

Retraction

Retracted: Simulation of Film and Television Transmission Path Based on Ant Colony Optimization Algorithm

Security and Communication Networks

Received 8 January 2024; Accepted 8 January 2024; Published 9 January 2024

Copyright © 2024 Security and Communication Networks. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

- [1] X. Zhao, "Simulation of Film and Television Transmission Path Based on Ant Colony Optimization Algorithm," *Security and Communication Networks*, vol. 2022, Article ID 2826527, 11 pages, 2022.

Research Article

Simulation of Film and Television Transmission Path Based on Ant Colony Optimization Algorithm

Xiyu Zhao 

Computer Science and Technology, Xidian University, Xi'an 123456, China

Correspondence should be addressed to Xiyu Zhao; 20031211593@stu.xidian.edu.cn

Received 16 June 2022; Revised 14 July 2022; Accepted 23 July 2022; Published 31 August 2022

Academic Editor: Jun Liu

Copyright © 2022 Xiyu Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the stagnation and precocity of the basic ant colony algorithm in solving the film and television transmission path problem, an ant colony optimization algorithm with forgetting factor is proposed. By adding forgetting factor to artificial ants, a new state transition formula is established, and pheromone update strategy is modified. Combined with the optimal error rate of the current solution, the state transition equation is adjusted. Due to the introduction of the forgetting factor, the new formula can be adjusted in time when the original ant colony algorithm calculation falls into the local optimal solution, and the selection and updation of the next node are carried out according to the new state transition scheme. Pheromone, which can be used to reduce the optimal value error, improves the tracking ability of the optimal value, modifies the path evaluation model, and calculates the probability of each path to the current optimal solution. The simulation results of the film transmission path problem (FTPP) example show that the improved algorithm takes less time and the result of path optimization is better.

1. Introduction

The film transmission path problem (FTPP) is a typical combinatorial optimization problem [1]. Based on the demand for timeliness and universality of film and television communication, many works hope to achieve the most coverage in the shortest time [2]. For the individual audience of film and television media, FTTP is a typical combination optimization problem [3]. How to coordinate and plan the optimal propagation path is the most important problem for FTTP. Solving the optimal solution given by FTTP is also the best way to reduce the consumption of resources in film and television transmission [4]. Ant colony optimization (ACO) is a kind of smart search algorithm. Ant colony algorithm has artificial intelligence based on collective behavior of decentralized self-organizing system [5, 6]. Ant colony algorithm has good advantages in solving optimal path of FTTP. The search process adopts a distributed computing method, and multiple individuals perform parallel computing at the

same time, which greatly improves the computing power and operating efficiency of the algorithm. However, it is easy to fall into local convergence [7]. The ant colony algorithm has the characteristics of positive feedback. The pheromone in the environment at the initial moment is exactly the same, and the ants complete the construction of the solution almost in a random way. These solutions are bound to have advantages and disadvantages. When the pheromone is updated, the ant colony algorithm leaves more pheromone on the path passed by the better solution, and more pheromone attracts more ants. This positive feedback process rapidly expands the initial difference and guides the whole system to evolve toward the optimal solution. Although positive feedback makes the algorithm have a better convergence speed, if the optimal solution obtained by the algorithm at the beginning is a suboptimal solution, the positive feedback will make the suboptimal solution quickly dominate, making the algorithm fall into a local optimum, and it is difficult to jump out of the local area. optimal. For the improvement of ant colony

algorithm, there are more famous elite strategy ant system and optimal sorting ant system. [8, 9]. In the grid environment, the balanced job scheduling is improved to optimize the water distribution system and solve the weapon target allocation problem [10].

Aiming at the stagnation behavior and premature convergence of basic ant colony algorithm in solving FTTP [11, 12], this paper constructs an ant colony optimization algorithm including forgetting factor, named as the forgetting factor ant colony optimization (FFACO) [13]. FFACO optimizes the transition formula and pheromone update rules. It shortens the time cost of propagating receptor conversion [14], maximizes the leading role of the current optimal solution, speeds up the convergence speed, and avoids the generation of local solutions [11]. The experimental simulation results confirm that FFACO can track the optimal solution by changing the forgetting factor [15], thus avoiding the accumulation of errors and improving the efficiency of the algorithm [16].

2. Ant Colony Optimization Algorithm

2.1. ACO Mathematical Model. When ants are looking for food, they make path selection by considering the length of the path and the pheromone on the path (the ant will release pheromone on the path it travels, and other ants can perceive this pheromone and prefer to choose a pheromone with a higher concentration). The pheromone concentration on the path is low at the beginning, and the ants will preferentially choose the shorter path according to the length of the path. With the passage of time, the pheromone on each path gradually accumulates and increases, and the shorter the path, the more ants walk through, the more pheromone remains on the path, and the more ants choose to take the path, thus forming a positive feedback. Eventually, the ants find an optimal path from the nest to the food source.

In the basic ant colony algorithm, ants mainly depend on the distance between nodes when choosing a path, that is, the shorter the distance, the greater the probability of the path being selected, and the longer the distance, the smaller the probability of the path being selected. As some subpaths in the global optimal path (subpaths are paths formed by the connection of any two nodes) may be longer, the probability of this subpath being selected is small, causing the ants to miss the subpaths that constitute the potential global optimal solution.

The FTTP problem is the problem of dissemination of works to various dissemination points. The location of the works can be regarded as the source of dissemination, and each receiving point can be regarded as a point in the dissemination path. Therefore, the FTTP problem is how to cover each dissemination point from the source of dissemination. The problem of the shortest path at the same time, which is represented by a fully weighted digraph $G = (V, A, d)$, variable $V = \{1, 2, \dots, n\}$ is the node set, where n is the number of nodes, $A = \{(i, j) | (i, j) \in V * V\}$ is an arc set, D is a set of arcs $arc(i, j)$. The purpose of FTTP is to find the shortest path when each recipient visits only once. When an ant is looking for food, it releases pheromones on

the path it travels, other ants can be attracted by the pheromone, and as more and more ants pass the path, more pheromones will deposit on this path, thereby attracting more ants to continue to choose this path [17, 18]. Therefore, according to this principle, ants search for the shortest path from the starting point to the food source and back again, the calculation logic is shown in the upper part of Figure 1, while the lower part of Figure 1 is the content of the time series algorithm for auxiliary calculation. This paper mainly introduces the optimal planning situation of ants in the time of searching paths and verifies the results of their shortest path selection.

