agriculture

MDPI

Article

Three-Dimensional Obstacle Avoidance Harvesting Path Planning Method for Apple-Harvesting Robot Based on Improved Ant Colony Algorithm

Bin Yan [1,2,3,*], Jianglin Quan [1,3] and Wenhui Yan [2,4]

[1] College of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China
[2] College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, China
[3] Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an University of Technology, Xi'an 710048, China
[4] College of Mechanical Engineering, Xi'an Shiyou University, Xi'an 710065, China
* Correspondence: yanbin@nwafu.edu.cn

Abstract: The cultivation model for spindle-shaped apple trees is widely used in modern standard apple orchards worldwide and represents the direction of modern apple industry development. However, without an effective obstacle avoidance path, the robotic arm is prone to collision with obstacles such as fruit tree branches during the picking process, which may damage fruits and branches and even affect the healthy growth of fruit trees. To address the above issues, a three-dimensional path -planning algorithm for full-field fruit obstacle avoidance harvesting for spindle-shaped fruit trees, which are widely planted in modern apple orchards, is proposed in this study. Firstly, based on three typical tree structures of spindle-shaped apple trees (free spindle, high spindle, and slender spindle), a three-dimensional spatial model of fruit tree branches was established. Secondly, based on the grid environment representation method, an obstacle map of the apple tree model was established. Then, the initial pheromones were improved by non-uniform distribution on the basis of the original ant colony algorithm. Furthermore, the updating rules of pheromones were improved, and a biomimetic optimization mechanism was integrated with the beetle antenna algorithm to improve the speed and stability of path searching. Finally, the planned path was smoothed using a cubic B-spline curve to make the path smoother and avoid unnecessary pauses or turns during the harvesting process of the robotic arm. Based on the proposed improved ACO algorithm (ant colony optimization algorithm), obstacle avoidance 3D path planning simulation experiments were conducted for three types of spindle-shaped apple trees. The results showed that the success rates of obstacle avoidance path planning were higher than 96%, 86%, and 92% for free-spindle-shaped, high-spindle-shaped, and slender-spindle-shaped trees, respectively. Compared with traditional ant colony algorithms, the average planning time was decreased by 49.38%, 46.33%, and 51.03%, respectively. The proposed improved algorithm can effectively achieve three-dimensional path planning for obstacle avoidance picking, thereby providing technical support for the development of intelligent apple picking robots.

Keywords: apple tree; ant colony algorithm (ACO); 3D path planning; biomimetic optimization; picking robot

Citation: Yan, B.; Quan, J.; Yan, W. Three-Dimensional Obstacle Avoidance Harvesting Path Planning Method for Apple-Harvesting Robot Based on Improved Ant Colony Algorithm. *Agriculture* **2024**, *14*, 1336. https://doi.org/10.3390/agriculture14081336

Academic Editor: Bruno Bernardi

Received: 4 July 2024
Revised: 5 August 2024
Accepted: 8 August 2024
Published: 10 August 2024

1. Introduction

Robot fruit picking [1–12] has become a worldwide research and application topic, and it is particularly urgent to conduct in-depth research on ensuring the success rate and efficiency of robot fruit picking. Apple orchards can be mainly divided into two categories based on the planting mode of fruit trees: traditional closed apple orchards and modern apple orchards (also named standard apple orchards). Traditional closed apple orchards (shown in Figure 1a) have intersecting branches and low and enclosed rows, resulting in poor mechanical passability in the orchard. Therefore, they are not suitable for robot

harvesting operations. When planting fruit trees in modern apple orchards, the dwarf rootstock dense planting mode is widely utilized. It has multiple advantages such as short tree crowns, convenient fruit tree management, early fruit-bearing, high apple yields, good quality fruit, and easy, intelligent operation by robots. Using this mode, it is easy to achieve standardization and scale management; thus, it is widely adopted by advanced apple-producing countries in the world. It is also the direction of modern apple industry development. Compared with several other modern apple orchard cultivation modes such as 'V', 'Y', and 'wall', the spindle cultivation mode has more fruiting branches and is more advantageous in fruit yield, making it widely used in modern standard orchards. Modern orchard spindle-shaped apple trees are shown in Figure 1b.

(**a**) (**b**)

Figure 1. Pictures of traditional closed apple orchard fruit trees (**a**) and cultivation model for modern spindle-shaped apple trees in orchards (**b**).

However, the complex distribution of internal branches in the crowns of apple trees is a challenge. The branches of fruit trees cause obstacles in picking paths when robot picking arms move to the position of the fruit to be picked. This problem is still a core difficulty that needs to be solved urgently. Therefore, planning an efficient intelligent obstacle avoidance fruit picking path is essential to ensure the success rate of robot picking and improve operational efficiency. Up to now, many researchers have conducted research on path planning algorithms for robot fruit picking. In terms of obstacle avoidance path planning for apple-picking robots, relevant research from recent years is shown in Table 1.

Table 1. Research on obstacle avoidance picking path planning for apple-harvesting robots.

Path-Planning Algorithm	For Modern Orchard Tree Shapes from a Broad Perspective	Year	3D Path Planning	References
Informed RRT*	N	2024	Y	[13]
Improved RRT	N	2023	N	[14]
Optimal sequential ant colony optimization algorithm (OSACO)	N	2023	Y	[15]
Artificial potential field and A* algorithm	N	2023	N	[16]
Biological stimulation neural network	N	2022	N	[17]
Improved particle swarm optimization (PSO)	N	2023	Y	[18]
Improved rapidly exploring random tree (RRT)	N	2024	Y	[19]
Deep deterministic policy gradient algorithm (DDPG)	N	2020	Y	[20]
Cauchy target gravitational bidirectional RRT* (CTB-RRT*)	N	2021	Y	[21]

Y: yes; N: no.

Despite the current research status of robot apple obstacle avoidance picking path planning algorithms, most existing algorithms have not been designed for standard orchard apple tree shapes. Therefore, the effectiveness of applying them to modern apple orchard trees that are widely planted still needs further verification. Up to now, there have been no reports on the research of full-field three-dimensional path planning algorithms for the obstacle avoidance picking of fruits in modern apple orchard fruit tree planting modes. The ant colony optimization algorithm (ACO) [22–30] is a representative of intelligent biomimetic algorithms. The algorithm simulates the process of ant colonies releasing pheromones along the path to find food, achieving the search for the optimal path. It has the advantages of adaptability and strong robustness, and is widely utilized in the path planning field. Therefore, to solve the problem caused by modern apple orchard fruit branches on robot fruit picking paths, spindle-shaped fruit branches that were widely planted in modern standard apple orchards were used as the research object, the ant colony algorithm was improved with a design integrating the bionic optimization mechanism of beetle antennae to construct an obstacle avoidance intelligent apple picking 3D path planning method. The proposed method is of great significance for ensuring the good success rate of robot picking and also provides technical support for the development of intelligent apple harvesting robots.

2. Construction of Apple Tree Branch Stem Model and Obstacle Map

2.1. Establishment and Visualization of Spindle-Shaped Apple Tree Trunk and Branch Model

Firstly, based on the actual size and tree shape characteristics of spindle-shaped apple trees, while keeping the relative size ratio of branches unchanged, a three-dimensional model of the spindle-shaped apple tree was established in SpeedTree software (Software for 3D Tree Modeling, version 9, Link of Speedtree software: https://store.speedtree.com/ (accessed on 8 August 2024)). The commonly used 3D modeling software mainly includes Speedtree, AutoCAD (version 2024, Link of the software: https://www.autodesk.com/campaigns/autocad-family (accessed on 8 August 2024)), SketchUp (version 2024, Link of the software: https://www.sketchup.com/zh-cn (accessed on 8 August 2024)), Blender (version 4.1, Link of the software: https://www.blender.org/ (accessed on 8 August 2024)), and Maya (version 2023, Link of the software: https://www.maya.com.sg/ (accessed on 8 August 2024)). The advantages of SpeedTree software [31] mainly include its simple operation, powerful functionality, high degree of customization, and extensive application support. SpeedTree Modeler allows users to easily design their own tree and branch structures by drawing them with a tablet or mouse; even users without professional modeling knowledge can quickly get started. SpeedTree can be integrated into various 3D modeling and rendering software, making it easy for users to import their created vegetation models into projects. SpeedTree has become a widely used tool in fields such as visual effects and digital landscape design due to its ease of use, comprehensive functionality, and highly customizable options. Thus, Speedtree was chosen in this study to model apple trees. The spindle-shaped apple tree mainly includes three types: free-spindle (also known as improved-spindle), high-spindle, and slender-spindle trees.

