Improved Optimal Path Algorithm Based on Ant Colony Algorithm

Zhaohua Long[1, a], Ruifang Dong[2, b]

[1]Chongqing University of Posts and Telecommunications, Chongqing 400065, China.

[2]Chongqing University of Posts and Telecommunications, Chongqing 400065, China.

[a]longzh@cqupt.edu.cn, [b]2524668099@qq.com

Keywords: Ant colony algorithm, ACS, particle swarm optimization algorithm, P-ACS

Abstract: Ant colony algorithm is based on Ant System (AS) and it is a very important group intelligence algorithm, which is used in many fields, but there are also some shortcomings. The classical ant colony algorithm is analyzed and studied, and an improved P-ACS algorithm is proposed based on ACS algorithm in this paper. Through analysis and experiment, it is found that although the performance of ACS algorithm is higher than AS algorithm, there are still some problems, such as: falling into local optimal solution, search stagnation, and slow initial convergence.The important reason for the above problems is that the pheromone update can not accurately reflect the actual situation of the path. Aiming at this problem, a P-ACS ant colony algorithm is proposed based on particle swarm optimization algorithm(PSO). The algorithm optimizes the pheromone update strategy from three aspects: pheromone concentration range setting, initial pheromone setting and global update strategy improvement.

1. Introduction

Routing algorithms have always been a key issue in the field of network research. To improve routing performance, researchers have made many attempts and improvements.

In [1], a routing algorithm based on the improved Dijksta algorithm for solving the optimal path is proposed. As the number of nodes increases, the search overhead is higher and the efficiency is low, which is not suitable for large-scale network environments.

In [2] proposes a classic routing algorithm for heuristic search, A* algorithm. The algorithm introduces an evaluation function for estimating the length of the path.

In [3], an improved optimal path algorithm based on simulated annealing algorithm is proposed. The simulated annealing algorithm is a probability-based algorithm. Initially set a value. As the algorithm performs, the temperature is continuously reduced, and the probability jumps out. The random path is randomly searched in space. Finally, the solution converges to the global optimal. . The characteristic of the algorithm is that it jumps out with certain probability and converges to the global optimal solution under the condition of local optimal solution.

In [4], the ant colony algorithm is applied to solve the optimal path. The ant colony algorithm is a bionic algorithm, which is a simulation of the foraging behavior of ants in nature. In the process of searching for food, ants leave a substance that can transmit information, called pheromone. When ants choose the path of the next step, they always choose a path with a high concentration of pheromone. Therefore, although a single ant has no regularity, many ant groups show positive feedback. The shorter the path, the more times the ants pass. The more the pheromone concentration remaining on the path is larger than the path length, the probability that the ant will select the path will be large, and finally an optimal path is searched through the information exchange between the individual ants.

Among the many algorithms for solving the optimal path, the ant colony algorithm and its improved algorithm have relatively good performance. It is robust, parallel, true feedback, easy to combine with other bionic algorithms, and is widely used to solve combinatorial optimization problems. This paper studies the classical ant colony ACS algorithm and proposes an improved P-ACS algorithm based on its shortcomings.

2. Ant colony algorithm

2.1 AS Mathematical Model

Given a directed graph of N nodes, M ants, the variable $C_{ij}(i, j = 1,2,3 \dots N)$ represents the cost value between the two nodes. $T_{ij}(t)$ represents the concentration of pheromone on the path of the i-node and the j-node in t iterations, which is used to simulate the secretion of ants. In the beginning we assume that the pheromone strength is the same on all paths, ie $T_{ij} = D$ (D is a given constant). The ant k (k = 1, 2, 3....M) selects the next path according to the concentration of the pheromone on the path during the search traversal. The ant calculates the transition probability to the next node according to the formula of (1), as follows:

$$P_{ij}^{k}(t) = \begin{cases} \frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{j \notin table_n} \tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)} & j \in table_n \\ 0 & j \notin table_n \end{cases} \tag{1}$$