The formal definition of basic ACO is provided below. In the initial stage of the algorithm, each ant is randomly placed on a node position:

$$p_{ij}^l = \begin{cases} \frac{\tau_{ij}^\alpha(t) \cdot \eta_{ij}^\beta(t)}{\sum_{k \notin tabu_k} \tau_{ik}^\alpha(t) \cdot \eta_{ik}^\beta(t)}, & j \notin tabu_k \\ 0, & j \in tabu_k \end{cases}, \quad (1)$$

$$\tau_{ij}(t+s) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}, \rho \in (0, 1), \quad (2)$$

α is an information heuristic factor, which determines the relative influence of pheromones, β is the expected heuristic factor, which indicates the relative importance of the expected value, and reflects the influence of path length on ant movement, ρ is a pheromone residue coefficient. $\tau_{ij}(t)$ is the trace amount of the pheromone at time ρ on $arc(i, j)$. The pheromone on the path will evaporate step by step. After time ρ , the track strength is updated according to equation (2):

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k. \quad (3)$$

In formula (3), $\Delta\tau_{ij}^k$ is the number of k ants in time t and time $t+s$ the amount of trace material (pheromones in real ants) per unit length, m is the number of ants. A pheromone updating model of ant circulation system is shown in formula (4), where Q is the total length of the scene path, and L_k is the length of the path traveled by the k th ant:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & arc(i, j) \text{ belongs to best tour} \\ 0, & \text{otherwise} \end{cases}. \quad (4)$$

2.2. ACO Algorithm Flow. The implementation steps of ACO algorithm are as follows:

- (1) In the initialization stage, the basic parameters required by the algorithm are generated $\alpha, \beta, \rho, \beta$ is the number of ants. N_c is the number of iterations. value between nodes $arc(i, j)$ initial pheromone concentration path $\tau_{ij}(0) = A$.
- (2) According to the probability calculated by formula (1), ants choose the position of the next node to move and update the corresponding $tabu_k$;

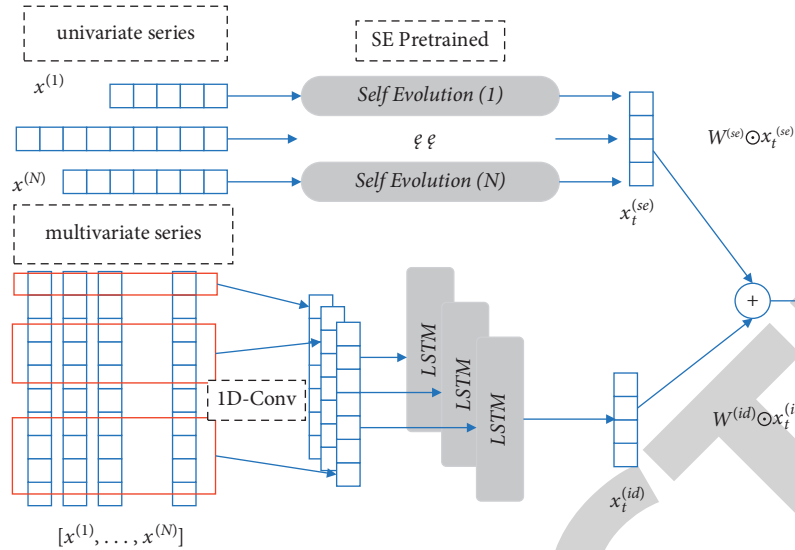


FIGURE 1: Ant colony time series diagram.

- (3) When ants complete a tour, they record the minimum cost value of the path they pass. If the value is better than the current optimal solution, the optimal solution is updated, in the case of confirming that the number of nodes passed through is the same, compare the value of the total length of the path that the ants often travel throughout the entire journey. The number of ants is m and select the next node according to the probability function to complete the propagation.
- (4) The number of pheromones left by ants on each path is calculated.
- (5) Repeat steps (3) to (4) until all ants have completed the whole propagation.
- (6) A cycle is completed, the pheromone concentration values deposited on the path are updated according to formulas (2)–(4).
- (7) When the algorithm reaches the specified termination times or reaches the set optimal error rate, the algorithm ends running and outputs the optimal solution.

The specific process is shown in Figure 2:

3. Improved Ant Colony Optimization Algorithm

3.1. Forgetting Factor Regulation Strategy. To solve the problems of local optimization, search stagnation, and slow initial convergence of traditional ant colony algorithm [19], the improved optimization algorithm adopts the local update method of adjusting pheromone and the global update strategy of pheromone, the forgetting factor can be used to determine the data distinction in the pheromone update process to keep records. Ants achieve a better balance in the process of searching and strengthening [20]. This algorithm has been tested on the known data [21]. When the error is

large, prefer a smaller forgetting factor to improve tracking sensitivity [22]. When the calculation error is small, the forgetting factor with a large value is preferentially selected to delay the memory scale and improve the recognition accuracy [23]. When comparing the forgetting factor, the number of times the forgetting factor is used is the main comparison standard. If the times are the same, the recorded path length is used as the comparison standard.

The core of the film and television communication problem is to solve the problem of correlation between nodes [24], as shown in Figure 3 for the node topology.

3.2. Initial Pheromone Setting. When the traditional ant colony algorithm constructs the solution, the pseudorandom factor is set as a constant, but the size of the pseudorandom factor often has a certain influence on the diversity and convergence speed of the algorithm in the early stage of the algorithm. When the pseudorandom factor is small, the ants will choose roulette with a high probability to construct the next solution, which increases the possibility of ants exploring unknown paths, thereby expanding the search space in the early stage and increasing the diversity of understanding, but it reduces the algorithmic complexity [25]. Conversely, when the pseudorandom factor is large, the ants will select the path according to the pseudorandom ratio with a large probability. Although the convergence speed of the algorithm is improved to a certain extent, if the previous ants fail to find a better path, due to the accumulation of pheromone, it will cause certain interference to the subsequent ant path construction, making the algorithm fall into a local optimum.