For the free spindle shape, also known as the improved spindle shape, the tree is about 3 m tall, with a crown width of 2.5–3 m and trunk height of 60–70 cm. The central trunk is upright and straight, with 10–15 lateral branches evenly distributed in a spiral pattern on the central trunk. The lateral branches at the lower part of central trunk are about 70–90° apart, and the distance between the upper and lower lateral branches on the same side is about 50 cm. The length of the lower main branch is about 1.5 m, and the length of lateral branches gradually decreases with the increase in the length of the central trunk. The tree shape presents a spindle shape that is larger at the bottom and smaller at the top. In terms of its tree structure and applicable scenarios, the tree has a moderate-height fixed trunk and high planting density. As the crown of tree moves from bottom to top, the small main branches become smaller and longer, without any lateral branches, and only various types of branch groups. The crown forms quickly, is easy to trim, has good ventilation and

light transmission, with a simple tree structure, is easy to manage, and is also suitable for densely planted orchards.

In terms of its high spindle shape, the central trunk is upright and sturdy, with fine and dense branches. The overall tree has high and fine spindles or a cylindrical shape, with a small and slender crown. The tree height is 3.5–4.0 m, the crown width is 1–1.5 m, and the trunk height is 80–90 cm. On the central trunk, there are 30–50 small and medium-sized fruiting branches arranged in a spiral pattern, with the fruiting branches directly attached to the small main branches. The average length of the small main branches is 0.5–0.8 m, and the angle between lateral branches is about 90–110°. The distance between the upper and lower lateral branches on the same side is about 20 cm. With regards to the tree structure and planting environment, the overall tree has a tall spindle or cylindrical shape, with a high trunk. The tree structure is relatively simple, without permanent main branches, mainly relying on cultivating strong and straight middle trunks to form a skeleton. The tree shape is suitable for close planting cultivation. Given the advantage of it having a smaller crown, the harvesting machine can achieve orchard operations between rows, making it convenient for mechanized picking.

The slender-spindle-shaped tree is 2–3 m tall, with a crown width of 1.5–2.0 m and trunk height of 60–70 cm. On the central trunk, there are 15–20 evenly arranged, thin, and horizontal lateral branches with similar thicknesses. The distance between the upper and lower lateral branches in the same direction is about 60 cm. The lower lateral branch is 1 m long, the middle branch is 70–80 cm long, and the upper branch is 50–60 cm long. The lateral branches at the lower part of the central trunk are about 90–110°, and lateral branches at the upper part of central trunk are about 70–80°. The diameter coarseness ratio of the central trunk to lateral branches is 1: 0.5. The whole tree is slender, with a large crown below and a small crown above, forming a slender spindle shape. In terms of tree structure and planting environment, this tree shape is one with the least backbone branches and the most fruiting branches, suitable for dense planting, dwarf rootstocks, and short-branch varieties. It has characteristic slender and horizontal lateral branches, with longer lower branches and shorter middle and upper branches.

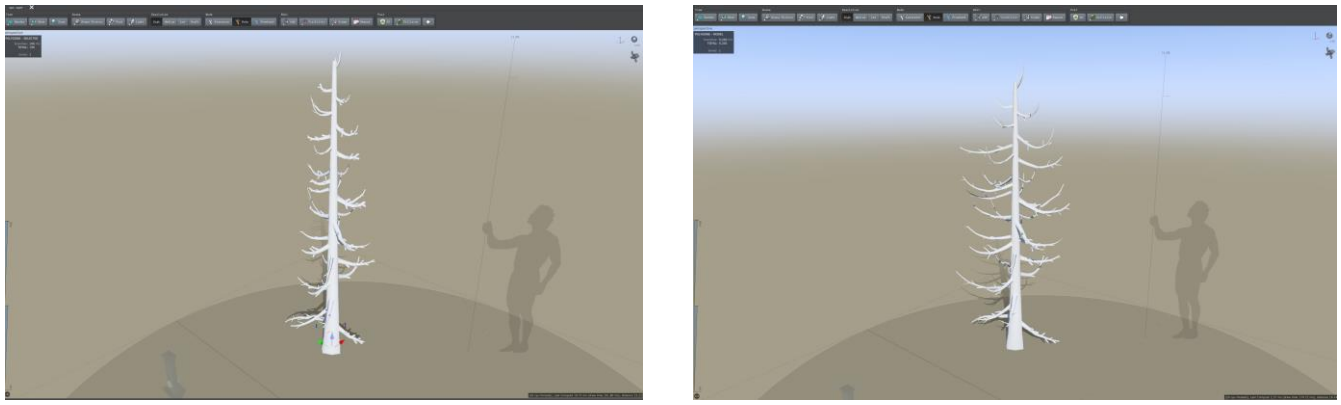
The size and tree shape characteristics of three typical spindle-shaped apple trees are shown in Table 2.

Table 2. The size and tree shape characteristics of three spindle-shaped apple trees.

Tree Shape	Tree Height (m)	Tree Trunk Height (cm)	Crown Width (m)	Number of Lateral Branches	Lateral Branch Angle (degree/°)	Distance of Lateral Branches in Same Side (cm)
Free spindle shape	3	60~70	2.5~3	10~15	70~90	50
High spindle shape	3.5~4.0	80~90	1~1.5	30~50	90~110	20
Slender spindle shape	2~3	60~70	1.5~2.0	15~20	70~110	60

During the modeling process, firstly, the central trunk was established. Then, the number of lateral branches, the angle between lateral branches and the central trunk, and the length of lateral branches was sequentially determined. Finally, the trunk height and crown width based on the relative size between the trunk height and tree height were adjusted. Taking the high-spindle-shaped apple tree as an example, the process of establishing its model is shown in Figure 2. Three-dimensional modeling results of the three typical spindle-shaped apple trees are shown in Figure 3.

(**a**) (**b**)

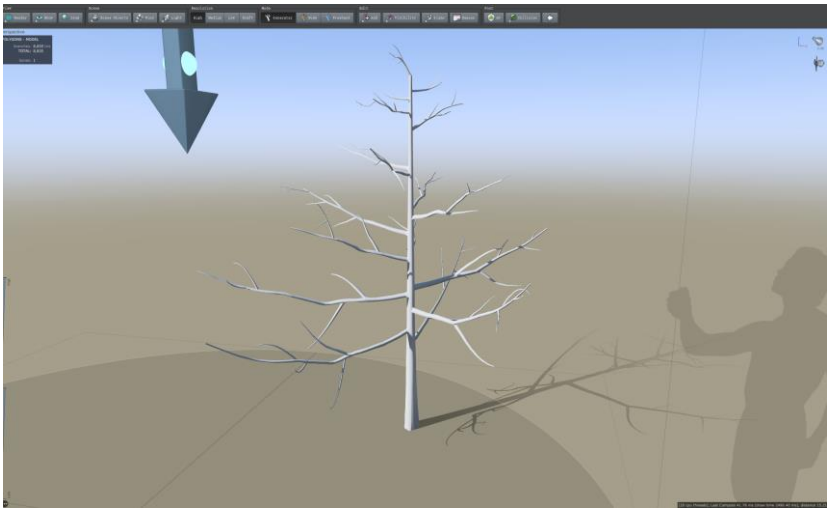


(**c**) (**d**)

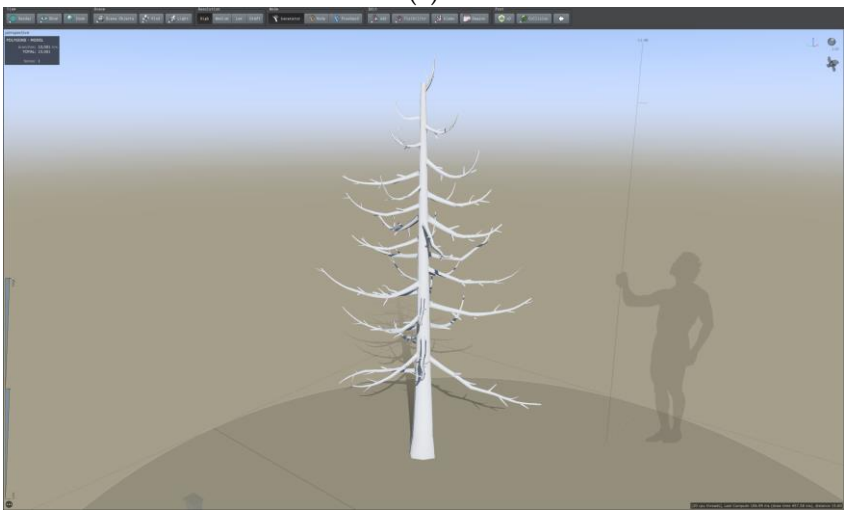
(**e**) (**f**)

Figure 2. The process of establishing a high-spindle-shaped model: determining the central stem (**a**), determining the number of lateral branches (**b**), adjusting the angle between lateral branches and the central stem (**c**), determining the relative length of lateral branches (**d**), adjusting the stem height (**e**), and adjusting the crown size (**f**).

