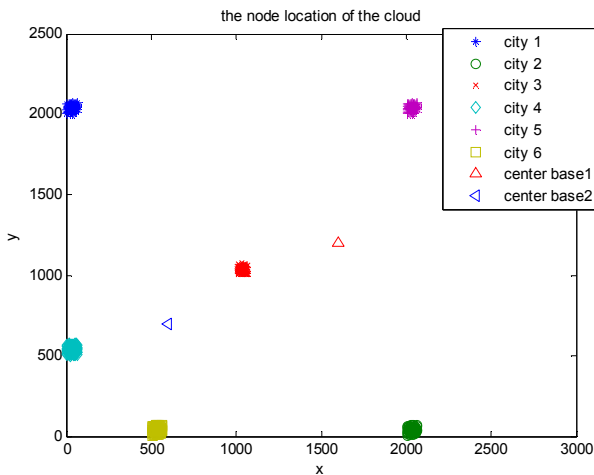


Figure 1 the node location of the cloud

Figure 1 shows the node of the experiment and the two center base. The reason why the city looks like one point is that inside distance of city is far smaller than the distance between the city.

Figure 2 show the improved PSO algorithm result of under situation one server node each city.

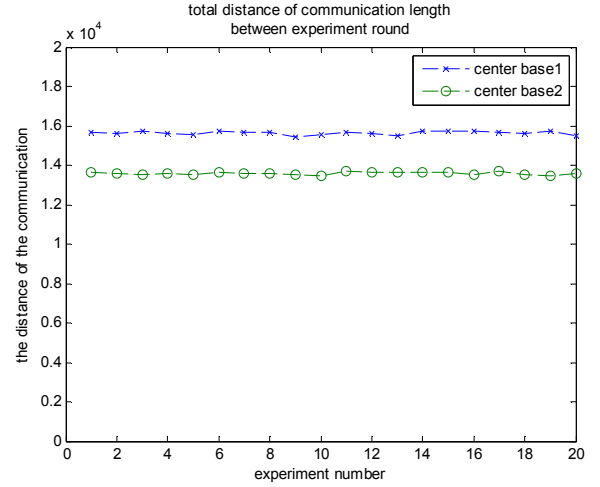


Figure 2 Total distance of communication length between experiment round

The figure 2 shows the total communication length between round numbers. The communication length between the different experiment round is close to each other because the stability of improved PSO algorithm and the region is fit for the experiment which fits theorem 2.

Figure 3 shows the relationship between the communication number and the communication length.

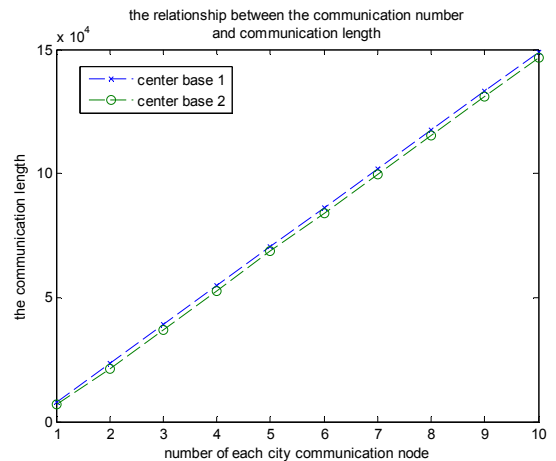


Figure 3 the relationship between the communication number and communication length

Figure 3 show the relationship between communication node number of each city and the total communication. The figure 2 shows the communication length with the increase of the node number of each city, and it is almost like to be direct ratio which is fits with inference2.

V conclusion

The improvement PSO algorithm is presented to handle the problem of cloud computing server points location. Though theoretical derivation, the correctness of the algorithm. The correctness of the theoretical derivation is verified by the experiment.

