Application of Ant Colony Algorithm in Selecting Shortest Engineering Path

Jinsong Tian, Manyu Chen*, Chao Chen, Xin Liu
School of Science, Anhui Agricultural University, Hefei, Anhui 230036, China
*Corresponding author’s e-mail: anhuiceliang@sina.com

Abstract: In engineering surveying, topographic maps can be used to find the shortest path. Traditional methods identify the smallest radius based on the design slope before finding the shortest path. This paper, by using the ant colony algorithm and MATLAB software, manages to select the shortest path in engineering practice, and compares the result with that obtained using the genetic algorithm. The results show that the ant colony algorithm can be applied in selecting the shortest path in engineering projects and it can deliver a stable and optimal result after 240 times of iteration. However, the genetic algorithm cannot achieve that even after 400 times of iteration. Therefore, the ant colony algorithm is superior to the genetic algorithm in the aspect of the calculation speed, the stability of achieving the optimal solution, and selection of the shortest path.

1. Introduction
For the path planning and design in engineering projects such as roads, railways, pipelines, and communications, it is often necessary to plan an optimal path connecting the starting point with the target point [1]. This paper, based on the ant colony algorithm and MATLAB software, manages to select the shortest path in engineering, and compares the result with that obtained using the genetic algorithm [2].

The results show that the ant colony algorithm can be applied in selecting the shortest path in engineering projects and it can deliver a stable and optimal result after 240 times of iteration. However, the genetic algorithm cannot achieve that even after 400 times of iteration.

2. Ant colony algorithm
The ant colony algorithm is derived from the study of ant foraging. It is found that ants release a pheromone on their way of food searching, and its concentration is positively correlated with the length of the path. Other ants will choose the path with higher pheromone concentration and also leave more pheromone on their way, which can update the pheromone and help to select a shortest path [3-5].

2.1 Basic principles
The algorithm flow based on MATLAB design is shown in Figure 1.
2.2 Three-dimensional modeling of planning area

The primary condition for implementing the path planning algorithm is the conversion of the topographic map of the planning area into a 3D model [4]. Here are the specific steps:

1. Set the inflection point in the lower left corner of the topographic map as the origin point o, and take the north-south direction as the X-axis, the east-west direction as Y-axis, and the zenith direction as the Z-axis. The maximum lengths in the X, Y, and Z directions are OA, OB, and OC, respectively, as shown in Figure 2.

2. Discretize the space after establishing the three-dimensional coordinate system. First, equally divide the 3D space along the OA direction into $s \times n+1$ planes ($i=1, 2, \ldots, n+1$), and then equally divide each plane along the OB and OC directions by $m$ and $l$. The more the divided spaces, the higher the similarity between the sets after discretion and the original one, as shown in Figure 3.

3. Through spatial discretization, the points in the planning area also correspond to the coordinates in the three-dimensional model, $a(i, j, k)(i = 1, 2, \ldots, n, j = 1, 2, \ldots, m, k = 1, 2, \ldots, l)$, where $i$, $j$, and $k$ are the division numbers of point a on OA, OB, and OC, respectively.

2.3 Pheromone update

The key of the ant colony algorithm is the pheromone update of each discrete point $a_{ij}$ after the pass of ants. The pheromone concentration affects the probability of subsequent ants passing the point [6]. The pheromone update can be divided into local update and global update.

The function of local update is to increase the probability of an ant passing a point that has never been passed through before. When an ant passes a point $a_{ij}$, its pheromone value decreases according to the following equation:

$$
\tau_{ijk} = (1 - \xi) \tau_{ijk}
$$

In Equation (1), $\tau_{ijk}$ refers to the pheromone value stored at the point $a_{ij}(i, j, k)$; $\xi$ refers to the attenuation coefficient of the pheromone.
Global update refers to the path evaluation and shortest path selection after the path is completed. The pheromone values of the nodes are increased to enhance the probability of subsequent passing. The equation for global update is:

$$\tau_{ijk} = (1 - \rho) \tau_{ijk} + \rho \Delta \tau_{ijk}$$ (2)

$$\Delta \tau_{ijk} = \frac{K}{\min \{\text{length}(m)\}}$$ (3)

In Equations (2) and (3), $$\rho$$ is the coefficient for pheromone update; length (m) is the length of the route through which the $$m$$th ant passes; K is a constant number.

