CHEMICAL ENGINEERING TRANSACTIONS
VOL. 62, 2017
Guest Editors: Fei Song, Haibo Wang, Fang He

ISBN 978-88-95608- 60-0; **ISSN** 2283-9216

A publication of
AIDIC
The Italian Association of Chemical Engineering
Online at www.aidic.it/cet

DOI: 10.3303/CET1762124

The Dynamic Simulation Platform for Chemical Gas Diffusion based on Cloud Computing

Qian Jia[a,b,c]

[a]North China University of Water Resources and Electric Power, Zhengzhou 450000, China;
[b]Henan Engineering Research Center of Water Pollution and Soil Damage Remediation, Zhengzhou 450000, China;
[c]Henan Key Laboratory of Water Environment Simulation and Treatment, Zhengzhou 450000, China
jiamoney@126.com

The toxic chemical gas leakage is a common safety accident in petrochemical enterprises. Once the leakage occurs in the process of storage and transportation, it will cause heavy losses and serious consequences. Therefore, it is of great significance to study the chemical gas leakage and diffusion in the plant. With the rapid development of internet technology, cloud computing is being more and more widely concerned and applied. Due to the dynamic and heterogeneous nature of service resources, developing appropriate resource allocation strategy according to the actual characteristics of cloud computing is the urgent problem to solve. In this paper, we propose a hybrid resource allocation algorithm based on ant colony algorithm and QoS algorithm. Firstly, we introduce the principle of ant colony algorithm and QoS algorithm. Secondly, an improved hybrid algorithm based on ant colony algorithm and QoS algorithm is proposed. Through the similarity analysis of the ant colony algorithm and the QoS algorithm, we can extract the service resources with high similarity. In the end, we make the simulation experiments on the three algorithms under the same conditions. The simulation results show that the proposed algorithm is superior to the traditional ant colony algorithm and the QoS algorithm in terms of performance and quality of service. We believe that it can be better applied to cloud computing environment resource scheduling under large scale tasks.

1. Introduction

The petrochemical industry plays an important role in the economic development of the country. The characteristics of petrochemical products are easy to corrosion and easy to spread, which can cause unexpected events during storage and transportation (Yu et al., 2015). Once the leakage occurs in the process of storage and transportation, it is difficult to prevent the diffusion, even cause serious consequences (Wang, 2016)).

The simulation system of dangerous chemical gas leakage diffusion can reduce the risk of gas leakage in chemical industry. With the rapid development of information technology, the cloud computing is becoming a mainstream computing paradigm. Through the integration of idle computing resources, to a certain extent, cloud computing can meet the needs of enterprises or individuals on the performance of computing. Cloud computing is a new computational model for task allocation according to requirements, which changes the original model of data processing by large servers. As a new computing model, the dynamic resource allocation with high efficiency and low cost becomes a hot research topic. Xiong (2012) proposed a task scheduling model based on genetic algorithm, which introduces the service quality standard to improve the fitness function. However, the ability of searching global optimal solution is relatively low. Liu (2011) proposed an improved particle swarm optimization algorithm by introducing the idea of reverse flight for mutation particle. This algorithm is relatively high demand for the surrounding environment because it does not consider the influence of inertia weight and learning factor on load balancing. Dutta (2013) presented a new version of the genetic algorithm which takes time and cost into consideration. The experimental results show that the proposed algorithm can guarantee a shorter task completion time and a lower cost. YarKhan (2002) introduces the application of simulated annealing algorithm (SA) in resource scheduling and the experimental results show that the algorithm has higher allocation efficiency. Li (2003) proposed the artificial fish school

Please cite this article as: Qian Jia, 2017, The dynamic simulation platform for chemical gas diffusion based on cloud computing, Chemical Engineering Transactions, 62, 739-744 DOI:10.3303/CET1762124

algorithm by simulating the behavior of fish. The algorithm has fast convergence speed and strong robustness, but it is not suitable for the scene with high precision.
In this paper, we proposed a hybrid resource allocation algorithm based on ant colony algorithm and QoS algorithm. First of all, we make a brief introduction on the principle of ant colony algorithm and QoS algorithm. Secondly, an improved hybrid algorithm based on ant colony algorithm and QoS algorithm is proposed. By making the similarity analysis on ant colony algorithm and the QoS algorithm, we can extract the high similarity service resources. In the end, we make a simulation experiments on three algorithms under the same conditions. The simulation results show that the proposed algorithm is superior to the traditional ant colony algorithm and QoS algorithm in terms of performance and quality of service.