In the ant system (as) algorithm, the initial pheromone concentration is evenly distributed. In the initial stage of the algorithm, the ants conduct a large number of blind search and generate many invalid paths, resulting in inaccurate pheromone concentration update on the path, weakening the heuristic significance of pheromone indicating path. This

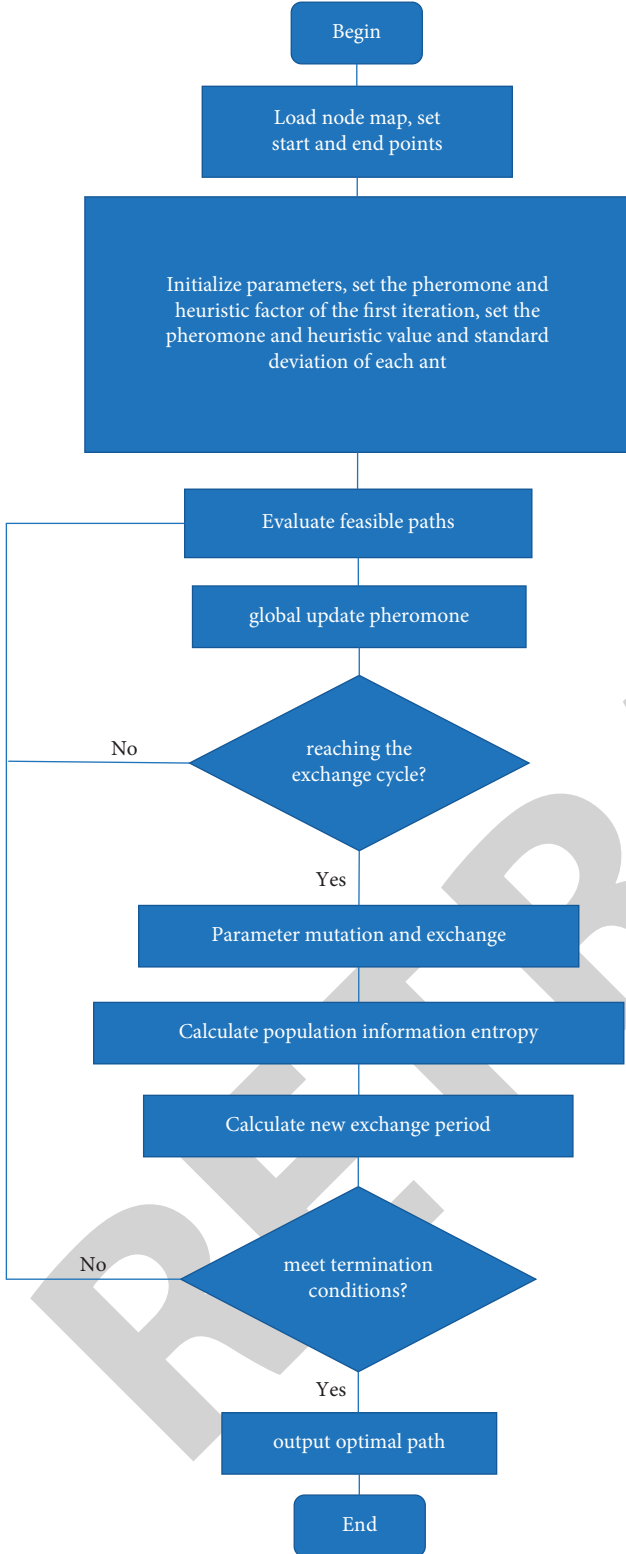


FIGURE 2: Ant colony algorithm execution flow chart.

problem not only makes the initial search time of the algorithm longer, but also due to the error of pheromone concentration updating, the search path falls into local optimum, which affects the performance of the algorithm. Inspired by the forgetting algorithm, this paper reduces the

value of forgetting factor in the early stage, that is, being able to find correlations from unknown things and refining the rules is the sign of true intelligence. Firstly, ant colony algorithm is used to find the information of suboptimal path, and then the value of forgetting factor is added to search for path information. A is the initial test pheromone matrix in the following equation:

$$A = \tau_0 * \text{ones}(N, N), \quad (5)$$

$$\tau_{ij}(0) = \begin{cases} 1/d_{ij} \\ l_N/n, i \neq j \\ 0, \text{ other} \end{cases}. \quad (6)$$

The initial pheromone is generated by $\tau_{ij}(0)$ in equation (6). It can realize the reasonable control of the initial pheromone concentration in the initial stage of the algorithm, which is conducive to the ability of ant colony algorithm to avoid choosing the path with high pheromone concentration when solving the FTTP and makes the algorithm easy to fall into the local optimal solution. It avoids searching on the nonoptimal path due to too many pheromones in ant colony system, and the pheromone concentration is too low to search for a long time. If the pheromone concentration is too low, subsequent ants will not choose this path. Even if this path may be the optimal path, it will be ignored because the pheromone concentration is too low.

3.3. Improved State Transition Strategy. Inspired by an improved distribution estimation algorithm, a forgetting factor is introduced to change the state transition rule of basic ant colony algorithm. When $E_{lp} \leq E_0$, according to formula (1), when each ant is sorted by the basic ant colony algorithm, the probability of moving to the next receptor node is given $E_{lp} > E_0$. The probability of ant selecting the next receptor node needs to be adjusted, and the probability of ant transferring to the next node needs to be adjusted $P_{newij}^k(t)$ as shown in the following equation:

$$P_{newij}^k(t) = \begin{cases} \text{argmax}_{j \in \text{allowed}_k}([\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta \cdot \lambda), E_{lp} > E_0 \\ P_{ij}^k(t), \text{ other} \end{cases}, \quad (7)$$

$$E_{lp} = \left| \frac{c_b - c_*}{c_*} \right|, \quad (8)$$

$$\lambda = 1 - E_{lp}. \quad (9)$$

As shown in equation (8), E_{lp} represents the optimal value error obtained by the second propagation of lp . E_{lp} represents the difference between the result and the ideal value of the algorithm. The smaller the size of E_{lp} , the better the algorithm performance. c_b represents the optimal value of the algorithm, and c_b is the ideal value. $\lambda \in (0, 1]$ comes from equation (9). If the optimal value of an ideal is unknown in the beginning, c_* can be represented by l_N as shown in the following equation:

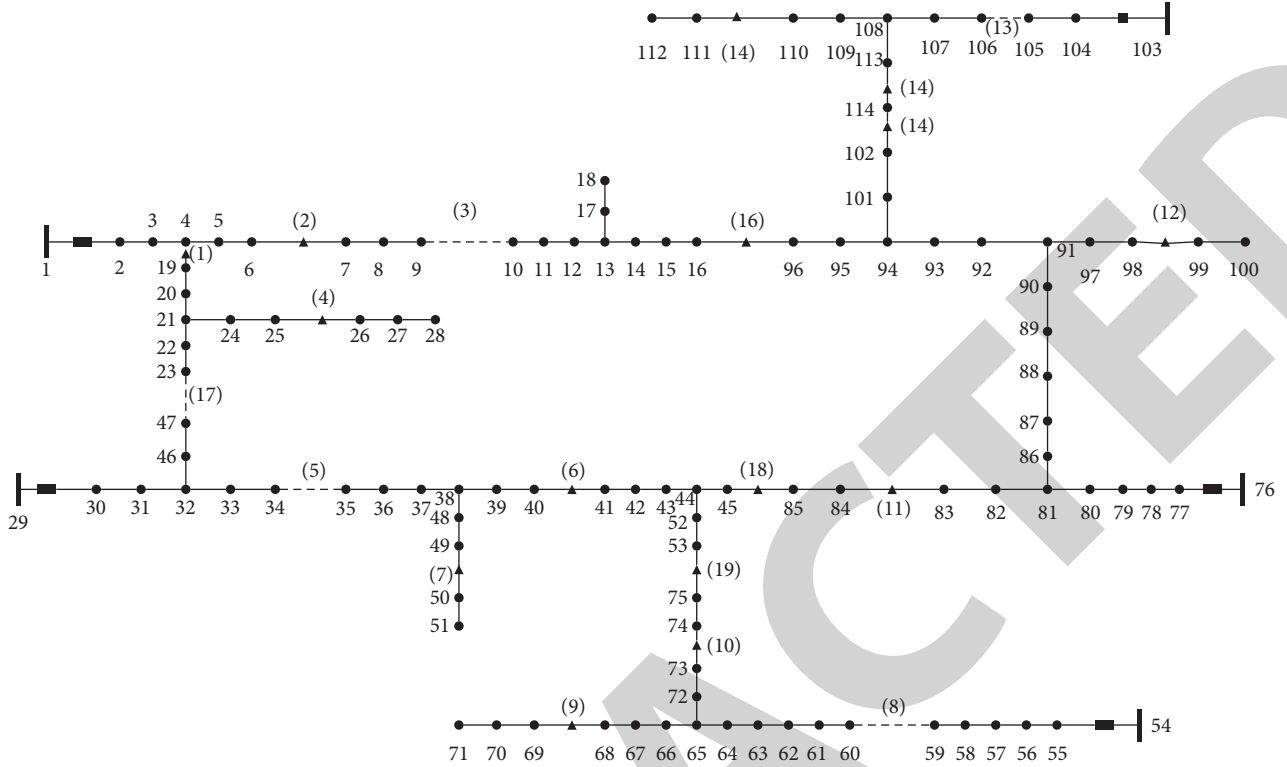


FIGURE 3: Video distribution node topology.

$$E_0 = \left\lfloor \frac{c_b - l_N}{l_N} \right\rfloor. \quad (10)$$

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + (1-\lambda) \cdot \tau_0, \lambda \in (0, 1), \quad (11)$$

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + (1-\lambda) \Delta \tau_{ij}(t) + \lambda \left(\frac{1}{L_{\text{best}}(n+1)} \right), \lambda \in (0, 1). \quad (12)$$

3.4. Improved Pheromone Updating Strategy. In the initial stage of the algorithm, the pheromone on all solutions is the same. With the advancement of the algorithm, the pheromone on the optimal solution will gradually increase, and the algorithm will gradually converge. But in the ant algorithm, ants always rely on the feedback information of other ants to reinforce learning without considering their own experience accumulation. Such blind obedience can easily lead to premature maturity and stagnation.

The updating rule of pheromone of each ant path is shown in equation (11). The initial path is constructed by the nearest neighbor method, first determine the current departure origin, search for the nearest nodes that have not been visited and repeat in turn, and finally return to the departure origin, then calculate τ_0 . In order to avoid the misleading effect of the global pheromone update strategy on the behavior of ants in the basic ant colony algorithm, the receptor node i and node j of τ_{ij} will be updated in time to improve the effective use of existing new information.

In order to enhance the feedback information of each optimal path to each ant, a new global pheromone-updating rule is proposed based on the traditional ant week model and elite strategy ant system. The rule can make full use of different information λ . Equation (12) is to add pheromones to the global optimal path:

First, FFACO, like the basic ant colony algorithm, updates the pheromone on the connection path on each receptor node after each iteration. After each iteration, the path contribution degree of the subpaths in the current global optimal path is judged. If the value of the path contribution degree is greater than the given threshold, it is determined as the optimal path, and the pheromone is updated again. The improved pheromone update strategy can increase the probability of the subpaths that constitute the potential optimal solution being selected and improve the convergence speed of the algorithm. At the same time, in order to speed up the algorithm's departure from the regional optimal solution, a local optimization method can be introduced to increase the randomness of the algorithm and the diversity of search results. If the path after mutation is better than the path before mutation, use the path after mutation to perform incremental update of subsequent pheromone; otherwise, continue to use the path before mutation to perform incremental update of subsequent pheromone. The realization process of mutation is shown in Figure 4. Two different nodes are randomly selected from the propagation path, the nodes between the two nodes are exchanged, and the mutation path is finally obtained.

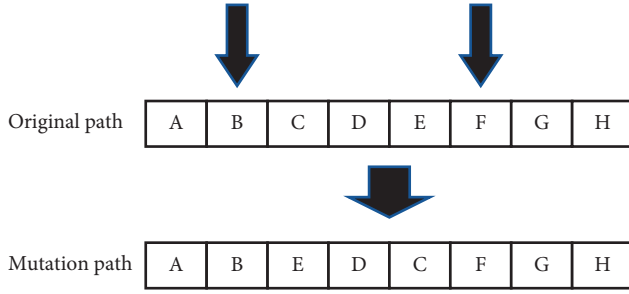


FIGURE 4: Ant colony propagation mutation process.

Because if the algorithm falls into the local optimal situation and deviates from the global optimal goal, the ants that follow the pheromone through the secondary path will get wrong path information, so when this situation occurs, the introduction of breeding mutation can make the original local. When the optimal situation is broken, the ants will recalculate the optimal solution to achieve the global optimal effect.

4. Experimental Environment and Results

4.1. Environment Experiment Setup. To analyze the performance of the improved algorithm, simulated annealing (SA), genetic algorithm (GA), particle swarm optimization (PSO), and ACO algorithms were tested on four different FTTP cases: chn31 test case and att48 test case datasets. The geographic coordinates of important nodes are recorded, and the ST70 and eil76 test cases, respectively, record the location and coordinate information of nodes. The SA, GA, PSO, ACO, and FFACO algorithms are applied to the four test cases, respectively. The simulation software MATLAB7 uses Intel(R) core(TM) i7-12700 CPU@4.90 GHz 12 cores and 12g running memory to conduct simulation experiments. The implementation of the algorithm will be introduced in detail below.