2.4 Heuristic function

When the ant moves in the three-dimensional model, we can set the current point as a, the next point as b, and the target point as c. In this way, a heuristic function for calculating the probability of the ant moving to each point can be set as:

$$H(i, j, k) = D(i, j, k)^{w1} \times Q(i, j, k)^{w2} \times S$$ (4)

In Equation (4), S is a safe value, and is taken as 0 or 1 to ensure that the selected path meets the design requirements; w1 and w2 are weight coefficients, representing the importance of D(i, j, k) and Q(i, j, k) in the heuristic function. D(i, j, k) is the distance between the current point and the next one, which urges the ant to select the shortest path. Q(i, j, k) is the distance from the next point to the target point, which urges the ant to select a point closer to the target.

$$D(i, j, k) = \sqrt{(x_c - x_b)^2 + (y_c - y_b)^2 + (z_c - z_b)^2}$$ (5)

$$Q(i, j, k) = \sqrt{(x_b - x_0)^2 + (y_b - y_0)^2 + (z_b - z_0)^2}$$ (6)

2.5 Searching method

Set the x-axis direction as the main direction in which the ant advances, which is to regard the movement of the ant in the planning area as moving from the plane where the current point $$p_i$$ is located to the point $$p_{i+1}$$ on the next plane $$\alpha_{i+1}$$. The selection method of the specific position of point $$p_{i+1}$$ in the plane $$\alpha_{i+1}$$ is as follows:

1. Calculate the number of all safety points in the plane $$\alpha_{i+1}$$, and the selection is based on the design requirements.
2. Calculate heuristic information values of all possible safety points from the point $$p_i$$ to the plane $$\alpha_{i+1}$$ according to the heuristic function in mentioned in 2.1.3.
3. Calculate $$p(i + 1, y_b, z_b)$$, the probability of selecting any point in the plane. The calculation model is.

$$p(i + 1, y_b, z_b) = \frac{\tau_{i+1, y_b, z_b} H(i, y_b, z_b)}{\sum \tau_{i+1, y_q, z_q} H(i, y_q, z_q)}$$ (7)

In Equation (7), $$\tau_{i+1, y_b, z_b}$$ is the pheromone value of the point $$p_{i+1, y_b, z_b}$$ in the plane $$\alpha_{i+1}$$. The probability of choosing an unreachable non-safe point is zero.

4. The specific position of the next point is obtained based on the calculated probability. Repeat the above steps until the final point is reached.

3. Verification

This paper takes the 1:500 digital topographic maps (CGCS2000 coordinate system) of Anhui Agricultural University High-tech Agricultural Park as an example, and uses the ant colony algorithm and the MATLAB programming to automatically search for the shortest path from the starting point to
the end point. In order to facilitate calculation, this paper takes the lowest point elevation in the planning area as 0, and determines the other points by the height difference from the lowest point; the X and Y coordinates on the topographic map correspond to the serial number coordinates in the model.

### 3.1 MATLAB programming

The steps to write a program in MATLAB are as follows:

1. Input the coordinates of the elevation points into the x, y, z matrix, and form a three-dimensional image. To make calculation easier, this paper first divides the planning area into 20*20 grids, and forms a 21*21 elevation matrix based on the elevation of each grid point (rounded to one decimal place), and then obtains an initial three-dimensional model after converting the measured coordinates of each point into the coordinate in the algorithm.

2. In order to make the algorithm faster and more accurate, this paper sets the initial parameters as follows. The size of ant colony is set as 20; the pheromone update coefficient $\rho = 0.2$; the number of algorithm iteration, 400 times; the starting point, (1, 10, 4.2); the ending point, (21, 4, 5.4). The safety value S is determined based on the height difference and the horizontal movement.

3. Calculate the heuristic value of the ant moving from the current point to the next one based on the above formulas. Then, search all 3D paths from the starting point to the end based on the pheromone and the heuristic value, and calculate the fitness value of each path. Finally, seek the optimal solution using the ant colony algorithm iteration.

4. Output the final optimal path and the graph of fitness change of the optimal individual, and evaluate the ant colony algorithm.

### 3.2 Program operation and graph drawing

After importing elevation points into a computer 3D model, initialize the relevant parameters and update the pheromone at the starting point before the algorithm starts, and then judge the safety value of the next point using the if function. In this way, all possible next points can be obtained.

