ISSN 2616-5775 Vol. 6, Issue 5: 9-13, DOI: 10.25236/AJCIS.2023.060502

Cloud workflow scheduling optimization research based on ant colony algorithm

Yongqiang Li[1,a], Da Chen[1,b], Junjie Hao[1,c,*]

[1]Hemei Company, Henan Energy Group Co., Ltd., Hebi, China
[a]lyq5365@163.com, [b]chenda430@163.com, [c]haojunjiehemei@163.com
[*]Corresponding author

Abstract: *In view of the fact that enterprises can improve productivity through workflows, the problem of huge number of multi-user workflows and unbalanced user load on cloud computing platforms is investigated, based on an ant colony algorithm for decision task optimisation. With the help of a colony intelligence algorithm, automatic scheduling of tasks and optimisation of resources are accomplished while satisfying the enterprise's needs, while meeting server time constraints and load balancing. The optimised cloud workflow scheduling algorithm builds an intelligent business system that is highly flexible and scalable, and three times faster in terms of processing speed than before the optimisation.*

Keywords: *cloud computing; workflow; ant colony algorithm; task scheduling*

1. Introduction

With the development of technology, workflow gradually shows a trend of increasing demand for storage space and higher requirements for fast parallel execution capabilities, while cloud computing encapsulates computing resources and storage resources across computing into services open to users[1], which makes cloud-oriented workflow proposed as a solution. From the perspective of a cloud computing user, workflow provides the definition, flexible configuration, and automated operation of complex applications, which in turn can improve the quality of cloud services; from the perspective of a cloud computing service, workflow provides automatic scheduling of tasks, optimization, and management of resources, which in turn compresses the operating costs of cloud computing. The completion of workflows depends on the scheduling of system resources, so how to optimize the scheduling of resources has become a hot topic of research.

Most of the existing studies consider the impact of task executor's efficiency, interest and experience on workflow execution, and less on workload when multiple instances arrive at the same time, which has the problem of inefficient algorithms. The literature [2] proposes a workflow scheduling algorithm that optimizes execution-style time by evaluating the range of suitable executions, but it causes the problem of over-input. Literature [3] quantifies the factors using member collaboration and task completion quality, and finally selects candidates with high overall scores for assignment.

The ant colony algorithm is an intelligent algorithm used to find the optimal path. The literature [4] used the ant colony algorithm to solve the travel quotient problem and achieved better results; the literature [5] used the ant colony algorithm to implement a distributed system architecture which can reduce the energy consumption of cloud data centers by dynamically allocating virtual machines; the literature [6] used the ant colony algorithm to solve the multi-objective resource-constrained task scheduling problem; the literature [7] used the ant colony algorithm to solve the resource scheduling problem of cloud platforms; The literature [8] combined the ant colony algorithm to optimize the task scheduling solution; the literature [9] investigated a scheduling control mechanism based on workflow instances. None of these algorithms link scheduling between tasks and do not consider task scheduling in the workflow state. In summary, in order to solve the problem of inefficiency of the algorithm due to the impact of executor load on the process when instances arrive densely, a task allocation strategy based on the ant colony algorithm is proposed to solve the problem of inefficiency of the algorithm due to workload. This paper firstly introduces the two major foundations of cloud workflow - cloud computing and workflow technology, and on this basis explains the background of cloud workflow, followed by an introduction to the application of the ant colony algorithm, and finally optimizes the ant colony algorithm and analyses the results of the optimized ant colony algorithm in cloud workflow.

ISSN 2616-5775 Vol. 6, Issue 5: 9-13, DOI: 10.25236/AJCIS.2023.060502

2. Cloud Workflow Technologies

2.1. Cloud computing

In 2007, IBM and Google announced their collaboration in the field of "cloud computing"[10], inheriting and developing distributed computing, parallel computing and grid computing. Cloud computing is a kind of distributed computing, through virtualisation technology, a large number of idle computer resources distributed on the network are combined into a huge resource pool, constituting a computer with super computing power, various application systems can obtain computing resources, storage space and various software services according to their needs, fully realising the function of dynamic autonomy; from the user's point of view, the user does not need to know anything about cloud From the user's point of view, the user does not need to know anything about cloud technology to access the services provided by the service provider[11], as long as the user puts forward his or her own requirements according to the task, the cloud data center automatically decomposes the task, allocates system resources and executes the task.