In the above formula, the specific meaning of each symbol is as follows:

$\alpha \rightarrow$ The heuristic factor of pheromone, which is used to reflect the effect of pheromone on ant selection. The larger the value, the greater the probability that the path with high pheromone concentration is selected by ants, which reduces the randomness of path selection to some extent; $\beta \rightarrow$ The desired heuristic factor, which is used to indicate the extent to which the pheromone is affected by the ant's choice of path; $\tau_{ij}(t) \rightarrow$ The pheromone concentration of the path after t iterations; $\eta_{ij}(t) \rightarrow$ Heuristic function, the expression of expression in this article $\eta_{ij}(t) = \frac{1}{C_{ij}}$; $table_n \rightarrow$ Ants can choose the next set of nodes.

When the amount of pheromone left on the path is too large, the heuristic information will be buried and the algorithm will easily fall into the local optimal solution. Therefore, after the algorithm completes one iteration, the amount of pheromone on each path needs to be updated. As shown in equations (2) and (3):

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij} \tag{2}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{3}$$

$\rho(0 < \rho < 1) \rightarrow$ The volatilization factor of the pheromone on the path; $(1-\rho) \rightarrow$ The persistence factor of the pheromone on the path; $\Delta\tau_{ij} \rightarrow$ the pheromone increment of the path (i, j) in this iteration; $\Delta\tau_{ij}^{k} \rightarrow$The amount of pheromone left by the k_{th} ant in the path (i, j) in this iteration, and in the ant-cycle model, if the k_{th} ant does not pass the path (i, j), the value of $\Delta\tau_{ij}^{k}$ is zero. $\Delta\tau_{ij}^{k}$ can be expressed by equation (4):

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k}, & \text{when the } k_{th} \text{ ant pass the path}(i, j) \\ 0, & other \end{cases} \tag{4}$$

$Q \rightarrow$ A normal number; $L_k \rightarrow$ The length of the path that ant k has traveled in this round of travel.

2.2 AS Flow Chart

The specific implementation steps of the AS algorithm are as follows:

1) Initialize basic parameters, including α, β, ρ, number of ants M, number of iterations t of the algorithm, cost value C_{ij} between nodes, initial path pheromone concentration T_{ij};

2) Start, all ants move the next node according to the probability calculated by formula (1), and update $table_n$;

3) When the ant completes a loop, records the cost value of the path it passes through, and if the value is better than the current optimal solution, updates the value of the optimal solution;

4) When all the ants complete a cycle, update the pheromone concentration on the path according to Equations (2), (3) and (4);

5) When the algorithm is iterated to a specified number of times, the algorithm ends and the optimal solution is output;

2.3 Ant Cycle System (ACS)

The Ant Colony System (ACS) has three improvements in the AS algorithm.

The ACS system improves the state transition rule and uses the pseudo-random proportional rule to select the next node. The specific transfer rule is shown in Equation (5):

$$\mathrm{P} = \begin{cases} \arg max\{\tau_{ij}(t)^{\alpha}.\eta_{ij}(t)^{\beta}\} & q \leq q_0 \\ P_{ij}^{k}(t) & q > q_0 \end{cases} \tag{5}$$

The global update operation only updates the pheromone concentration on the optimal path, and its idea stems from the update strategy of elite ants. In the AS system, since all the paths that the ants pass are updated, the pheromone concentration on the path does not change much, which reduces the efficiency and probability of searching for the optimal path. In the ACS system, because the importance of the optimal path is highlighted, the pheromone concentration tends to bias the optimal path, which speeds up the search for the optimal path. The specific update rules are shown in Equations (6) and (7):

$$\tau_{ij}(t+1) = (1-\rho) * \tau_{ij}(t) + \rho\Delta\tau_{ij} \tag{6}$$

$$\Delta\tau_{ij} = {}^{Q}/_{C^{best}} \;\; Edge\ (i,j)\ is\ the\ global\ optimal\ solution, otherwise\ it\ is\ 0 \tag{7}$$