After updating the pheromone at the starting point and judging the security value of the heuristic function, we can start to iteratively search for the shortest path, which is the core part of the algorithm. When each ant advances from the starting point to the next point according to the probability $p(i + 1, y_b, z_b)$, record the current shortest path of each iteration, update the pheromone of the nodes, and calculate optimal individual fitness value corresponding to the number of times of the current iteration. Through multiple fitting, the optimal path can be obtained.

Finally, simulate and output the shortest path through a 2D model diagram (Figure 4) and a 3D model diagram (Figure 5) so as to facilitate engineering planning and application.

### 3.3 Comparison of ant colony algorithm and genetic algorithm

The ratio of the algorithm fitness value to the number of iterations is the reflection of the convergence speed. When the fitness value tends to be stable, the possibility of the algorithm optimization reduces,
and the optimal solution will be obtained [8]. From Figure 6(a), it can be seen that as the number of iterations increases, the optimal individual fitness value of the ant colony algorithm decreases continuously, and the declining is the fastest during the 0 to 50th iteration. When iterations is over more than 240 times, the fitness value no longer decreases, indicating that the algorithm results tend to be stable and the shortest path is obtained [9].

This paper refers to the genetic algorithm for comparison [10–12]. 400 times of iteration using the genetic algorithm and the same data delivers the variation tendency of the optimal individual fitness value, as shown in Figure 6(b). From Fig. 6(a) and (b), it can be clearly seen that the declining speed of the variation trend of the fitness value obtained using genetic algorithm is slower than that obtained using the ant colony algorithm. Besides, after 400 times of iteration, the fitness value obtained using the genetic algorithm still keeps decreasing, suggesting that the optimal solution is still not obtained. Therefore, the ant colony algorithm is superior to genetic algorithm in terms of calculation speed and stability.

![Image](image.png)

(a). the ant colony algorithm  
(b). the genetic algorithm

Figure 6. Variation of adaption degree using the ant colony algorithm and the genetic algorithm

4. Conclusion and prospects
This paper, based on MATLAB program, complete the iterative optimization of the path with good efficiency and stability, using ant colony algorithm, following the steps of heuristic function calculation and pheromone update. The result is shown in the form of image, thus improving its practical value. However, the algorithm may still have a deceptive problem. Subsequent research can further optimize the selection of fitness functions and other bionic algorithms. In conclusion, ant colony algorithm provides a new idea for finding the shortest path in engineering measurement.

Acknowledgments
Research on Practical Teaching of Surveying and Mapping Engineering Based on Discipline Competition (The Teaching Research Project of Anhui Agricultural University, Grant No 2018aujyxm047).

References
[1] Zhang, Z.L. (2013) Engineering Surveying. Wuhan University Press, Wuhan.
[2] Shi, F. (2011) Analysis of 30 Cases of Intelligent Algorithms in MATLAB, Beijing University of Aeronautics and Astronautics Press, Beijing.
[3] Hua, L. (2011) Research on Path Planning Based on Ant Colony Algorithms, Nanchang Aviation University, Nanchang.
[4] Zhao, J.P. (2012) Ant Colony Optimization for Mobile Robot Path Planning, Northeastern University, Shenyang.
[5] Wang, J.H. (2017) Ant Colony Algorithm Based on Improved Initial Pheromone and Public Path,
[6] Wang, X. (2005) The New Method and Simulation of Path Planning Based on Ant Colony Algorithm, Computer Simulation, (7):60-62.
[7] Ou, C.J. (2013) Application of Ant Colony Algorithms in GIS, Surveying & Mapping and Spatial Geographic Information, (8):187-188.
[9] Tian, J.S. (2012) Parameter Estimation of Measurement Adjustment Based on Artificial Fish Swarm Algorithms, Urban Geotechnical Investigation & Surveying, 43(2):125-130.
[10] Li, J. (2013) Research on Robot Path Planning Based on Ant Colony Algorithms and Genetic Algorithms, Harbin Institute of Technology, Harbin.
[11] Duan, J.T. (2017) Application of Evolutionary Computing Algorithms in Path Optimization, Electronic Technology and Software Engineering, (10):146-150.
[12] Hao, B. (2010) Research on Path Planning Method of Mobile Robot Based on Improved Genetic Algorithms, Computer Engineering and Science, (7):104-107.