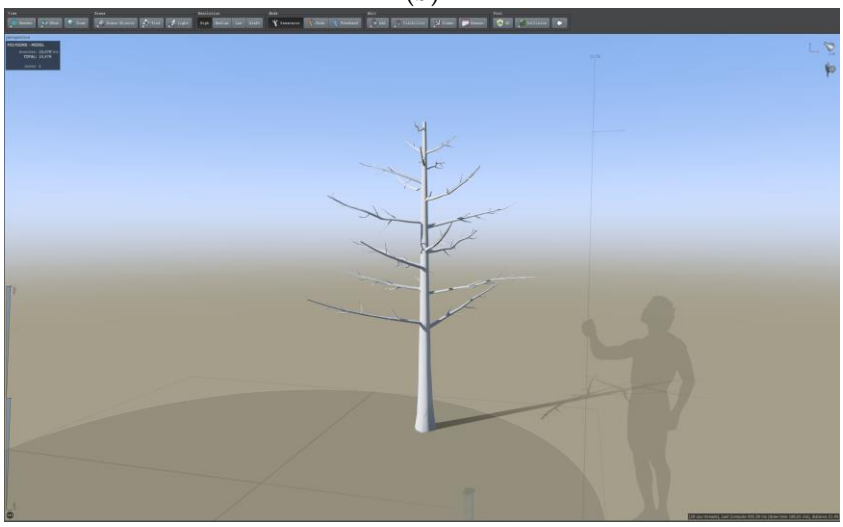
The spindle-shaped apple tree 3D model established in SpeedTree was converted into an STL file that can be read in MATLAB (Version 2024a The Mathworks Inc., Natick, MA, USA). The established tree trunk model was placed in a space enclosed by the non-negative half axis of the x-y-z coordinate axis and visualized in MATLAB. The visualization results of the three typical spindle-shaped apple trees are shown in Figure 4.



(**a**)



(**b**)



(**c**)

Figure 3. Modeling results of three typical spindle-shaped tree: free-spindle (**a**), high-spindle (**b**), and slender-spindle (**c**) trees.

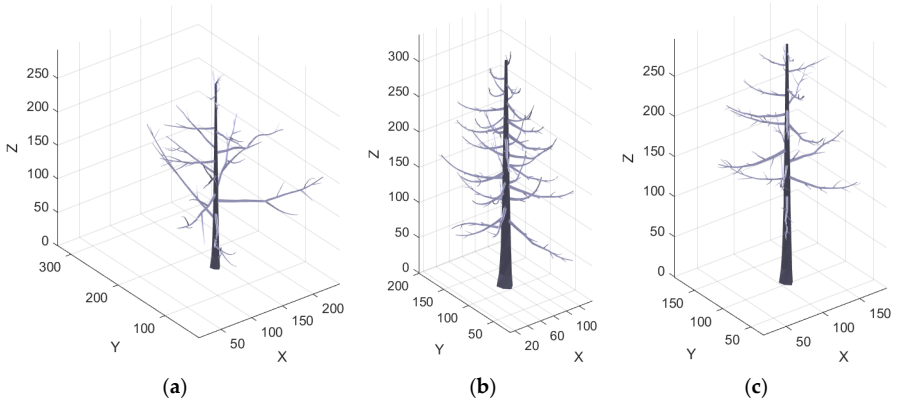


Figure 4. Visualization results of three typical spindle-shaped trees: free-spindle (**a**), high-spindle (**b**), and slender-spindle (**c**) trees.

2.2. Representation of Obstacles and Construction of Maps in Three-Dimensional Path Planning

The 3D model itself contains a large amount of data, which causes an inconvenience in searching and the calculation of path planning, and thus affects obstacle avoidance efficiency. Therefore, in order to improve the efficiency of path planning, it is necessary to preprocess the established 3D model of apple tree branches. Simplifying the complex 3D model into several small cubes containing feature points, to reduce the amount of data in the 3D spatial obstacle map, and to compensate for local representativeness of feature points, is necessary. The operation not only expands the boundary of feature points but also increases the safety range in path planning, reducing computational complexity.

Then, three-dimensional feature points were extracted. Feature points were expanded to form a small cube containing the original feature points, and several small cubes around the branch trunk feature points were obtained. Our procedure was inspired by the grid method, where encoded grids are utilized to represent maps, and which is commonly used to model the environment in path planning. We marked grids containing obstacles as obstacle grids; otherwise, we marked them as free grids, which were used as the basis for path searching. These small cubes containing feature points were considered obstacles; the obstacles were considered infeasible cubes and marked as 1. The remaining free space was marked as 0, and a three-dimensional 0–1 matrix was constructed to form a three-dimensional obstacle map of tree branches. The small cube map containing branch trunk feature points was utilized as an obstacle environment map for obstacle avoidance path planning. The expansion effect of feature points is shown in Figure 5.

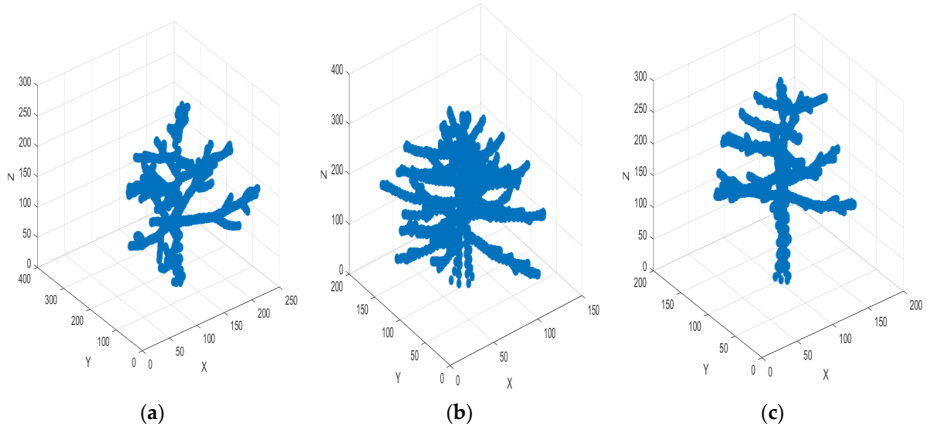


Figure 5. The expansion effect of feature points in three typical tree shapes: free spindle (**a**), high spindle (**b**), and slender spindle (**c**).

3. Obstacle Avoidance 3D Path Planning Based on Improved Ant Colony Algorithm

3.1. Traditional Ant Colony Algorithm

The ant colony algorithm (ACO) is an intelligent biomimetic optimization algorithm that simulates the behavior of ants searching for food, guides searches with pheromones, and ultimately finds the best path. During the process of searching for food, ants mark paths they walked on and release pheromones, which affects the selection of subsequent ant paths. During free space movement, ants mark paths while conducting path searches. During searching, ants choose appropriate paths based on the concentration of pheromones, and paths with more pheromones are more likely to be selected. The more paths ants traverse, the more pheromones they contain, attracting more ants to choose these paths. As the number of iterations increases, the optimal path is ultimately found.