The local update strategy of the path is added in the ACS, that is, the pheromone concentration of the path is locally volatilized, and the probability that other paths are selected is increased, and the specific update rule is as shown in Equation (8):

$$\tau_{ij}(t+1) = (1-\rho) * \tau_{ij}(t) + \rho * \tau_0 \tag{8}$$

Its τ_0 is a constant, ranging from 0 to 1. When an ant passes a certain edge, it volatilizes the pheromone on the path with a certain probability, increasing the probability that other paths are searched, which can effectively avoid the algorithm being too early. Stuck into a state of stagnation.

3. P-acs

Compared with the AS algorithm, the ACS algorithm improves the performance of the algorithm. However, it is found through analysis and experiment that the ACS algorithm will still fall into the problem of local optimal solution, search stagnation and slow initial convergence. The important reason for the above problems is that the pheromone update cannot accurately reflect the actual situation of the path. Therefore, this paper proposes a P-ACS ant colony algorithm based on particle swarm optimization (PSO) [5]. The algorithm optimizes the pheromone update strategy from three aspects.

3.1 Pheromone concentration range setting

In the P-ACS algorithm, the algorithm of the maximum and minimum ant system is used to limit the range of pheromone concentration to $[\tau_{\min}\ \ \tau_{\max}]$, which avoids the search caused by excessive pheromone on the non-optimal path in the ant colony system. The phenomenon of stagnation and pheromone concentration is too low for long-term search. The specific update rule is: when the pheromone concentration on the path is higher than $\tau_{\max}$, it is updated to $\tau_{\max}$; when the pheromone concentration on the path is lower than $\tau_{\min}$, it is updated to $\tau_{\min}$.

3.2 Initial pheromone setting

In the ACS algorithm, the initial pheromone concentration is equal, leading to the blind search of the ants in the early stage of the algorithm, generating many invalid paths, causing errors in the update of the pheromone concentration on the path. This problem not only causes the initial search time of the algorithm to be long, but also causes the searched path to fall into local optimum due to the error of pheromone concentration update, which affects the performance of the algorithm. Combining the characteristics of particle swarm optimization (PSO) algorithm (initial particle random distribution, strong global search ability and faster search speed), this paper firstly finds sub-optimal path information through particle swarm optimization algorithm, and then information about these paths. The prime concentration is set to τ_{max}, and the pheromone values of other paths are p_0, where the range of p_0 values is limited between τ_{min} and τ_{max}, excluding the maximum and minimum values.

3.3 Improvements to the global update strategy

In the ACS algorithm, the global update strategy only updates the pheromone concentration on the single-cycle optimal path, and the idea stems from the elite ant's update strategy. In the AS algorithm, the update strategy is to update only all the paths that the ants pass, which results in little change in the pheromone concentration on the path, ignoring the role of the optimal path. These two algorithms are too extreme. On the one hand, the ACS algorithm pays too much attention to the optimal path effect of each iteration. Too much increase of the optimal path pheromone can easily lead to the algorithm falling into local optimum. On the other hand, the optimal path of a single iteration Strongly related to the global optimal path. Therefore, in the P-ACS algorithm, we introduce the function$\mu(x) = 1 - \tanh(x)$ as the adaptive dynamic change factor in the global update rule, where $\tanh(x)$ is the activation function in the neural network. This function can effectively distinguish the pheromone addition of different optimal solutions in each iteration, avoiding excessive pheromone over-addition on individual shorter paths instead of global optimal paths, resulting in partial optimality. The specific update rules are shown in Equations (9), (10), and (11):

$$\tau_{ij}(t+1) = (1-\rho) * \tau_{ij}(t) + \mu(x)\rho\Delta\tau_{ij} \tag{9}$$

$$\Delta\tau_{ij} = {}^{Q}/_{C^{best}} \ \ \textit{Edge } (i,j) \textit{ is the global optimal solution, otherwise it is } 0 \tag{10}$$

$$x = \frac{Cost_{thisValue} - Cost_{min}}{\overline{Cost} - Cost_{min}} \tag{11}$$

$Cost_{thisValue}$ represents the optimal path cost value in this iteration. $Cost_{min}$ represents the current global optimal solution cost value. $\overline{Cost}$ represents the cost value.