4.2. Algorithm Parameter Configuration

4.2.1. Parameter Configuration of SA. In the basic version of the simulated annealing algorithm, the initial temperature of SA is 120, the termination temperature is 1, and the cooling coefficient is 0.99. The operators applied to the exchange of propagating receptor nodes are shown in Table 1. The nodes that have been swapped are shown in bold in the table, and the maximum number of iterations is 200.

4.2.2. Parameter Configuration of Genetic Algorithm. The population size of genetic algorithm is 100 individuals. Starting from the nodes randomly selected by each individual, the nearest neighbor heuristic algorithm is used to generate the initial population. In the case of mutation, one individual is randomly selected from the population, and the mutation operation is carried out according to the relevant probability. In the case of recombination, the maximum algebra is 200.

TABLE 1: Two node random exchange operators.

2	6	8	7	5	3	1	9	4
2	7	8	6	5	3	1	9	4

4.2.3. Parameter Configuration of Particle Swarm Optimization Algorithm. The population size of particle swarm optimization algorithm is 100. The particle position is randomly initialized and the optimal path is searched circularly. Each particle searches the space separately; they remember the optimal solution they have found, and they also know the optimal solution currently found by the entire particle swarm. By calculating the population fitness value, the current optimal and the historical optimal values are updated. Combined with the cross-mutation operation, when the length of the new path is less than the historical optimal value, the update is accepted. In the case of crossover and mutation of each individual in the intermediate population, set the maximum number of evolutions to be 200.

4.2.4. Parameter Configuration of ACO Algorithm. The parameters of ACO algorithm are initialized to $\alpha = 1$, $\beta = 2$, $Q = 10$, $\rho = 0.2$, $m = 10$ and $N_c = 200$. The maximum number of iterations is 200, m is the number of ants, n is the number of nodes in this algorithm, α is the importance factor of the pheromone, β represents the important factor of heuristic function, ρ is represents the pheromone residue coefficient.

4.2.5. Parameter Configuration of FFACO Algorithm. The parameters of FFACO algorithm are initialized as follows: $\alpha = 1$, $\beta = 2$, $m = 10$, $\lambda = 0.1$, $\rho = 0.1$, and $N_c = 200$. N_c is the maximum number of iterations and its value is 200, α is important factors representing pheromones, β represents the important factor of heuristic function, m is the number of ants, λ is the forgetting factor, n is the number of nodes in the FTTP problem, and ρ represents the pheromone residue coefficient.

4.3. Experimental Results. By testing the time-consumption and the global optimal solution of the four data sets of TSPLIB, the superiority of the FFACO algorithm is fully demonstrated. In order to make the results more readable and easy to compare, the design of the five algorithms described (SA, GA, PSO, PSO, and FFACO) is as follows: each algorithm is tested on four public test cases. The average solution, the optimal solution, the worst solution, and the deviation rate from the known optimal solution is recorded from the results of 30 experiments. The simulation results of the four datasets are shown in Table 2, and the best known solution (BKS) of each test case is shown in Table 3. The number shown in the BKS row of Table 3 is the number of nodes of the best-known solution obtained by the four test cases in each algorithm environment, that is, the maximum number of nodes.

Table 2 shows the calculation results of each algorithm. The first column shows the various test cases, in ascending order of problem size; where the mean deviation of the

TABLE 2: Experimental calculation results.

Test cases	Algorithm	Avg	Dev (%)	Best	Dev (%)	Worst	Dev (%)
Chn31	SA	16988.515	10.480	15509.798	0.864	18590.447	20.898
	GA	15840.574	3.015	15380.515	0.023	17216.769	11.964
	PSO	16176.277	4.198	15380.515	0.023	17171.998	11.673
	ACO	15831.075	2.953	15601.920	1.463	16169.094	5.151
	FFACO	15505.948	0.839	15380.515	0.023	15890.953	3.342
Att48	SA	43947.775	21.101	39670.780	18.343	51922.190	54.890
	GA	40414.146	20.560	35879.259	7.032	45812.567	36.664
	PSO	41542.136	23.925	37802.644	12.770	44898.418	33.937
	ACO	36049.283	7.539	35002.329	4.416	36713.262	9.520
	FFACO	34155.483	1.890	33600.561	0.234	34994.793	4.394
St70	SA	1506.879	123.241	1320.669	95.655	1667.052	146.971
	GA	1189.746	76.259	1031.540	52.821	1417.894	110.058
	PSO	1241.762	83.965	1021.708	51.364	1391.826	106.196
	ACO	1310.852	94.200	1288.044	90.821	1330.420	97.099
	FFACO	699.366	3.610	681.755	1.001	740.408	9.690
Eil76	SA	1528.475	184.103	1392.716	158.869	1674.280	211.204
	GA	1144.966	112.819	983.099	82.732	1264.024	134.949
	PSO	969.793	80.259	862.578	60.330	1058.072	96.668
	ACO	576.657	7.185	565.447	5.143	585.340	8.799
	FFACO	569.427	5.842	555.003	0.929	587.595	9.218

TABLE 3: Optimal solution of benchmark example.

Test cases	Chn31	Att48	St70	Eil76
BKS	15377	33522	675	538

second solution and its second column are the mean deviation of the second solution, the best and worst solutions were calculated in this way from 30 obtained from the results of the runs. Only the first 200 evolution results of the algorithm are taken for each run, The change trend of the subsequent operation is similar to that of the first 200 times, and the change is relatively gentle. The calculation results of each algorithm show its true value and deviation from the optimal value. The best value results for each dataset in the table are shown in bold. Due to the introduction of the forgetting factor, the FFACO algorithm reduces the situation of falling into the local optimum during the execution process and can calculate the global optimum faster.

For the test case of chn31, the results of FFACO are very good. For the propagation case of 48 nodes, FFACO can still find the closest optimal solution (dev. 0.234%), while the deviation rate of SA, GA, PSO, and ACO optimal solutions is much higher than that of FFACO, the average and worst solutions are the bias is also much higher than FFACO.

Overall, FFACO performed the best in all of the above test cases over 200 evolutions of the optimal solution. Among the 12 optimization results in Table 3, 11 are FFACO, and the improved algorithm greatly improves the performance of the ant colony algorithm. For test cases with different problem sizes, SA, GA, and PSO require a large number of iterations to optimize the optimal path, which is the main reason why their performance is inferior to FFACO. For small problems, ACO and FFACO have significant differences in mean solution deviation, optimal solution deviation, and worst solution deviation. As the

problem size increases, the results of ACO and FFACO become better than other procedures. The advantage of ACO is that it can get better results with each iteration and generate new solutions based on the learned data, but FFACO has better performance than ant colony algorithm in training optimal and mean values. Computational results show that the proposed improvement is effective for TSP and compared with SA, GA, PSO, and ACO.