The execution process of ant colony algorithm is as follows: Firstly, the parameters are initialized to generate an obstacle map. Then, the ant colony begins iteration and updates pheromones on the nodes on the map to calculate the transition probability. The transition probability is related to the concentration of pheromones and the length of the path. Based on the transition probability, the next node is selected. Finally, the termination condition of algorithm iteration is determined [22–26].

3.2. Improvement of Ant Colony Algorithm

3.2.1. Uneven Distribution of Pheromone Concentration

In the traditional ant colony algorithm, the initial pheromone concentration distribution in a mobile space is uniform, with values of the same size. The pheromone concentration at different nodes from the target point is basically the same. During the initial search for a path, ants have little difference in pheromones on all feasible nodes, which leads to insufficient heuristic information in the initial search of the ant colony, making it difficult to converge, resulting in a longer path planning time. Therefore, the uneven distribution of initial pheromone concentrations in a feasible solution space in the study enhanced the differences between different paths, and had a certain guiding effect on ants in the early stage of path searching.

The distance calculation formula is shown in Formula (1):

$$L = \sqrt{(x_0 - d_0)^2 + (y_0 - d_1)^2 + (z_0 - d_2)^2} \quad (1)$$

L is the distance from a feasible spatial point (x_0, y_0, z_0) to the target point (d_0, d_1, d_2). The closer to the target node, the larger the initial value of corresponding pheromones, while the farther away from the target node, the smaller the initial value of corresponding pheromones, thus achieving an uneven distribution of initial pheromones. The specific implementation method is as follows: Firstly, the distance between all free nodes (x_0, y_0, z_0) in the 3D map and the target node (d_0, d_1, d_2) is calculated. Secondly, based on the total amount of information and distance, the initial pheromone value is obtained. Finally, the initial pheromone values of each free node in the matrix are stored.

3.2.2. Improvement of Pheromone Update Rules

The decay rate of pheromones (ρ) on all paths is consistent, while traditional ant colony algorithms update pheromones on the paths traveled by ants, and the advantages of shorter paths cannot be demonstrated, resulting in lower search efficiency. Therefore, in the study, the attenuation rate of pheromones was improved, resulting in dynamic variation.

The specific improvement method for pheromone update rules is as follows: the pheromone decay rate is associated with the fitness value (*BestFitness*) used in the ant colony algorithm to measure the path length and height. The path fitness value dynamically changes with the path length and height. When the path length becomes longer, the fitness value will increase, and the pheromone decay rate on longer paths will also increase accordingly. When the path distance becomes shorter, the fitness value will decrease, and the pheromone decay rate on shorter paths will decrease. On the other hand, setting a range

for variations in the pheromone decay rate limits the process of pheromone decay within a controllable range. If the decay rate of pheromones is too high and the volatilization rate of pheromones on the path is too fast, it will lead to a rapid decrease in the residual pheromone content on the path. Ants may not be able to perceive pheromones transmitted between ant colonies immediately, reducing their ability to transmit information. Ants may choose other paths that have not been searched for and skip the better ones that have already been explored, resulting in repeated searches. At the same time, the positive feedback mechanism is weakened and the search speed is reduced. On the contrary, if the decay rate of pheromones is too small, more pheromones will remain on the path. In the early stage of ant exploration, the concentration of pheromones on the path is too high, and the algorithm falls into local optimal solutions too early on. The search stops early on, and the global optimal path cannot be obtained, which reduces the optimization ability of the algorithm.

The mathematical expression for the improved pheromone update rule is as follows:

$$\rho = \begin{cases} \rho_{\min}, \rho < \rho_{\min} \\ 0.3 \cdot h \cdot BestFitness, \rho_{\min} \leq \ \rho \leq \rho_{\max} \\ \rho_{\max}, \rho > \rho_{\max} \end{cases} \tag{2}$$

$$Fitness = h_1 + l_1 \tag{3}$$

In the formula, $BestFitness$ represents the best fitness value, $Fitness$ represents the fitness value, l_1 represents the path length, and h_1 represents the average height of the path. h is dependent on a specific situation, and was 5.1×10^{-3} in this study.

3.2.3. Integrating of the Beetle Antenna Optimization Mechanism

Although the ant colony algorithm has good stability and wide-ranging applications in path planning, there may still be situations in which it falls into local optima, resulting in the obtained results potentially not being the global optima. Therefore, it is essential to optimize the initial path obtained. In the study, the optimization mechanism of the beetle antenna algorithm was used [32–35]. The beetle antenna search algorithm is a biomimetic algorithm that simulates the process of beetles searching for food using their antennae. Beetles have two antennae that can sensitively capture changes in the surrounding environment. When there is a difference in the odor concentration sensed by the left and right antennae, they tend to move in the direction of the higher concentration. By adjusting the movement direction according to the concentration difference between antennae, they ultimately find food. The mathematical model of the beetle antenna algorithm is as follows:

The position of the left antenna of a beetle is calculated as follows:

$$X_l = X_0 + \frac{1}{2} l \cdot dir \tag{4}$$

The position of right antennae of a beetle is calculated as follows:

$$X_r = X_0 - \frac{1}{2} l \cdot dir \tag{5}$$

The fitness function of the antennae of a beetle is calculated as follows:

$$f(X) = \sqrt{(x_1 - x_\tau)^2 + (y_1 - y_\tau)^2 + (z_1 - z_\tau)^2} \tag{6}$$

Among them, $X_0(x_1, y_1, z_1)$ is the centroid position of the beetle, $X_\tau(x_\tau, y_\tau, z_\tau)$ is the endpoint position, l is the distance between two antennae, and dir is a unit random vector. The fitness function of the left antenna of a beetle is $f_l = f(X_l)$, the fitness function of right antenna of a beetle is $f_r = f(X_r)$, and $step$ is the step size for forward movement, which was set as 3 in this study; $w = f_l - f_r$.

The formula for updating the centroid position of beetle antennae is as follows:

$$X = X - step \cdot dir \cdot sign(w) \tag{7}$$

$$sign(w) = \begin{cases} 1 & w > 0 \\ 0 & w = 0 \\ -1 & w < 0 \end{cases} \tag{8}$$

If the fitness function value of the left antenna is greater than that of the right side, then $sign(w)$ equals 1, and the beetle moves to the right. If the fitness function value of the left antenna is smaller than that of the right side, the $sign(w)$ equals -1, and the beetle moves to the left. If the fitness function value of the left antenna is equal to that of the right, then $sign(w)$ equals 0, and the centroid position of the beetle remains unchanged.

On the basis of the initial path planned by the ant colony algorithm, the global path is optimized by utilizing the fast search speed and small data computation of the beetle antenna search algorithm. Due to the existence of some turning points in the optimized path of the beetle antenna algorithm, further smooth processing is needed on the planned path.

3.2.4. Smooth Processing of Planning Paths

The path planned introduced after Section 3.2.3 was a curve directly connected by discrete points, with redundant points, that is, unnecessary, repetitive, or tortuous nodes were in the path. If the generated path is directly applied to the harvesting robot, it not only increases the distance the robot moves across, but also may cause unnecessary pauses or turns during the movement, thereby affecting the harvesting efficiency of the robot. Therefore, a cubic B-spline curve was further adopted in this study to smoothen the basic path optimized by the beetle antenna algorithm.

The mathematical expression for the n-fold uniform B-spline curve is as follows:

$$P_n(t) = \sum_{j=0}^{n} E_{j,n}(t) P_j, \ t \in (0,1) \tag{9}$$

In the equation, P_j represents the equation of the control point in segment j, while $E_{j,n}$ represents the basis function of the sub-uniform B-spline curve, and the expression is as follows:

$$E_{j,n}(t) = \frac{1}{n!} \sum_{m=0}^{n-j} (-1)^m C_{n+1}^m (t + n - m - j)^n \tag{10}$$

While $n = 3$, by substituting the above equation, the basis function of the cubic uniform B-spline curve was obtained, and the expression is as follows:

$$\begin{cases} E_{0,3}(t) = -\frac{1}{6}t^3 + \frac{1}{2}t^2 - \frac{1}{2}t + \frac{1}{6} \\ E_{1,3}(t) = \frac{1}{2}t^3 - t^2 + \frac{2}{3} \\ E_{2,3}(t) = -\frac{1}{2}t^3 + \frac{1}{2}t^2 + \frac{1}{2}t + \frac{1}{6} \\ E_{3,3}(t) = \frac{1}{6}t^3 \end{cases} \tag{11}$$

$$P_j = \frac{1}{6} \tag{12}$$

There are many types of B-spline curves, and B-spline curves are a special case of Bezier spline curves. Based on the distribution characteristics of nodes, they can be subdivided into uniform B-spline curves, non-uniform B-spline curves, quasi-uniform B-spline curve, and piecewise Bezier curves.