3.4 P-ACS Flow Chart

The P-ACS algorithm consists of two parts. The first part uses the particle swarm algorithm to search for optimized path information, where P_{best} represents the individual optimal solution and G_{best} represents the group optimal solution. The second part updates the pheromone concentration based on the path information searched by the particle swarm optimization algorithm, and makes full use of the improved parallel computing power, positive feedback mechanism and high precision of the ACS algorithm to find the optimal path. The P-ACS algorithm combines the advantages of the two algorithms. Compared with a single algorithm, the search efficiency and accuracy of the algorithm are improved. The fourth chapter shows that the P-ACS is superior to the ACS algorithm in performance.

4. Experimental analysis

tables or figures from other Publications, they must ask the corresponding publishers to grant them the right to publish this material in their paper.

4.1 Experiment and Simulation

The P-ACS algorithm solves the optimal solution is a dynamic feedback process. The performance of ant colony algorithm is four parameters: ant number M, pheromone heuristic factor α, expectation heuristic factor β, pheromone volatilization factor ρ, pseudo-random scale factor q_0. And efficiency plays a decisive role, N is the size of the problem. This paper cites the literature [6,7] for the TSP problem experiment to obtain the optimal combination parameters as the experimental parameters of the P-ACS algorithm (the classical problem of the optimal path, the RINA network routing algorithm studied in this paper is essentially Belong to the category of solving the optimal path). The parameter settings of the P-ACS algorithm are shown in Table 1.

Table 1. Three Scheme comparing.

Parameter	Value
M	$\sqrt{N} \sim {}^{N}/_{2}$
α	1
β	2.5
ρ	0.5
Q	1
q_0	To be determined
τ_0	0.3
Pheromone concentration interval	[0.1, 0.9]
NC	300

The experiment uses the network topology of the structure shown in Figure 1 to determine the parameter values to be determined in the parameters (50 experiments in this experiment, where the cost of the edge between nodes is a randomly generated integer between [1, 10]), find the node the optimal path from S to node D with the lowest cost. First, the parameters are initialized for the standard particle swarm optimization algorithm. Here, the initial particle number is set to 15, each particle dimension is 15 and iterated 10 times to obtain optimized path information, and then information is made for these paths. The initial distribution of the prime.

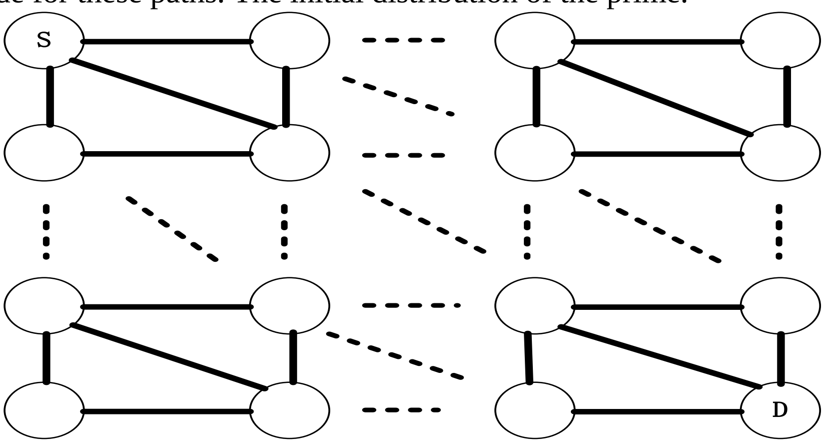

Figure 1. Experimental topology.

