The scheduling of power transaction with Improved swarm optimization algorithm

Lei Wang 1,2 a, Jingwei Lv 1,2, b, Qingbo Wang 1,2, c, Shuhong Shi 1,2, d and Zhenliao Lv 3, e

1 Nari Group Corporation/State Grid Electric Power Research Institute, Nanjing 210003 China
2 Beijing Kedong Electric Power Control System Co., Ltd., Beijing 100194 China
3 Computing Center, Northeastern University, Shenyang 110819, China

a ouhe163@126.com, b lvjingwei1993@163.com, c wangqingbomail@sina.com,
d wangqingbomail@sina.com, e lzl@mail.neu.edu.cn

Abstract: The application characteristics of power industry are very consistent with the technology mode of cloud computing. Task and resource scheduling is one of the basic problems in the cloud computing environment. It is a big challenge to find the optimal solution in a limited time because of the large solution search space. To meet the need of the higher requirements for quality of service and response in the power trading cloud computing platform, a resource scheduling method is proposed based on Improved Particle Swarm Optimization (PSO) algorithm. Firstly, weight of particle is optimized. And then dual fitness function for particle selection strategy is proposed not only aiming at minimum completion time, but also considering quality of computing measurement. Simulation experiments show that the algorithm has a significant improvement in task completion time and resource usage.

Keywords: Power transaction; Particle swarm optimization; Resource scheduling; Cloud computing.

1. Introduction

With the development of national grid towards big data informatization, the business data governance and integration of power enterprises are promoted to meet the requirements of data and business sharing. Resource scheduling in cloud computing environment is a process of allocating resources from the provider to users or services. The resources here can be physical or virtual resources. In most cloud computing environments, physical resources are usually virtualized first. Therefore, resource scheduling problems focus on virtual resources and application-level scheduling. Resource scheduling and task scheduling are two closely related scheduling problems in cloud computing. Task scheduling will be involved in the process of resource scheduling, and the effect of resource scheduling will also affect the results of task scheduling. Because of the particularity of the power trading cloud platform, especially for the application of spot trading, it is very necessary to ensure the quality of service (QoS) of users. It not only relates to the experience of users using cloud computing, but also affects the security of the grid. Therefore, QoS-based scheduling not only aims at the minimum completion time or the optimal
span, but also considers the quality of service. Formally, scheduling involves tasks that must be scheduled under resource constraints to optimize some objective functions [1].

At present, there are many researches based on QoS, literature [2] improved the Min-Min algorithm, according to the user's requirements to optimize the throughput of the system. Reference [3] Tasks with high requirements are scheduled preferentially with network. Reference [4] A multi-QoS scheduling strategy is proposed for multiple workflows in cloud computing. Different scheduling strategies with different QoS constraints are formulated for different types of users. Literature [5] Dynamic changes and migrations of cloud services are achieved through network virtualization without affecting service performance. The Openflow component is used to realize the service migration of virtual machine and improve the quality of cloud service. Reference [6] An improved genetic algorithm based on simulated annealing is proposed. The simulation results show that it is feasible and practical. Document [7] proposes a resource scheduling strategy based on greedy algorithm for cloud computing. This algorithm makes the task completion time short and the resource utilization rate high. These scheduling algorithms, which aim at improving the scheduling strategy, restrict the user characteristics and cannot use the dynamic changes of user requirements very well. In many scheduling algorithms [8-14], scheduling based on biological population algorithm is the mainstream method.

In summary, this paper aims at the dynamic characteristics of resource pool in the power trading cloud platform, optimizes resource scheduling and improves particle swarm optimization algorithm to meet the response time and quality of service objectives, and proposes a dual fitness function optimization to improve service performance.

2. Particle swarm optimization algorithm

2.1. Standard particle swarm optimization algorithm

Particle swarm optimization (PSO) is a calculation method. Each particle contains position and speed and moves through multi-dimensional search space. In each iteration, each particle determines its velocity according to its best position and the best particle position of the entire particle. Particle swarm optimization (PSO) includes a combination of local search and global search. Particle swarm optimization (PSO) has been widely used because of its simplicity and wide application value.

The first step to apply particle swarm optimization to scheduling problem is to encode the problem. A common method is to represent a particle as a 1 *n vector, n as the task and value assigned to each location of the number resource index. Therefore, particle representation resources are mapped to tasks. The M *N position matrix represents a solution based on matrix encoding, where M is the number of resources and N is the number of completed tasks. The elements of this matrix can have a value of 0 or 1, and there is a single element constraint in each column with a value of 1. The resource scheduling scheme can be expressed as matrix form in Formula 1.

\[
S = \begin{bmatrix}
S_{11} & S_{12} & \cdots & S_{1n} \\
S_{s1} & S_{s2} & \cdots & S_{sn} \\
\cdots & \cdots & \cdots & \cdots \\
S_{n1} & S_{n2} & \cdots & S_{nn}
\end{bmatrix}
\]

(1)

In this matrix, \(S_{mn}\) represents the mapping of task n in resource m, which is usually a 0/1 two valued matrix. A value of 0 indicates no allocation and a value of 1 for task assignment. Correspondingly, the position X and velocity V of particles can also be expressed in the form of matrices.
In the position matrix \( X \), the position of the particle represents a scheduling scheme in resource scheduling, and each vector in the speed matrix \( V \) iteratively calculates the variable of the particle position transformation for the corresponding scheduling scheme.