4.4. Result Analysis. In order to enhance the comparison of the ability of solving the optimal solution and analyze the effectiveness of the algorithm in this paper, the evolutionary trajectory of the optimal solution of SA, GA, PSO, ACO algorithm, and the algorithm proposed in this paper are compared. The experimental results are shown in Figures 5(a)–5(d). It can be seen from the above that both the improved algorithm and ACO algorithm can quickly approach the optimal solution at the initial stage of iteration, while SA, GA, and PSO algorithms are obviously behind them. Moreover, the improved algorithm can still further optimize the optimal solution when the ACO is in a stagnant state. It shows its good tracking ability to the optimal solution. It can be seen from Figure 5(d) that the improved algorithm can quickly approach the optimal solution when solving the eil76 test case, and the evolution curve of the optimal solution of other algorithms is obviously different from that of other algorithms.

From the perspective of the problem scale, the scales of the four test cases chn31, att48, ST70, and eil76 increase in turn. In the limited number of iterations, FFACO and ACO algorithms have better solving ability, but compared with ACO algorithm, FFACO has better adaptability to solve FTTP of different scales. In terms of the quality of the solution, the proposed FFACO algorithm will update the optimal solution in each iteration. Without too much

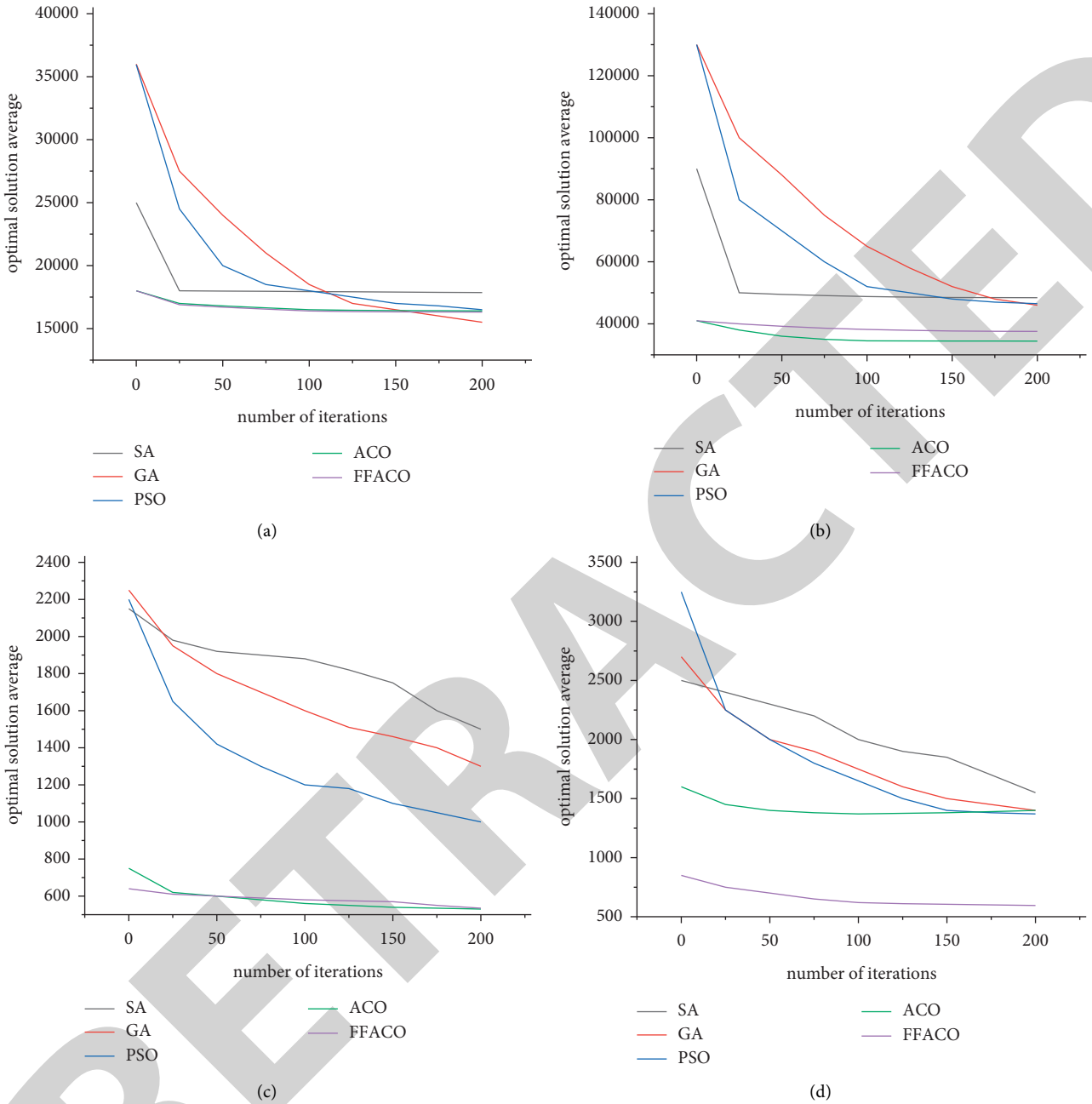


FIGURE 5: Algorithm iteration results comparison. (a) Evolution trajectory of optimal solution (example chn31). (b) Evolution trajectory of optimal solution (example att48). (c) Optimal solution of ST3. (d) Evolution trajectory of optimal solution (example eil76).

consideration of the size of the problem to be solved, the whole solving process is relatively stable, which effectively improves the performance of the ant colony algorithm.

It can be seen from Table 4 that the optimal solution is time consumption of the five algorithms in solving different test cases, and the bold font is the optimal value. Compared with SA, GA, and PSO algorithms, FFACO algorithm has obvious advantages in finite iteration calculation. Compared with ACO algorithm, FFACO algorithm has better ability to jump out of local optimal solution. It can track the optimal solution well and reduce the time consumption. For different test cases, the size of the required solution increases in turn,

and the time for each algorithm to find the optimal solution also increases. However, FFACO algorithm is still the most time-saving in finding the optimal solution.

Figure 6 is the global optimal solution path-planning diagram of the proposed FFACO algorithm for the test case. The abscissa and ordinate shown in the figure are the description of the node location coordinates. The serial numbers 1–31 in Figure 6 are the locations of 31 nodes and their serial numbers. The roadmap here is the route that starts from node 1 and is calculated by the FFACO algorithm, passing through all nodes and returning to node 1. optimal transmission path.