The services provided by the cloud are transparent to the user, who simply puts forward his requirements through a simple interface and then leaves it to the huge cloud computing processing center, which first automatically splits the program into countless smaller sub-programs and then allocates them to multiple service sub-nodes in the cluster for processing, with the results aggregated by the processing center after each sub-node has performed its task and responded to the user. In this way, network service providers can execute tens of millions of tasks in a short period of time, realising the computing capabilities of a supercomputer.

2.2. Workflow task scheduling

From the emergence of simple form-based applications in the 1980s, to mature heterogeneous, loosely coupled business workflows and grid workflows in the 1990s. To the more recent cloud workflow management systems through virtualisation technology, where all computing or storage resources are managed in a unified manner so that task scheduling can be carried out in an orderly and efficient manner and business processes can be automated.

Zhu et al. investigated the application of simulated annealing and forbidden search algorithms to solve workflow scheduling problems in Grid environments[12]. Tao et al. considered execution span and energy as scheduling objectives and used a two-stage genetic algorithm to obtain a scheduling optimal solution through crossover operations with multiple parents[13]. Hai et al. authors designed an improved genetic algorithm IGA for virtual machine selection when scheduling workflows and obtained a better scheduling solution than the original genetic algorithm[14]. The above algorithms have problems such as being prone to local optima and slow convergence, so this paper selects the ant colony algorithm for task scheduling.

2.3. Application of ant colony algorithm

Table 1: Common examples of ant colony algorithms

Question name	Researcher	Name of algorithm
Traveling Salesman Problem	Dor igo,Maniezzo&Colorni	AS
	Gambardella&Dorigo	Ant-Q
	Stutzle&Hoos	MMAS
Assignment problem	Maniezzo	ANTS-QAP
	Maniezzo&Colorni	AS-QAP
Vehicle path planning	Bullnhe imer, Hartl&Strauss	AS-VRP
	Reimann, Stummer&Doerner	SbAS-VRP
Scheduling problem	Pfahringer	AS-OSP
	Bauer et al.	ACS-SMTTP
	Den Besten,Stutzle&Dorigo	ACS-SMTWTP
	Blum	ACO-SSP
Robot path planning	Feihu Jin,Bingyong Hong	ACA
Continuous function optimization	Lei Wang, Qidi Wu	ACA

Ant colony algorithm (ACO)[15], also known as ant algorithm, is a kind of bionic algorithm obtained

ISSN 2616-5775 Vol. 6, Issue 5: 9-13, DOI: 10.25236/AJCIS.2023.060502

by simulating the path finding way of ants in nature, and is a kind of chance-based algorithm used to find the optimal path in a graph. The higher the concentration of pheromone, the more likely the path is to be chosen by later ants, thus forming a positive feedback phenomenon. It is able to find the shortest path that starts from the origin, passes through a number of given demand points and eventually returns to the origin.

The existence of an ant colony algorithm employs a positive feedback mechanism that allows the search process to converge and eventually approach the optimal solution.

Each individual can change its surroundings by releasing pheromones, and can sense real-time changes in its surroundings, and communicate indirectly between individuals through the environment. It is easy to find the global optimal solution. The common examples are shown in Table 1.

3. Ant colony optimization algorithm

3.1. Ant colony algorithms

In the 1990s, Dorigo M et al studied the foraging behaviour of a highly structured ant colony by designing a "double-branching bridge experiment" and found that the colony always found the shortest path to food and could search again to produce a new shortest path as the location of food changed. It was observed that ants can secrete pheromones along the path to find food and the rest of the ants can sense the concentration of pheromones during movement and thus choose the direction of movement. Figure 1 shows the flow chart of the ant colony algorithm.

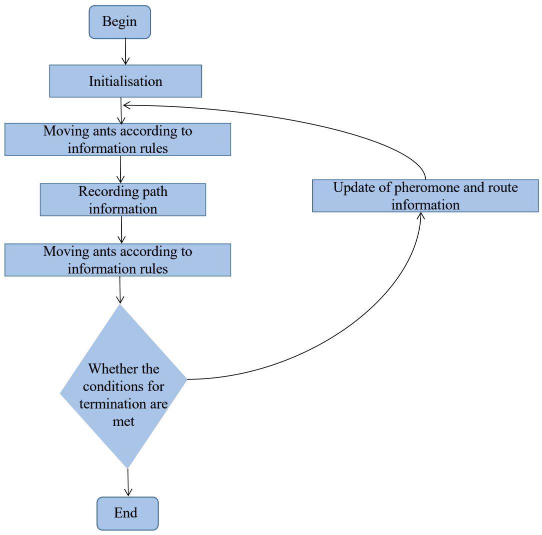

Figure 1: Shows the flow chart of the ant colony algorithm.