Due to the fact that both uniform and non-uniform B-spline curves do not pass through the starting and ending control points during the fitting process, and that the internal control points of segmented Bezier curves are discontinuous, quasi-uniform B-spline curves retain the geometric properties of the endpoints of Bezier curves, that is, they pass through the

starting and ending control points during the fitting process, which cannot be achieved by uniform and non-uniform B-spline curves.

On the other hand, in practical applications, quadratic and cubic B-spline curves are mainly used, but quadratic B-spline curves also do not pass through the starting and ending control points during the fitting process. Therefore, to ensure the integrity of the planned path, as well as its continuity and smoothness, quasi-uniform B-spline curves were utilized to finely smoothen the path and remove redundant nodes in the obstacle avoidance path in this study. Due to the higher degree of the curve, more segments of the curve were affected by control points; the higher the degree of the derivative of the curve, the more peaks and valleys the curve will have, which is not conducive to local control. Therefore, the quasi-uniform cubic B-spline curve was ultimately selected as the basis function for the smooth processing of the path to satisfy the requirements of path planning.

In an actual harvesting operation of the apple-picking robot, the starting position of the apple-picking robot arm is a point outside the tree crown, and the ending point is a point inside the apple tree crown (the spatial position of the fruit to be picked). In this study, the randomly set starting and ending points of the path planning followed this principle according to the actual operation of the apple-picking robot. The designed obstacle avoidance picking path is the running trajectory of the end-effector for the apple-picking robotic arm, which can be applied to different configurations of agricultural-fruit-picking robotic arms.

Based on the improved ACO algorithm, a pair of starting points, [170, 83, 147] and [100, 118, 228], were randomly selected from a slender spindle-shaped apple tree for path planning. The unsmooth and smoothed paths were compared, and it was obvious that the number of inflection points in the smoothed path was significantly reduced. A comparison of the path before and after smoothing is shown in Figure 6.

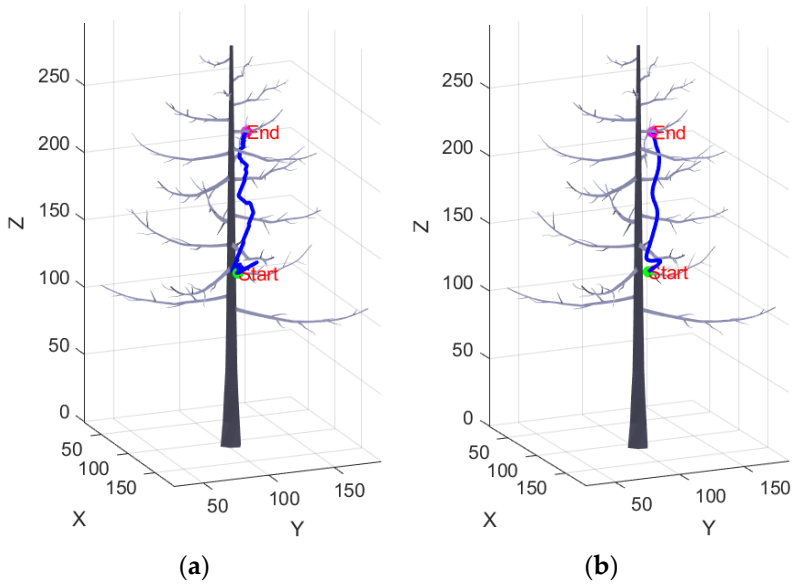


Figure 6. Comparison of path smoothing before smoothing (**a**) and after smoothing (**b**).

Based on the above improved design, an optimized ant colony algorithm was created, which changed the pheromone update strategy by adjusting the uneven distribution of initial pheromones and integrating the optimization mechanism of the beetle antenna algorithm to overcome the shortcomings of traditional ant colony algorithms, which are prone to getting stuck in local optima and to slow convergence speed. Finally, the planned path was smoothed by utilizing a cubic B-spline curve, and a smooth path for robot obstacle avoidance picking was obtained.

The specific steps of the three-dimensional path planning method for the obstacle avoidance picking of spindle-shaped apple trees based on the improved ant colony algo-

rithm are as follows: Firstly, the parameters are initialized, an obstacle map is generated, and then the ant colony starts iteration. At the same time, the pheromones on the nodes on the map are updated, and the transition probability is calculated. Then, the next node is selected based on the transition probability, the path of the ant colony algorithm is recorded, and whether or not the initial path satisfies the termination condition of algorithm iteration is determined, that is, whether or not the maximum number of iterations or the target point is reached. If not, the node is searched for again. Finally, the path is smoothened after the second optimization, and final planned path is output. A flowchart for the improved ant colony algorithm is shown in Figure 7a. The improvements to the proposed obstacle avoidance path planning algorithm compared to the original algorithm are indicated by yellow modules.

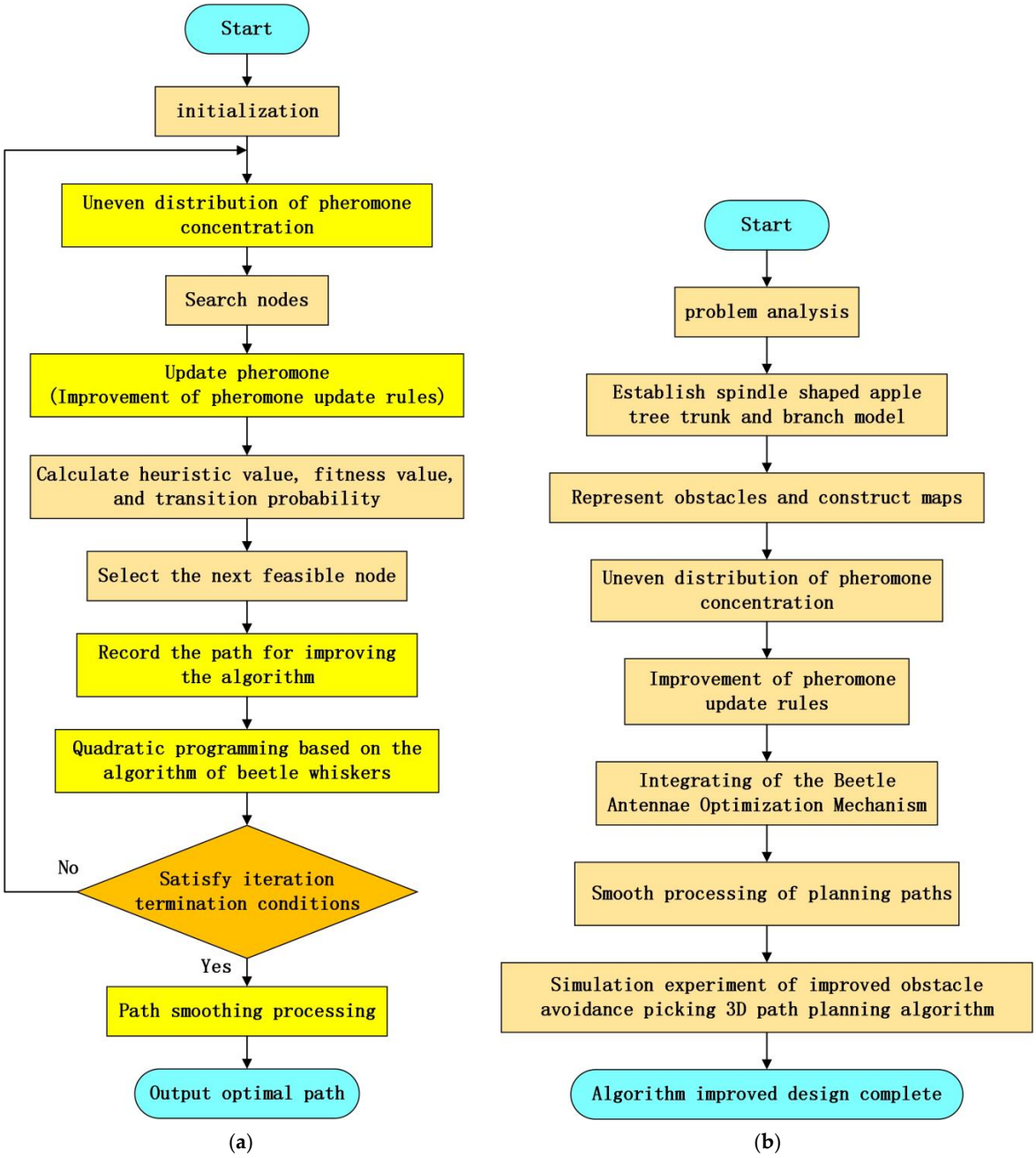


Figure 7. Flowchart of the improved ant colony algorithm (**a**) and flowchart for the whole study (**b**).