4.2 Results and Analysis

To avoid the contingency of the data, after the network topology and cost values are initialized, the P-ACS algorithm proposed in this paper is repeated 10 times to study the relationship between the q_0 parameter and the average optimal solution value and the average optimal solution iteration number. The experimental data is shown in Table 2:

Table 2. q_0 impact on P-ACS algorithm.

q_0	Average optimal solution	Average number of iterations
0.1	23.1	49
0.3	22.3	71
0.5	21.7	101
0.7	22.1	84
0.9	23.4	43

The experimental results in Table 2 show that the q_0 parameter has a certain influence on the efficiency of the P-ACS algorithm. When the value of q_0 is around 0.5, the experiment has the best effect. So, in the subsequent experiments in this paper, q_0 takes a value of 0.5.

5. Conclusion

The main content of this chapter is based on the ACS ant colony algorithm to propose an improved P-ACS algorithm for solving the optimal path problem. First, the mathematical model of the basic ant colony algorithm (AS) is abstracted, and the advantages and disadvantages of the AS algorithm are analyzed and summarized. Based on the AS algorithm, the researchers also proposed an improved algorithm based on the AS algorithm. The classic improved algorithms include sorting-based ant system, elite ant system, maximum and minimum ant system (MMAS), and ant colony system (ACS). This chapter provides an in-depth analysis of the improvements of these four ant colony algorithms and focuses on the ACS algorithm. Finally, based on the ACS algorithm, this paper proposes an improved algorithm (P-ACS). The main improvement points include three aspects: (1) pheromone concentration limit, improved algorithm to absorb the idea of maximum and minimum ant system algorithm, information The range limit of the prime concentration update is in one interval; (2) the update of the initial path pheromone concentration, the P-ACS algorithm combines the advantages of the particle swarm optimization algorithm and the ant colony algorithm, and the algorithm initially searches for the suboptimal path through the particle swarm optimization algorithm. The path pheromone concentration is set to the maximum value, and the pheromone concentration of other paths is randomly given a value within the range; (3) the global update strategy is improved, and the P-ACS algorithm increases the dynamic update factor of the global update to avoid Excessively increasing the single optimal path pheromone leads to the algorithm falling into local optimum. Finally, the experimental parameters of the P-ACS algorithm in the fourth chapter are determined through experiments and previous research results.

Acknowledgments

I would also like to thank my tutor for helping me. After deciding to do this, my tutor will often come to us for academic discussions and put forward some very important ideas, whether it is for my subject or for My growth has been greatly helped. At the same time, I would like to thank my brothers and sisters for setting an example for me, studying hard, studying hard, and being friendly.

References

[1] Kuznetsov N A, Fetisov V N. Enhanced-robustness Dijkstra algorithm for control of routing in the IP-networks [J]. Automation and Remote Control, 2008, 69(2):247-251.

[2] Chabini I, Shan L. Adaptations of the A* algorithm for the computation of fastest paths in deterministic discrete-time dynamic networks [J]. Intelligent Transportation Systems IEEE Transactions on, 2002, 3(1):60-74.

[3] Xu, Ying, Li, et al. A simulated annealing based genetic local search algorithm for;multi-objective multicast routing problems [J]. Annals of Operations Research, 2013, 206(1):527-555.

[4] Sun Y, Dong W, Chen Y. An Improved Routing Algorithm Based on Ant Colony Optimization in Wireless Sensor Networks [J]. IEEE Communications Letters, 2017:1-1.

[5] Wang D, Tan D, Lei L. Particle swarm optimization algorithm: an overview [J]. Soft Computing, 2018, 22(2):387-408.

[6] Yan X Y, Xia G L. Parameter Settings ofthe Ant Colony Algorithm in Solving TSP [J]. Computer Knowledge & Technology, 2016.

[7] Heng-Ji D U, Yong L I. Research on Affect Performance of Parameter Settings in Ant Colony Algorithm [J]. Modern Computer, 2012