3.2. Optimised ant colony algorithm

To address the problem of unstable population quality caused by strong randomness in the initialisation phase of the ant colony algorithm, the initialisation of the ant colony algorithm using Tent chaotic sequences not only improves the quality of the initial individuals and enhances the stability of the algorithm, but also makes the search time decrease, the convergence speed increase and enhances the optimisation-seeking accuracy of the algorithm. Secondly, to address the situation that the search phase is prone to local optimum, the addition of bi-directional Tent chaos perturbation is added to improve the local search capability, increase the population diversity and reduce the possibility of falling into a local optimum.

Chaos is an objective non-linear phenomenon with the advantages of high randomness and regularity, where the Tent mapping and the chaotic sequence expression generated by the Tent mapping.

$$G(y)=\begin{cases}2y, 0\le y\le \frac{1}{2}\\ 2(1-y), \frac{1}{2}<y\le 1\end{cases} \tag{1}$$

ISSN 2616-5775 Vol. 6, Issue 5: 9-13, DOI: 10.25236/AJCIS.2023.060502

The Tent mapping is transformed using a Bernoulli shift to produce a chaotic sequence, and the transformed equation is shown in (2).

$$y_{i+1} = \left(2y_i\right)\mathrm{mod}1 \tag{2}$$

Then the chaotic perturbations generated by the chaos algorithm are as follows.

Step 1 Generate chaotic variables G(i) in Eq. (2)

Step 2 Introduce chaotic variables in the solution space, taking Xi as the i-th dimensional variable, whose carrier introduction is represented as shown in Eq. (3).

$$X_i = \min_i + (\max_i - \min_i) \cdot G_i \tag{3}$$

Step 3 The individual to be prepared for the chaotic perturbation is given as X', the chaotic perturbation variable is set as $newX$, and the perturbed individual is denoted as $newX'$, then $newX'$ can be derived from Equation (4).

$$newX' = \left(X' + newX\right)/2 \tag{4}$$

Step 4 Chaos perturbation is performed on the current optimal ant individual in the optimization-seeking stage, and a new type of chaotic perturbation is used for the multidimensional function with the following equation.

$$R_{\mathrm{d}} = \alpha\beta \cdot \left|\frac{1}{n} \cdot \sum_{k=1}^{n} Nest_{k,d} - Nest_{best,d}\right| \tag{5}$$

where, $\alpha = \begin{cases} 1, 0 \le a \le 0.5 \\ -1, 0.5 < a \le 1 \end{cases}$, α generates the positive and negative directions, β is the scale factor, $\frac{1}{n} \cdot \sum_{k=1}^{n} Nest_{k,d}$ is the average value, and $Nest_{best,d}$ is the optimal variable value.

The optimised process is shown in Table 2.

Table 2: Optimised ant colony algorithm flow

Algorithm 1.1 Optimized ant colony algorithm flow
1 Initialization parameters colony size u, total pheromone release q , pheromone α, heuristic function importance β, pheromone volatility factor ρ, the aximum number of iterations itermax
2 Chaos initializes the population and, using Equation 2, puts the chaotic sequence into the solution space of the ant colony objective function
3 while(iter<itermax) do
4 for i in u
5 for Each solution construction
6 1)Construction of solutions to further problems following pheromone and heuristic information guidelines
7 2)Perform pheromone local updates
8 end for
9 end for
10 1)Local search using certain obtained solutions as a starting point
11 Use Equation 5 for bidirectional chaotic perturbation and determine the Tent perturbation radius from the calculated results Determine the direction of positive and negative random perturbations by α to retain the best choice
12 2)Global pheromone update based on acquired solutions
13 end while
14end

4. Experimental Numerical Simulation Analysis

WorkflowSim was selected as the experimental platform, and the initialization parameters were set as follows: population size 300, maximum number of iterations 100, and threshold 0.6. Table 3 shows the results of scheduling by particle swarm algorithm, ant colony algorithm and optimized ant colony algorithm under the same experimental environment.