An overall design flowchart of the study is shown in Figure 7b, which mainly includes eight steps: problem analysis, representation of obstacles and construction of maps, determination of the uneven distribution of pheromone concentrations, improvement of pheromone update rules, integration of the beetle antenna optimization mechanism, smooth processing of planning paths, and conduction of the simulation experiment. Finally, the improved 3D obstacle avoidance path planning algorithm is obtained.

4. Simulation Experiment and Analysis of Improved Obstacle Avoidance Picking 3D Path Planning Algorithm

4.1. Experiment Results and Analysis of Improved Algorithm

In order to verify the effectiveness of the improved proposed 3D path planning algorithm for obstacle avoidance in picking from the crown of the spindle-shaped apple tree, a 3D path planning simulation experiment was conducted using the MATLAB 2024a platform on the Windows 11 system. Three different pairs of starting target points were taken for each of the three spindle-shaped tree shapes, and three sets of obstacle avoidance experiments were conducted on internal target points (representing the spatial position of the fruit) from three different directions outside the tree, with 20 experiments in each group. The experimental data were subsequently averaged.

The definition and mathematical expression of the planning success rate are as follows:

$$planning\ success\ rate = \frac{Number\ of\ successful\ searches}{Number\ of\ successful\ searches + Number\ of\ failed\ searches} \times 100\% \quad (13)$$

In the equation, $Number\ of\ successful\ searches$ represents the number of nodes searched within the map range but not within obstacles. $Number\ of\ failed\ searches$ represents the number of nodes searched within obstacles or outside the map range. $planning\ success\ rate$ represents the ratio of successful search times divided by the total search times (i.e., the ratio of the number of grids searched within the map range but not inside the obstacle divided by the total number of search grids).

The size of the free-spindle apple tree 3D environment map was 240 × 330 × 280 (cm × cm × cm), the size of high-spindle apple tree 3D environment map was 140 × 200 × 340, and the size of the slender-spindle apple tree 3D environment map was 140 × 200 × 340. The simulation results of obstacle avoidance picking path planning for free-spindle apple trees utilizing improved algorithms are shown in Table 3, and the three-dimensional visualization results of the simulation experiment are shown in Figure 8:

Table 3. Experiment results of three-dimensional path planning for free-spindle apple tree.

Test Group	Starting Point	Target Point	Planning Time (s)	Length of the Path (cm)	Planning Success Rate (%)
1	(57, 40, 99)	(130, 159, 162)	4.0189	165.5681	96.2969
2	(200, 93, 115)	(129, 161, 208)	3.5484	146.0332	96.2237
3	(156, 292, 108)	(126, 160, 161)	4.1873	160.9183	96.2969

According to Table 3, the improved algorithm has high success rates in obstacle avoidance for the free-spindle tree shape, all of which are above 96%, and the planning time is short. As shown in Figure 8, which shows the results of the three sets of path planning visualization experiments, the improved algorithm can effectively avoid branches and trunks and reach the target point inside the crown of free-spindle fruit trees. Furthermore, the planned path is relatively smooth, indicating that he proposed algorithm is effective in obstacle avoidance inside the free-spindle tree crown.

The simulation results of obstacle avoidance picking path planning for high-spindle apple trees using improved algorithms are shown in Table 4, and the three-dimensional visualization results of the simulation experiment are shown in Figure 9.

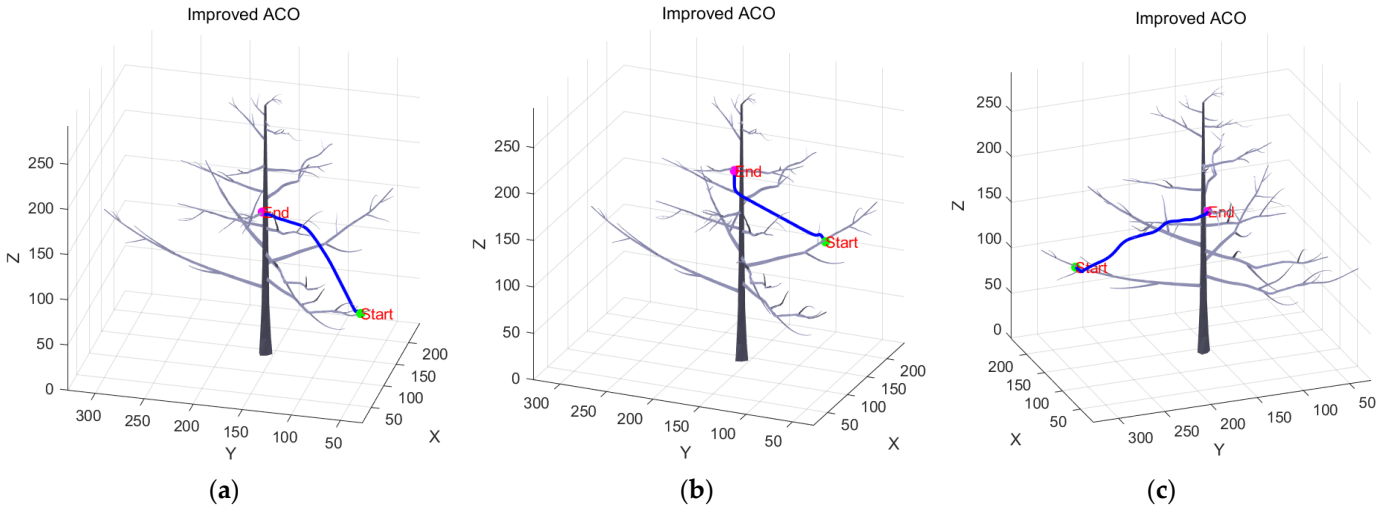


Figure 8. Visualization experiment results of three-dimensional path planning for free-spindle apple tree: group 1 (**a**), group 2 (**b**), and group 3 (**c**).

Table 4. Experiment results of three-dimensional path planning for high-spindle apple tree.

Test Group	Starting Point	Target Point	Planning Time (s)	Length of the Path (cm)	Planning Success Rate (%)
1	(30, 90, 99)	(72, 120, 194)	4.02495	123.6424	86.6658
2	(103, 35, 65)	(94, 102, 181)	4.01175	146.7101	95.0046
3	(114, 167, 91)	(80, 123, 179)	3.46035	122.7225	96.29711