Reference

- [1] M. Armbrust, A. Fox, R. Griffith. A view of cloud computing. *Communications of the acm*. Vol. 53, iss. 4, april 2010, pp. 50-58
- [2] D. Nurmi, R. Wolski, C. Grzegorzczak. The eucalyptus open-source cloud-computing system. *cluster computing and the grid*, 2009. ccgrid '09. 9th pp. 124-131
- [3] L. youseff, M. Butrico, D. D. Silva. Toward a unified ontology of cloud computing. *Grid computing environments workshop*, 2008. gce '08 pp. 1-10
- [4] qian wang , kui ren , wenjing lou. Privacy-preserving public auditing for data storage security in cloud computing. *infocom*, 2010 proceedings ieee. pp. 1 - 9
- [5] L. Wang, G. Laszewski, A. Younge. Cloud computing: a perspective study. *new generation computing*. Vol. 28, iss. 2 , 2010, pp. 137-146
- [6] M. D. D. Assunção, A. D. Costanzo. Evaluating the cost-benefit of using cloud computing to extend the capacity of clusters. *Hpdcc '09 proceedings of the 18th acm international symposium on high performance distributed computing* pp. 141-150
- [7] K. Kumar, Y. H. Lu. Cloud computing for mobile users: can offloading computation save energy?. *Computer*. Vol. 43 , iss. 4 , 2010, pp. 51 - 56
- [8] n. Sulta. cloud computing for education: a new dawn?. *International journal of information management* 2010. pp. 109 - 116.
- [9] H. Takabi, J. B.d. Joshi, g.-j. Ahn. Security and privacy challenges in cloud computing environments. *Ieee security and privacy*. Vol. 8 , iss. 6, 2010, pp. 24 - 31
- [10] S. Subashini, V. Kavitha. A survey on security issues in service delivery models of cloud computing. *Journal of network and computer applications*. Vol. 34. 2011, pp. 1-1
- [11] Q. Zhang, L. Cheng, R. Boutaba. Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*. 2010, vol. 1, iss. 1. pp. 7-18
- [12] Q. Wang, C. Wang, K. Ren , W. J. Lo, J. Li. Enabling public auditability and data dynamics for storage security in cloud computing. *Ieee transactions on parallel and distributed systems*, vol. 22, no. 5, may 2011 pp. 847-858
- [13] kennedy j, eberhart r c. Particle swarm optimization[c]//proceedings of ieee international conference on neural networks. piscataway new jersey: institute of electrical and electronics engineers, inc. 1995:1942-1948.
- [14] M. Mandal, A. Mukhopadhyay. A multiobjective pso-based approach for identifying non-redundant gene markers from microarray gene expression data. *Computing, communication and applications (iccca)*. pp. 1-6
- [15] b. Liu, l. Wang, and y. H. Jin. An effective pso-based memetic algorithm for flow shop scheduling. *Ieee transactions on systems, man, and cybernetics—part b: cybernetics*, vol. 37, iss. 1, feb. 2007. pp. 18-27
- [16] l. Messerschmidt, a. P. Engelbrech. Learning to play games using a pso-based competitive learning approach. *Evolutionary computation*, vol. 8, iss. 3, jun. 2004. pp. 280 - 288
- [17] K. W. Chau. application of a pso-based neural network in analysis of outcomes of construction claims. *Automation in construction*. vol. 16, 2007, pp. 642-646
- [18] Mohamed Ahmed Mohandes. Modeling global solar radiation using Particle Swarm Optimization (PSO). *Solar Energy*. vol. 86, iss. 11, nov. 2012, pp. 3137 - 3145
- [19] J. Yu, S. Wang, l. Xi. Evolving artificial neural networks using an improved pso and dpso. *Neurocomputing*. vol. 71, 2008, pp. 1054 - 1060
- [20] C.-L. Huang, J.-F. Dun. A distributed pso - svm hybrid system with feature selection and parameter optimization. *Applied soft computing*. vol. 8, iss. 4, september 2008, pp. 1381 - 1391
- [21] M. Mandal, A. Mukhopadhyay. A multiobjective PSO-based approach for identifying non-redundant gene markers from microarray gene expression data. *Computing. Communication and Applications (ICCCA)*. pp. 1-6
- [22] Archana Sarangia, Rabi Kumar Mahapatra, Siba Prasada Panigrahi. DEPSO and PSO-QI in digital filter design. *Expert Systems with Applications*. Vol. 38, Iss. 9, Sep. 2011. pp. 10966-10973
- [23] J. L. Fernandez-Martinez, E. G.-Gonzalo Stochastic Stability Analysis of the Linear Continuous and Discrete PSO Models. *Evolutionary Computation*. Vol. 15 , Iss. 3, Jun. 2011. pp. 405 - 423
- [24] .A. Sarangi a, R. K. Mahapatra b, S. Prasada Panigrahi. DEPSO and PSO-QI in digital filter design. *Expert Systems with Applications*. Vol. 38, 2011. pp. 10966 - 10973
- [25] A. Chandra, A. Chatterjee, P. Siarrya. A new social and momentum component adaptive PSO algorithm for image segmentation. *Expert Systems with Applications*. *Expert Systems with Applications*. Vol. 38, 2011. pp. 4998 - 5004