2. Basic theory and method

2.1 Ant colony algorithm

Ant Colony Optimization (ACO) is an algorithm of finding the optimal path, which is proposed by Marco Dorigo in 1991. The principle of ACO is based on the behaviour of ant looking for shorter paths during foraging. ACO is a random search algorithm that the choice of path between different points depends on the choice of ant ahead. In other words, the more a path is selected in the past, the easier it is to be selected at present (Wang, 2014)).
Assuming that set *P* represents the collection of *n* points, and set *L* represents the collection of arbitrary lines between two points.

$$S=\{s_1,s_2,s_3,\ldots s_n\} \tag{1}$$

$$L=\{l_{ij} \mid p_i,p_j \subset P\} \tag{2}$$

Ant colony algorithm is based on the assumption that the ants will not choose the same point after passing it. We set a tabu table $tabu_x$ for each ant to prevent it from passing again. According to the behaviour of ants, the tabu table is set for each ant in the algorithm. When an ant passes through a certain point, the point is added to its tabu list to avoid selecting the point secondly. As the ants move, new elements are added to the tabu table at any time. At moment, we can get the transition probability of ant *x* from point *i* and point *j*.

$$p_{ij}^x(t)=\begin{cases}\dfrac{\mu_{ij}^{\alpha}(t)\lambda_{ij}^{\beta}(t)}{\sum\limits_{k\in U_x}\mu_{ik}^{\alpha}(t)\lambda_{ik}^{\beta}(t)} & j\in U_x\\ 0 & otherwise\end{cases} \tag{3}$$

U_x is the collection of points that ants have not visited before. α and β respectively represent the importance of pheromone and path intrinsic attributes.

$$\lambda_{ij}(t)=\frac{1}{l_{ij}} \tag{4}$$

In the above formula, l_{ij} represents the distance between point *i* and point *j*.
When all the points are added to the tabu table $tabu_x$, the ant completes a traversal, and the best path information is updated accordingly.

$$\mu_{ij}(t+\Delta t)=(1-\rho)\mu_{ij}(t)+\Delta\mu_{ij}(t) \tag{5}$$

$$\Delta\mu_{ij}(t)=\sum_{x=1}^{m}\Delta\mu_{ij}^x(t) \tag{6}$$

where, $\rho(0\leq\rho<1)$ indicates the volatilization coefficient of pheromone, and $(1-\rho)$ represents the residual coefficient of information. $\Delta\mu_{ij}^x(t)$ represents the pheromone of ant *x* between point *i* and point *j* in at reversals. $\Delta\mu_{ij}(t)$ represents the total amount of pheromone on the loop path, and $\Delta\mu_{ij}(0)=0$.
For Ant-Cycle model:

$$\Delta\mu_{ij}^{x}(t)=\begin{cases}\dfrac{R}{C_x} & ant\ x\ pass\,(i,j)\,in\,this\,cycle\\ 0 & otherwise\end{cases} \tag{7}$$

For Ant-Quantity model:

$$\Delta\mu_{ij}^{x}(t)=\begin{cases}\dfrac{R}{l_{ij}} & ant\ x\ pass\,(i,j)\,between\ t\ and\ t+\Delta t\\ 0 & otherwise\end{cases} \tag{8}$$

For Ant-Density model:

$$\Delta\mu_{ij}^{x}(t)=\begin{cases}R & ant\ x\ pass\,(i,j)\,between\ t\ and\ t+\Delta t\\ 0 & otherwise\end{cases} \tag{9}$$

In the above formula, *R* is a constant, which represents the total amount of pheromone released during the loop. C_x is the total length of the path traversed by ant *x*.
The ant colony algorithm has good robustness that it can find a new path when the system is abnormal. In addition, because the ant colony algorithm adopts the distributed mechanism, it is easy to combine with other methods to adapt to different situations

2.2 QoS algorithm

When a cloud task is submitted to the cloud platform, the task analyser will firstly analysed the need and then upload it to the server. The server will match the resource information with the demand parameters, and find the appropriate resource for distribution (Jiang, 2012).
Assume that the current collection of tasks is *T* and *n* represents the number of tasks.

$$T=\{T_1,T_2,T_3,\cdots T_n\} \tag{10}$$

$$T=(IP,C,N,E) \tag{11}$$

In the above formula, *IP* represents the primary key of the task, *C* means the type of the task, *N* represents a set of computations and the vector *E* is the expectation vector of QoS.

$$E=\begin{bmatrix}e_{iA}\\ e_{iB}\\ e_{iC}\\ e_{iD}\end{bmatrix} \tag{12}$$

In the above formula, e_{iA} is the expectation of the task T_i to the time, e_{iB} is the expected value of the cost, e_{iC} is the expected value of bandwidth and e_{iD} is the reliability expectation value of the node. Thus, the task T_i can be represented as follows.

$$T_i=(ip,c_i,w_i,e_{ik}) \tag{13}$$

Where, the parameters satisfy the following conditions.