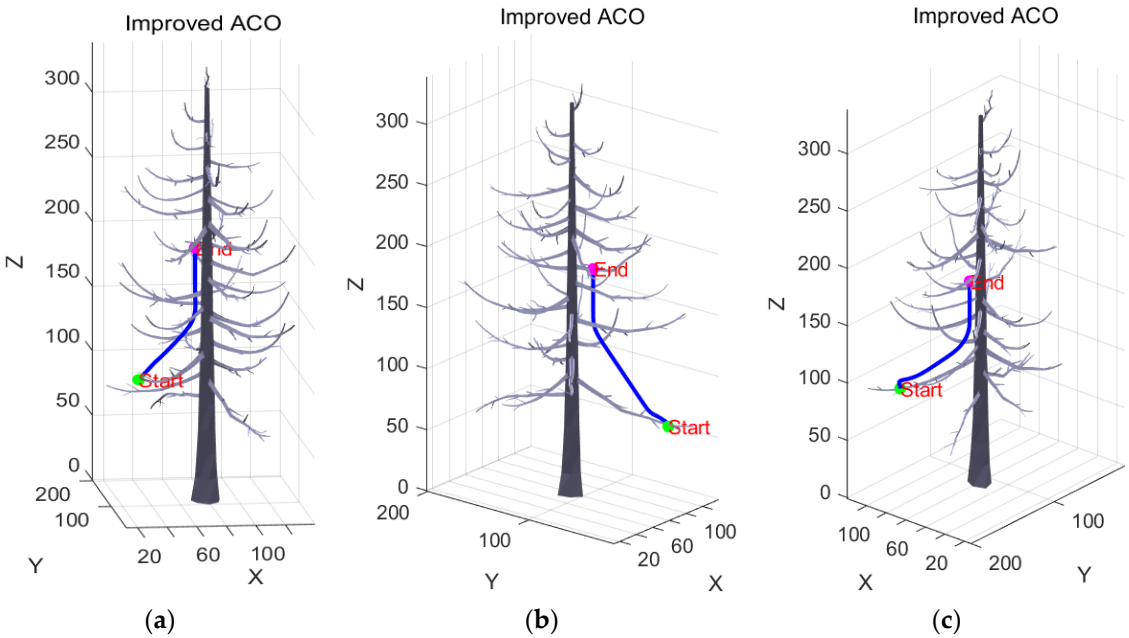


Figure 9. Visualization experiment results of three-dimensional path planning for high-spindle apple tree: group 1 (**a**), group 2 (**b**), and group 3 (**c**).

As shown in Table 4, the success rate of obstacle avoidance using the improved algorithm in high-spindle trees is all above 86%, and the planning time is relatively short. According to this, combined with the path planning results of the three sets of simulation experiments, as shown in Figure 9, the improved algorithm can still successfully plan feasible smooth paths in high-spindle trees with densely distributed branch trunks, indicating

that the algorithm is effective in obstacle avoidance path planning within the high-spindle tree crown.

The simulation results of obstacle avoidance picking path planning for slender-spindle apple trees utilizing the improved algorithms are shown in Table 5, and the three-dimensional visualization results of the simulation experiment are shown in Figure 10:

Table 5. Experiment results of three-dimensional path planning for slender-spindle apple tree.

Test Group	Starting Point	Target Point	Planning Time (s)	Length of the Path (cm)	Planning Success Rate (%)
1	(50, 35, 65)	(101, 107, 195)	4.09965	174.0005	92.2800
2	(153, 170, 99)	(99, 108, 209)	3.7123	154.4546	93.6202
3	(151, 44, 131)	(103, 94, 214)	3.18245	121.5680	96.2876

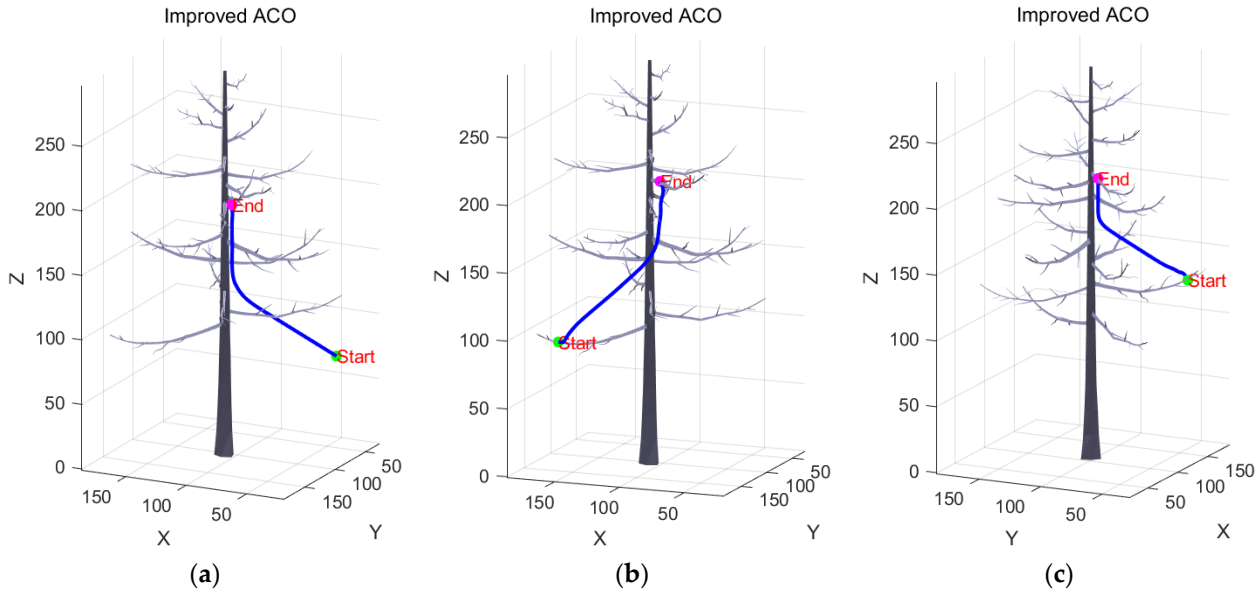


Figure 10. Visualization experiment results of three-dimensional path planning for slender-spindle apple tree: group 1 (**a**), group 2 (**b**), and group 3 (**c**).

As shown in Table 5, the success rates of obstacle avoidance using the improved algorithm in slender-spindle fruit trees are all above 92%, and the planning time is relatively short. Combined with the path planning results of the three sets of simulation experiments, as shown in Figure 10, the improved algorithm can successfully plan a feasible smooth path, proving the effectiveness of the proposed algorithm for obstacle avoidance fruit picking path planning in slender-spindle fruit trees.

Based on the differences between three tree structures explained in Section 2.1, the distance of lateral branches on the same side directly determines the complexity of path planning, and the smaller the value, the greater the difficulty of path planning. For 3D obstacle avoidance path planning tasks, due to the larger distance of lateral branches on the same side of a free-spindle apple tree, their path planning success rates are relatively high compared to those for the other two tree structures, with a planning success rate of over 96%. For high-spindle apple trees, due to their minimum distance of lateral branches on the same side, their planning success rate is relatively low, with rates below 90%. For the slender-spindle-shaped apple tree, the distance of lateral branches on the same side is greater than that for the free-spine-shaped apple tree and less than that of the high-spine-shaped apple tree; thus, its planning success rate is in the middle of that for the other two tree types.

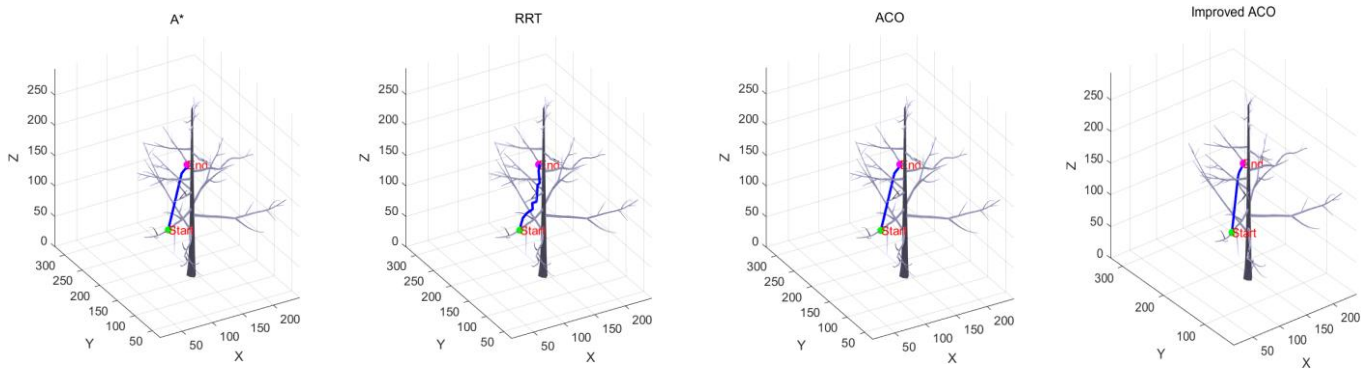
4.2. Comparative Experiments and Analysis of Different 3D Path Planning Algorithms

In order to further analyze the performance of the proposed obstacle avoidance path planning algorithm, comparative simulation experiments were conducted on three typical spindle-shaped apple trees based on the A* algorithm, RRT algorithm, traditional ant colony algorithm, and improved ant colony algorithm for path planning. On the established map, tree branches are obstacles, and the other parts are free, feasible areas. Each algorithm conducted 40 obstacle avoidance path planning simulation experiments for three typical spindle-shaped apple tree shapes, and finally, the average of the experimental results was calculated.