TABLE 4: Time consumption of algorithm for optimal solution.

Test cases	Algorithm	Time (s)
Chn31	SA	5.632
	GA	5.489
	PSO	1.414
	ACO	0.807
	FFACO	0.577
Att48	SA	5.986
	GA	4.948
	PSO	1.686
	ACO	1.403
	FFACO	0.955
St70	SA	7.918
	GA	4.251
	PSO	2.139
	ACO	2.504
	FFACO	1.497
Eil76	SA	8.404
	GA	4.640
	PSO	2.649
	ACO	2.868
	FFACO	1.666

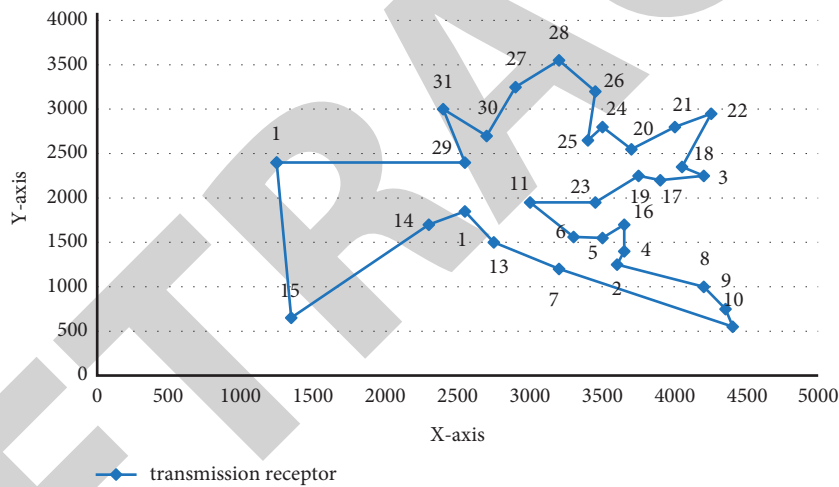


FIGURE 6: Global optimal solution route (example chn31).

Figure 7 is the global optimal solution path-planning diagram of the Att48 test instance, and the coordinates in the figure are the coordinate descriptions of the nodes.

The trend curve of the algorithm objective function value is shown in Figure 8. The conventional ant colony algorithm converges after more iterations and falls into the local optimum; the improved ant colony algorithm finds the global optimal solution with only a few iterations, and the convergence rate is fast, which verifies the effectiveness of the improved algorithm.

To sum up, compared with ACO algorithm, FFACO algorithm proposed in this paper has better solution ability and better adaptability to FTTP of different sizes. At the

same time, in terms of 3D path planning, FFACO can record more path selection situations beyond the plane due to the introduction of the forgetting factor, which ensures that the path planning ability is still accurate in the 3D scale, which is greatly superior to the ACO algorithm. Compared with SA, GA, and PSO algorithms, FFACO algorithm significantly saves the time cost of finding the optimal solution. The algorithm proposed in this paper can effectively improve the premature stagnation phenomenon in the process of basic ant colony convergence. The whole solution process is relatively stable and effectively improves the solution quality of ant colony algorithm, which makes the improved algorithm have better robustness and convergence.

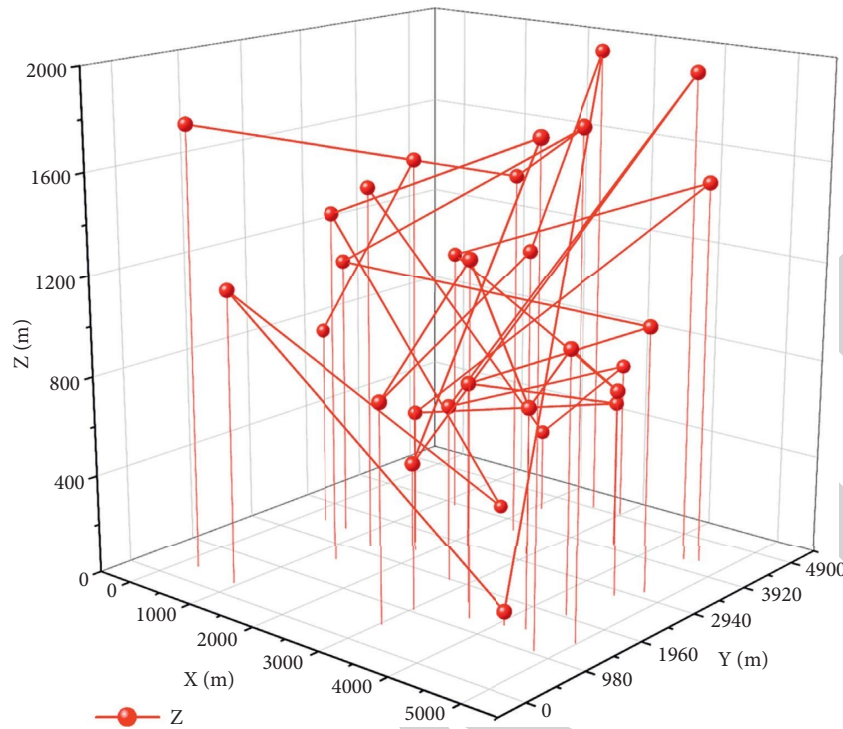


FIGURE 7: Global optimal solution route (example Att48).

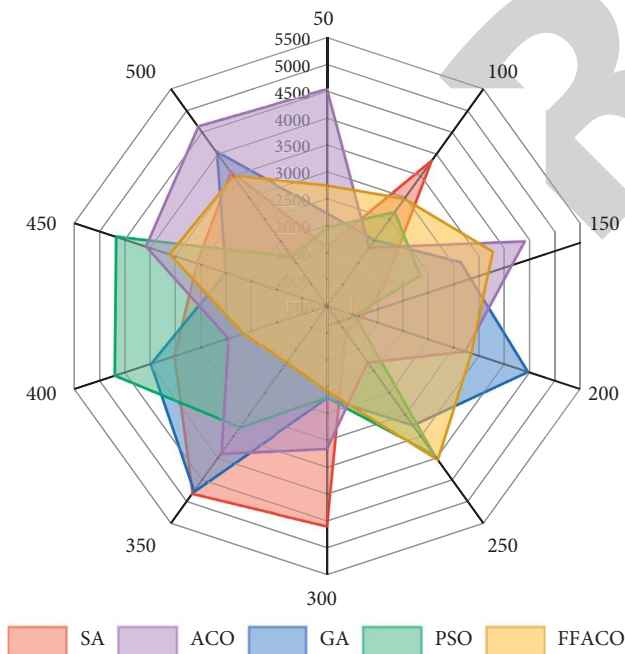


FIGURE 8: Algorithm comparison chart.