$$\begin{cases}e_{iA}+e_{iB}+e_{iC}+e_{iD}=1\\ e_{iA},e_{iB},e_{iC},e_{iD}\in(0,1)\end{cases} \tag{14}$$

In this paper, the priority of task execution is determined by the matching satisfaction variance of the tasks in the system. The greater the variance of the matching satisfaction is, the larger the difference of the task in the system is. The specific sorting process is as follows.
(1) Calculating the matching degree between task and resource by weighted Euclidean distance formula.
According to the QoS demand vector of task and the performance parameters of the computing resources, we can get the matching degree to find the optimal matching combination. The formula of the matching degree between user requirements and resource nodes is as follows:

$$M_{ij}=\sqrt{\sigma_A(a_j-e_{iA})^2+\sigma_B(b_j-e_{iB})^2+\sigma_C(c_j-e_{iC})^2+\sigma_D(d_j-e_{iD})^2} \quad (15)$$

Where, σ_A, σ_B, σ_C and σ_D are the partial coefficients of QoS parameters. a_j, b_j, c_j and d_j are the ability of providing relevant QoS requirements. The larger M_{ij} is, the lower the matching degree between the task and the resource node is, which will decrease the satisfaction of user. On the contrary, the smaller M_{ij} is, the higher the customer satisfaction is. When the value of M_{ij} is 0, the task can get the optimal solution in the resource node, and users can get the highest quality of service too.
(2) Getting the match satisfaction.
We use weighted Euclidean distance to calculate the matching degree of task and node. The satisfaction degree of task T_i on execution node j is as follows.

$$TM_{ij}=1-M_{ij} \quad (16)$$

(3) Getting the satisfaction variance.

$$Mvar_i=TM_{ij}^2-A^2 \quad (17)$$

Where, A is the average satisfaction degree of the whole service systems.

$$A=\frac{\sum_{i=1}^{j}TM_{ij}}{j} \quad (18)$$

(4) Getting the priority of resource scheduling.
According to the value of the task and the QoS preference, the tasks are sorted according to the order from large to small. In the process of task priority setting, the influence of a single task on the whole tasks is an important index of task urgency.

3. Improved hybrid allocation algorithm

The hybrid allocation algorithm is based on ant colony algorithm and QoS algorithm, which makes up for the shortcomings of the two algorithms. In this paper, the optimal allocation strategy is obtained by calculating the similarity between ant colony algorithm and QoS algorithm.
Assume that the current task set is represented by vector $T_i(0<i\le m)$, and the set of computing resource is represented by vector $R_j(0<j\le n)$, we can represent the matrix of demand coefficient for computing resources in ant colony algorithm by the matrix as follows.

$$Ant_{ij}=\begin{pmatrix} a_{11} & \cdots a_{1j}\cdots & a_{1n} \\ \vdots & & \vdots \\ a_{i1} & \ddots & a_{in} \\ \vdots & & \vdots \\ a_{m1} & \cdots a_{nj}\cdots & a_{mn} \end{pmatrix} \quad (19)$$

On the same principle, we assume that the matrix of demand coefficient for computing resources in ant colony algorithm can be represented by the matrix Qos_{ij} as follows.

$$Qos_{ij}=\begin{pmatrix} q_{11} & \cdots q_{1j}\cdots & q_{1n} \\ \vdots & & \vdots \\ q_{i1} & \ddots & q_{in} \\ \vdots & & \vdots \\ q_{m1} & \cdots q_{nj}\cdots & q_{mn} \end{pmatrix} \quad (20)$$

We use Pearson algorithm to make analysis on the result matrix of the two algorithms.

$$sim(Ant,Qos)=\frac{\sum_{i\in T_{aq}}(Ant_{ij}-\overline{Ant})(Qos_{ij}-\overline{Qos})}{\sqrt{\sum_{i,j\in T_{aq}}(Ant_{ij}-\overline{Ant})^{2}(Qos_{ij}-\overline{Qos})^{2}}} \tag{21}$$

Where, T_{aq} is the intersection of the two results.

$$T_{aq}=Ant\bigcap Qos \tag{22}$$

4. Simulation experiment and result analysis

Cloudsim is a cloud computing simulation tool launched by Melbourne University's Grid Laboratory in Australia and the Gird bus project, which can be used to perform task scheduling in cloud environments. In order to show the performance of the improved algorithm more intuitively, experiments are carried out by ant colony algorithm, QoS algorithm and hybrid algorithm under the same conditions. The simulation experiment is carried out from the following two dimensions.

(1) The number of computing resource nodes is constant.

The number of tasks in the simulation environment is gradually increased from 50 to 1000, while the computing resource is constant.