The inertia weight \( \omega \) plays an important role in the iteration process of particle velocity and position updating. The larger the \( \omega \) value is, the stronger the particle motion is, which means that the global search ability of particles is stronger. However, the flexibility of particles is limited, which means that the local search ability of particles decreases. In general, the weights of particles are taken by the random iteration method with the specified threshold (4).

\[
\omega = \begin{cases} 
\frac{\omega_{\text{max}} - \omega_{\text{min}}}{\text{iter}_{\text{max}}} \times \omega_{\text{min}} & \omega \not\in \left\{ \omega_{\text{min}}, \omega_{\text{max}} \right\} \\
\omega_{\text{min}} + (\omega_{\text{max}} - \omega_{\text{min}}) \times \text{rand} & \omega \in \left\{ \omega_{\text{min}}, \omega_{\text{max}} \right\}
\end{cases}
\]  

(4)

2.2. Improved particle swarm optimization algorithm

In order to better balance the global search ability and the local search ability of the particle, Rand in Eq. (4) no longer uses uniformly distributed random numbers, but uses Gaussian distribution of mean \( \mu \) and standard deviation \( \sigma \). This can prevent the algorithm from falling into local optimum and improve the global search ability to achieve global optimum. It also increases the diversity of particles.

Therefore, the position and velocity of particles are updated to:

\[
\begin{align*}
V_{t+1} &= \tau V_t + c_1 r_1 (P_{\text{best}} - V_t) + c_2 r_2 (G_{\text{best}} - V_t) \\
V_{t+1} &= P_t + V_{t+1}
\end{align*}
\]  

(5)

Here, \( V_t \) and \( P_t \) are the velocity and position of the particle at time \( t \), \( \tau \) is the parameter that the user controls the current velocity of the particle, \( c_1 \) and \( c_2 \) are acceleration parameters related to the particle itself and the global particle respectively, and \( r_1 \) and \( r_2 \) are random numbers that regulate the state of the particle. \( P_{\text{best}} - V_t \) represents the best particle adjustment difference of individuals. Accordingly, \( G_{\text{best}} - V_t \) represents the global optimum particle adjustment difference.

Particle fitness computing is the key to particle swarm optimization. Considering the balance between response time and quality of service in the power trading cloud platform, a dual-objective fitness model is proposed. The fitness is decomposed into two parts, as shown in the formula (6).

\[
F = \lambda \left( \frac{1}{\max \left( \sum \text{time}_{i,j} \right)} \right) + (1 - \lambda) \left( \frac{1}{\max_{x \in \Phi} (L_x)} \right)
\]  

(6)
The formula consists of two objectives. The first part is the task completion time of resource j, where \( time_{i,j} \) represents the execution time of task i in resource j. The latter part is the user service quality of the current task. \( x_i \in D(k) \) represents the response time of all subtasks k that the task i depends on. \( L_i \) represents the quality of service constraints, and is the indicator of the quality of service for each task. In this paper, it is easy to calculate, assuming that the QoS index of task i is constant.

2.3. Resource scheduling process
The algorithm first initializes the particle swarm parameters, then initializes the position and speed of the particle swarm, and then enters the iterative scheduling process. The specific resource scheduling process is shown in Figure 1.

Fig. 1 Flowchart of resource scheduling
3. Simulation experiment

In order to verify the effectiveness of the algorithm, a cloud resource scheduling simulation environment is built. CloudSim simulation environment was built on a server equipped with Intel i5-2400M CPU, Nvidia HD Graphics 3.10GHz processor and 16G memory. It is assumed that the user has submitted several tasks, and then allocated the tasks to the different tasks through the scheduling resources to perform the tasks. Through comparative experiments, the performance of ACO and PSO under different task conditions and resource supply conditions is compared. The specific experimental environmental parameters are shown in Table 1.

| Item            | Parameters | Configuration |
|-----------------|------------|---------------|
| Data resource   | # of data host | 5             |
|                 | Share of data | Yes           |
| VM resource     | # of VM     | 2-10          |
|                 | Memory of VM | 512M          |
| task            | # of tasks  | 20-50         |

According to the parameters in Table 1, the resource scheduling is simulated under the condition of changing the number of virtual machines and tasks. The completion time and resource utilization of ant colony algorithm and particle swarm optimization are compared. The experimental results are shown in figures 2 and table 2.