5. Conclusion

This paper proposes an ant colony algorithm with a forgetting factor and an improved pheromone update model. The main idea is to introduce a forgetting factor in the ants of the basic ant colony algorithm. The forgetting factor can modify the weight of the local pheromone in time and update the global pheromone. The adjustment of the

forgetting factor can achieve a good tracking effect on pheromone and improve the stagnation behavior and convergence of the basic ant colony algorithm. For the film and television communication scene, the local optimal situation in the information transmission process can be well avoided, that is, the transmission effect is very good in a certain area, but the transmission of a large coverage cannot be achieved. Second, the optimized algorithm can better ensure the film and television. Information spreads the widest in the shortest time. Finally, the experimental results show that the algorithm has a good ability to find the optimal solution, and the time cost is low. Future work will focus on theoretical evaluation of the validity of the path evaluation model.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding this work.

References

- [1] Q. Y. Wang, G. Q. Zhou, R. H. Song, Y. Xie, M. Luo, and T. Yue, "Continuous space ant colony algorithm for automatic selection of orthophoto mosaic seamline network," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 186, pp. 201–217, 2022.

- [2] J. Peng, Y. B. Li, H. W. Kang, Y. Shen, X. Sun, and Q. Chen, "Impact of population topology on particle swarm optimization and its variants: an information propagation perspective," *Swarm and Evolutionary Computation*, vol. 69, pp. 100990–101135, 2022.
- [3] H. L. Ma, "Methods for solving combinatorial optimization problems: a review," *International Core Journal of Engineering*, vol. 7, no. 11, pp. 15–26, 2021.
- [4] E. V. Ivohin, O. F. Voloshyn, and M. F. Makhno, "Modeling of information dissemination processes based on diffusion equations with fuzzy time accounting," *Cybernetics and Systems Analysis*, vol. 57, no. 6, pp. 896–905, 2021.
- [5] J. H. Wu and Q. Y. Li, "Method for forecasting urban national sports and fitness demand based on ant colony algorithm," *Computational Intelligence and Neuroscience*, pp. 1–7, 2021.
- [6] S. D. Li, T. Y. Luo, and L. Wang, "Tourism route optimization based on improved knowledge ant colony algorithm," *Complex & Intelligent Systems*, pp. 19–31, 2022.
- [7] Y. X. Tong and X. H. Li, "Ant colony algorithm for time optimal load balancing scheduling problem strategy," *Journal of Physics: Conference Series*, vol. 2010, no. 1, pp. 66–80, 2021.
- [8] L. Yun and H. Lu, "Diagnosis strategy design method based on ant colony algorithm," *Journal of Physics: Conference Series*, vol. 1894, no. 1, pp. 72–89, 2021.
- [9] J. Xu, "A modified AODV routing protocol using in WSN based on ant colony algorithm," in *Proceedings of the 2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT 2021)*, pp. 123–126, Sanya, China, December 2021.
- [10] C. L. Li, K. Jiang, and Y. L. Luo, "Dynamic placement of multiple controllers based on SDN and allocation of computational resources based on heuristic ant colony algorithm," *Knowledge-Based Systems*, vol. 241, pp. 111–135, 2022.
- [11] J. B. Zhao, X. M. You, Q. Duan, and S. Liu, "Multiple ant colony algorithm combining community relationship network," *Arabian Journal for Science and Engineering*, vol. 47, pp. 10531–10546, 2022.
- [12] C. W. He and J. Mao, "AGV optimal path planning based on improved ant colony algorithm," *MATEC Web of Conferences*, vol. 232, p. 03052, 2018.
- [13] X. L. Zhang and Q. Zhang, "Optimization of PID parameters based on ant colony algorithm," in *Proceedings of the 2021 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS 2021)*, pp. 265–268, Xi'an, China, March 2021.
- [14] G. H. Song, Y. B. Wang, and Y. G. Li, "Dynamic mathematical model of information spreading on news platform," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 65–79, Article ID 2174190, 2021.
- [15] Q. K. Zhu and M. Priyan, "Computer application in game map path-finding based on fuzzy logic dynamic hierarchical ant colony algorithm," *International Journal of Fuzzy Systems*, vol. 24, pp. 2513–2524, 2021.
- [16] W. Yan, "Application of ant colony algorithm and artificial intelligence in training simulation of athletes in sports arena," *Journal of Ambient Intelligence and Humanized Computing*, pp. 61–79, 2021.
- [17] M. Mukesh and P. S. Om, "Generating and prioritizing optimal paths using ant colony optimization," *Computational Ecology and Software*, vol. 5, no. 1, pp. 65–77, 2015.
- [18] W. Bo, F. Tong, and D. Y. Huang, "Mobile edge computing for LEO satellite: A computation offloading strategy based improved ant colony algorithm," in *Proceedings of the 11th International Conference on Computer Engineering and Networks (CENet2021)*, pp. 818–830, Singapore, November 2021.
- [19] J. Wu, M. J. Fan, Y. Liu, Y. Zhou, N. Yang, and M. Yin, "A hybrid ant colony algorithm for the winner determination problem," *Mathematical Biosciences and Engineering*, vol. 19, no. 3, pp. 3202–3222, 2022.
- [20] A. Nithya, K. Amudha, F. Musthafa et al., "An optimized fuzzy based ant colony algorithm for 5G-MANET," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 81–99, 2022.
- [21] Q. Q. Li, "Research on optimization of mesh routing based on greedy particle ant colony algorithm," *Scientific Journal of Intelligent Systems Research*, vol. 4, no. 1, pp. 41–55, 2022.
- [22] E. M. Ekanayake, "Novel method for solving balance and unbalance assignment problem using new approach for ant colony algorithm," *Mathematical Modelling and Applications*, vol. 6, no. 4, pp. 33–47, 2021.
- [23] X. Ziang, J. S. Du, Q. T. Chen, and X. Wang, "Enhancing the labor division in the balancing of apparel assembly lines with parallel workstation through an improved ant colony algorithm," *Journal of Engineered Fibers and Fabrics*, vol. 16, pp. 16–30, 2021.
- [24] F. Wu and D. He, "Contactless distribution path optimization based on improved ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–11, Article ID 5517778, 2021.
- [25] Z. D. Wang, C. H. Wu, and J. Xu, "Research on path planning of cleaning robot based on an improved ant colony algorithm," *MATEC Web of Conferences*, vol. 336, pp. 54–70, 2021.