(2) The number of tasks is constant.

The number of computing resources nodes in the simulation environment is gradually increased from 20 to 200 when the number of tasks is constant.

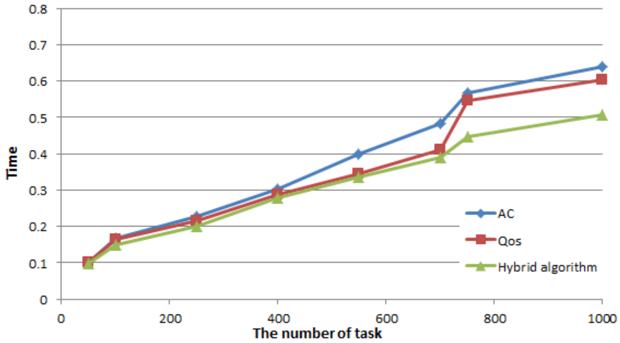

Figure 1: The relationship between the number of tasks and time

Figure 1 is the simulation comparison of three algorithms when the number of the resource nodes is 200. It can be seen in the figure that the computation time increases as the number of tasks increases. When the number of tasks reaches a certain value, the computing resources tend to be saturated and the subsequent tasks need to queue for resource allocation. As a whole, due to the better allocation method, the hybrid algorithm has better performance in time consuming.

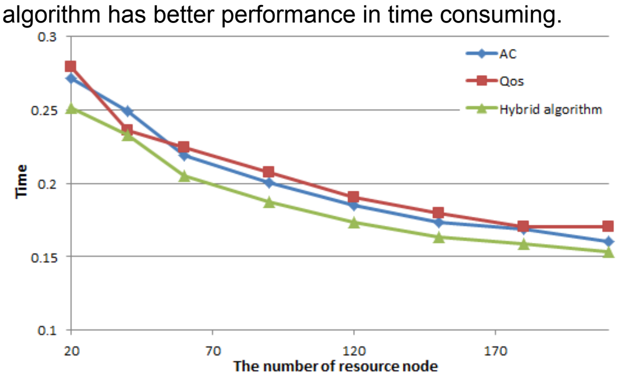

Figure 2: The relationship between the number of resource node and time

Figure 2 is the simulation comparison when the number of tasks is set to 300. In the figure, we can get that the computing time is gradually reduced with the increase of computing resource nodes. When the resource node reaches a certain value, since the three allocation algorithms do not need to wait in line, the change of computing time tends to be smooth. Compared with the other two algorithms, the hybrid algorithm has better performance in general.

Due to the huge amount of calculation, the dynamic simulation system for chemical gas diffusion requires high performance servers as the foundation. Through the experimental analysis, we can get that the improved algorithm can effectively reduce the cost of hardware in chemical plant or industrial park.

5. Conclusion

According to the characteristics of petrochemical industry and the present situation of safety management, combined with the development trend of information technology, a simulation platform for chemical gas leakage diffusion based on cloud computing is proposed. As for the resource allocation algorithm for cloud computing, this paper proposes a hybrid algorithm based on ant colony algorithm and QoS algorithm. Through the similarity analysis of the above two algorithms, a better resource allocation strategy is obtained. Simulation results show that the improved algorithm has higher resource scheduling strategy.

Acknowledgment

Research Startup Fund of North China university of Water Resources and Electric Power (No.: 201621).

References

Jiang H.J., 2012, Resoure Allocation algorithm for cloud computing based on QoS, Taiyuan: Taiyuan University of Technology, 22-24

Li X.L., 2003, A new intelligent optimization method-artificial fish school algorithm, Hangzhou: Zhejiang University, 22-25

Liu W.J., Zhang M.H., Guo W.Y., 2011, Cloud computing resource schedule strategy based on MPSO Algorithm, Computer Engineering, 11(37), 43-47.

Singh A., Dutta K., 2013, A Genetic Algorithm Based Task Scheduling for Cloud Computing with Fuzzy logic, Ieie Transactions on Smart Processing & Computing, 2(6), 367-372.

Wang X.H., 2014, Study on Resource Allocation Algorithms in Cloud Computing Environment, Chongqing: Chongqing University, 10-16.

Xiong C.C., Feng L., Chen L.X., Su J., 2012, Study on task scheduling algorithm based on genetic algorithm in cloud computing, Journal of Huazhong University of Science and Technology(Natural Science Edition), 1(40), 15-19.

YarKhan A., Dongarra J., 2002, Experiments with scheduling using simulated annealing in a grid environment, GRID, 2536, 232-233.

Yu F.F., Tian W.H., Wang T.C., 2015, Design and application of assessment system for process safety management in petrochemical enterprises, Journal of Safety Science and Technology, 4(11), 112-114.