![Fig. 2 Time comparison result under different number of tasks](image)

| Item                   | ACO          | Improved PSO |
|------------------------|--------------|--------------|
| Maximum usage rate     | 35.83%       | 36.97%       |
| Minimum usage rate     | 28.46%       | 32.56%       |
| Average usage rate     | 32.15%       | 34.77%       |

As can be seen from the results of Figure 2, the task completion time of the improved PSO algorithm decreases as a whole compared with the ant colony algorithm. Moreover, as the number of tasks increases, the rate of decline is even more significant. From the resource utilization of Table 2, we can see that the utilization rate of the improved PSO algorithm has been improved, and the resource utilization is more balanced.
In order to further illustrate the adaptability of the improved PSO algorithm in the dynamic changes of resources, by changing the number of virtual machines, the improved PSO algorithm and ACO ant colony algorithm are simulated and compared. The experimental results are shown in Figure 3. As can be seen from Figure 3, when the number of virtual machines increases, the task completion time of the improved PSO algorithm decreases faster than that of the ant colony algorithm, which shows that the improved PSO algorithm can more effectively adapt to the dynamic scheduling of resources.

![Fig. 3 Time comparison result under different number of virtual machines](image)

4. Conclusion
Aiming at the high requirement of quality and response time in power cloud platform, the resource list is established by using the dynamic characteristics of resource pool, and the response time and quality of service fitness measurement indexes are proposed. In the process of candidate list generation, based on the improved particle swarm optimization algorithm, the inertia weight and update strategy of the particles are adjusted. Simulation results show that the proposed resource scheduling method can further shorten the response time and improve the resource utilization under the condition of satisfying the quality of service. Next, the main research should focus on the algorithm application system development, and establish an algorithm extension platform, laying the foundation for further in-depth study.

Acknowledgments
This work was financially supported by State grid technology project “Research on Key Technologies of Mobile Application and Big Data Analysis in Power Trading Platform”SGEPRI-KJB-KJ (2017)2941.

References
[1] Karger, David, Cliff Stein. Scheduling algorithms, Algorithms and theory of computation handbook[M]. Chapman & Hall/CRC, 2010.
[2] He X.S., Sun X. QoS Guided Min-Min Heuristic for Grid Task Scheduling [J]. Journal of Computer Science and Technology, 2003, 18(4):442-451.
[3] Sameer S. C., Joshi R.C. A Heuristic for QoS Based Independent Task Scheduling in Grid Environment[C]. International Conference on Industrial and Information Systems, 2010:102-106.
[4] Meng X., Cui L.Z., Wang H.Y., et al. A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing[C]. IEEE International SymPosium on Parallel and Distributed Processing with Applications, 2009:629-634.
[5] Hao F., Lakshman T.V., Mukherjee S., et al. Enhancing Dynamic Cloud-based Services using Network Virtualization [J]. Computer Communication Review, 2010, 40(1), 67-74.
[6] Liu F, Bi L, Yang J. An Improved Genetica Algorithm for Cloud Computing Resources Scheduling [J]. Computer Measurement & Control, 2016, 24(5): 202-206.
[7] Xiang B, Zhang B.B, Zhang L. Greedy-Ant: Ant Colony System-Inspired Workflow Scheduling
for Heterogeneous Computing [J]. IEEE Access, 2017, 5(2): 11404-11412.

[8] Sun D W, Chang G R, Li F Y, et al. Optimizing Multi-Dimensional QoS Cloud Resource Scheduling by Immune Clonal with Preference [J]. Acta Electronica Sinica, 2011, 39(8): 1824-1830.

[9] Madivi R., Kamath S.S. A hybrid bio-inspired task scheduling algorithm in cloud environment [C]. Computing, Communication and Networking Technologies (ICCCNT), 2014 International Conference on. IEEE, 2014: 1-7.

[10] Bossche R.V. D., Vanmechelen K., Broeckhove J. Cost-efficient scheduling heuristics for deadline constrained workloads on hybrid clouds [C]. Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on. IEEE, 2011: 320-327.

[11] Duan W J, Fu X L, Wang F, et al. QoS constraints task scheduling based on genetic algorithm and ant colony algorithm under cloud computing environment [J]. Journal of Computer Applications, 2014, 34 (S2): 66-69.

[12] Nie Q B, Cai T, Wang N. Application of improved ant colony algorithm in resource allocation of cloud computing[J]. Computer Engineering and Design, 2016,37(8):2016-2020.

[13] Zhao H W, Li S P. Research on resources scheduling method in cloud computing based on PSO and RBF neural network. [J]. Computer Science, 2016, 43(3): 113-118.

[14] Song Y J, Ye C M, Huang Z X. Cuckoo search algorithm for multi-resource leveling optimization[J]. Journal of Computer Applications, 2014, 34(1): 189-193.