Table 3: Algorithm scheduling results

Algorithms	Time	Processing speed
Particle swarm algorithm	77.19	20
Ant colony algorithm	62.37	44
Improved ant colony algorithm	44.82	98

As can be seen from Table 3, in terms of time, the improved ant colony algorithm improves 41.94% over the particle swarm algorithm and 28.14% over the ant colony algorithm. In terms of processing speed, the improved ant colony algorithm is 4.9 times faster than the particle swarm algorithm and 3 times faster than the ant colony algorithm.

5. Conclusions

In order to solve the problem of unbalanced task assignment load in cloud workflow, this paper proposes a cloud workflow scheduling method based on an improved ant colony algorithm, which models the task load through the improved ant colony algorithm, enabling multiple targets to achieve better load balancing, and experimentally verifies the effectiveness of the proposed method, i.e., it reduces the total process load while ensuring maximum collaboration and improves the execution efficiency.

References

[1] Zhang L, Chen Y, Sun R, et al, A task scheduling algorithm based on PSO for grid computing, International Journal of Computational Intelligence Research, Vol.4, No.1, 2008, pp.37- 43.
[2] Mboula J E N, Kamla V C, Djamegni C T. Cost-time trade-off efficient workflow scheduling in cloud [J].Simulation Modelling Practice and Theory, 2020, 103: 102107.
[3] Shen M, Tzeng G H, Liu D R. Multi-criteria task assignment in workflow management systems [C]//Proceedings of the 36th annual Hawaii international conference on system sciences.Big Island, HI, USA: IEEE, 2003.
[4] Huang Z, Lu X, Duan H.A task operation model for resource allocation optimization in business process management [J]. IEEE Transactions on Systems, Man, and CyberneticsPart A: Systems and Humans, 2012, 42 (5): 1256- 1270.
[5] Farahnakian F, et al. Using ant colony system to consolidate VMs for green cloud computing [J]. IEEE Transactions on Services Computing, 2015, 8 (2): 187-198.
[6] Li Zhongjin, Ge Jidong, Yang Hongji, et al A security and cost aware scheduling algorithm for heterogeneous tasks of scientific workflow in clouds [J] Future Generation Computer Systems, 2016, 65: 140- 152 DOI: 10.1016/j. future .2015. 12.014.
[7] Hu Haiyang, Li Zhongjin, Hu Hua, et al. Multi-objective scheduling for scientific workflow in multicloud environment [J]Journal of Network and Computer Applications, 2018, 114: 108- 122. DOl: 10. 1016/j.jnca. 2018.03.028.
[8] Peng Wuliang, Wang Cheng 'en. ACO for solving resource- constrained project scheduling problem [J]. Journal of System Simulation, 2009, 21 (7) : 1974-1978
[9] Hu Haiyang, Zhang Xiaofei, Hu Hua, et al. Decision support method based on process mining [J]. Computer Integrated Manufacturing Systems, 2013, 19 (8): 1755-1770
[10] IBM. Google and IBM announced university initiative to addressinternet-scale computing challenge [EB/OL].http: //www-03.ibm.com/press/us/en/pressrelease/22414 wss, 2007 October.
[11] Shen M, Tzeng G H, Liu D R Multi-criteria task assignment in workflow management systems[C]. Proceedings of the36th Annual Hawaii International Conference on System Sciences Washington, D.C. USA: IEEE, 2003: 202b. DOI: 10.1109/HICSS .2003.1174458.
[12] Zhu Z, Zhang G, Li M, et al. Evolutionary multi-objective workflow scheduling in cloud[J]IEE Transactions on Parallel and Distributed Systems, 2015, 27(5): 1-20.
[13] Tao Fei, Feng Ying, Zhang Lin. CLPS-GA: A case library and Pareto solution-based hybrid genetic algorithm for energy-aware Cloud service scheduling [J]. Applied Soft Computing, 2016, 19(2): 264-279.
[14] Hai Z, Tao K, Zhang X. An approach to optimized resource scheduling algorithm for open-source cloud systems [C] / Fifth Annual China Grid Conference. IEEE Press, 2015: 124-129.
[15] J. L. Deneubourg, S. Aron, S. Goss, and J. M. Pasteels. The self-organizing exploratory pattern of the argentine ant [J]. Journal of Insect Behavior, 1990.3(2).