Among them, the size of the 3D map of free-spindle-shaped apple tree is 240 × 330 × 280. The starting point of the comparative simulation experiment path was (79, 117, 114), and the ending point was (140, 169, 178). The path planning results are shown in Table 6 and Figure 11.

Table 6. Comparative experiment results of three-dimensional path planning for free-spindle-shaped apple tree.

Path Planning Algorithm	Planning Time (s)	Number of Path Nodes	Length of the Path (cm)	Planning Success Rate (%)
A*	1.5670625	86	107.9408	89.4007
RRT	0.0069	14	116.6064	99.2373
Traditional ACO	6.223555	65	105.7946	96.2963
Improved ACO	3.15065	101	105.9369	96.2975



(**a**) (**b**) (**c**) (**d**)

Figure 11. Visualization experiment results of three-dimensional path planning for free-spindle-shaped apple tree: A* (**a**), RRT (**b**), traditional ACO (**c**), and improved ACO (**d**).

According to Table 6 and Figure 11, it can be seen that for the path planning of free-spindle apple trees, the traditional ACO algorithm has advantages in terms of planning path length and success rate; however, the planning time is relatively long. The success rate of the RRT algorithm planning was high, the planning time was short, and the computational complexity was small. Although the number of path nodes was small, it could quickly plan the path, but obstacle information might not have been ignored, resulting in the planned path not being precise enough with a long path length. The performances of the A* algorithm in terms of the planning path length, success rate, and number of path nodes were not good enough. The improved ACO algorithm increased the planning speed by 49.38% compared to that achieved by the traditional ACO algorithm, and the local optimization of the traditional ACO algorithm was effectively improved. The planning success rate was 96.2975%, which satisfies application requirements.

The size of the 3D map of the high-spindle-shaped apple tree is 140 × 200 × 340. The starting point of the comparative simulation experiment was (60, 60, 133), and the ending point was (89, 114, 196). The path planning results are shown in Table 7 and Figure 12.

Table 7. Comparative experiment results of three-dimensional path planning for high-spindle-shaped apple tree.

Path Planning Algorithm	Planning Time (s)	Number of Path Nodes	Length of the Path (cm)	Planning Success Rate (%)
A*	1.80659231	101	97.1706	86.6717
RRT	0.01174359	12	103.1142	82.4508
Traditional ACO	6.10630257	64	94.5848	96.2963
Improved ACO	3.27722500	101	94.5292	96.2975

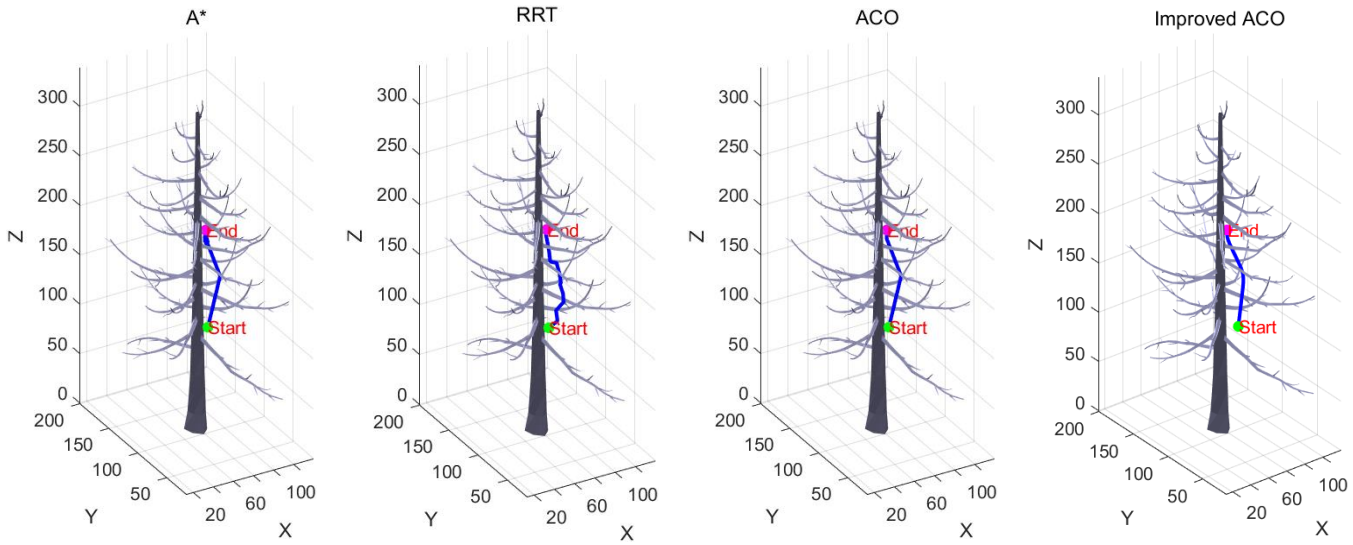


Figure 12. Visualization experiment results of three-dimensional path planning for high-spindle-shaped apple tree: A* (**a**), RRT (**b**), traditional ACO (**c**), and improved ACO (**d**).

According to Table 7 and Figure 12, for the path planning of high-spindle apple trees, the improved ACO algorithm was superior to the other three algorithms in terms of path length and planning success rate. Furthermore, the improved ant colony algorithm increased the planning speed by 46.33% compared to that achieved by the original ant colony algorithm. Although the planning speeds of the A* and RRT algorithms were relatively fast, they had relatively long planned path lengths, and relatively low planning success rates, reaching 86.6717% and 82.4508%, respectively. From Figure 12, it can be seen that the RRT algorithm and A* algorithm had more inflection points in path planning, and there were also obvious inflection points in the path planning of the traditional ACO algorithm, while the path planning of the improved ACO algorithm was smoother. Therefore, the proposed improved path planning algorithm is also applicable to high-spindle-shaped apple trees with densely distributed branches.

The three-dimensional map size of the slender-spindle-shaped apple tree was 140 × 200 × 340. The starting point of the comparative simulation experiment path was (140, 50, 100), and the ending point was (87, 105, 161). The path planning results are shown in Table 8 and Figure 13.

According to Table 8, in terms of path length, the traditional ACO algorithm was superior to the A* and RRT algorithms, and the planning success rate was relatively high. However, the planning time was relatively long. The RRT algorithm had a shorter planning time, but the number of path nodes was too small. Compared with other algorithms, the length of paths planned by the RRT algorithm was too long, and the quality of planned paths was relatively low. From Figure 13, it can be seen that there were many inflection points on the path of the RRT algorithm. If the algorithm is directly applied for the movement of the picking robotic arm, it may cause the robotic arm to conduct unnecessary

redundant operations, thereby reducing the efficiency of the task. The planning success rate of the A* algorithm was relatively low; therefore, in picking environments with dense obstacles, it is more prone to getting stuck in local optima, contrasting other algorithms. The planned path may not reach the target point, thus reducing the effectiveness of the path. The improved ACO algorithm, while ensuring a higher success rate and shorter path, increased the planning speed by 51.03% compared to that of traditional ACO algorithms, with a planning time of 3.089275s, and the local optimization of the traditional ACO algorithm was effectively improved. Although the path length was slightly increased, it still had advantages compared with RRT algorithms.

Table 8. Comparative experiment results of three-dimensional path planning for slender-spindle-shaped apple tree.

Path Planning Algorithm	Planning Time (s)	Number of Path Nodes	Length of the Path (cm)	Planning Success Rate (%)
A*	1.79383	101	102.9661	87.9665
RRT	0.0064675	13	113.2323	99.6133
Traditional ACO	6.3079625	62	100.6271	96.2963
Improved ACO	3.089275	101	106.3939	96.2975

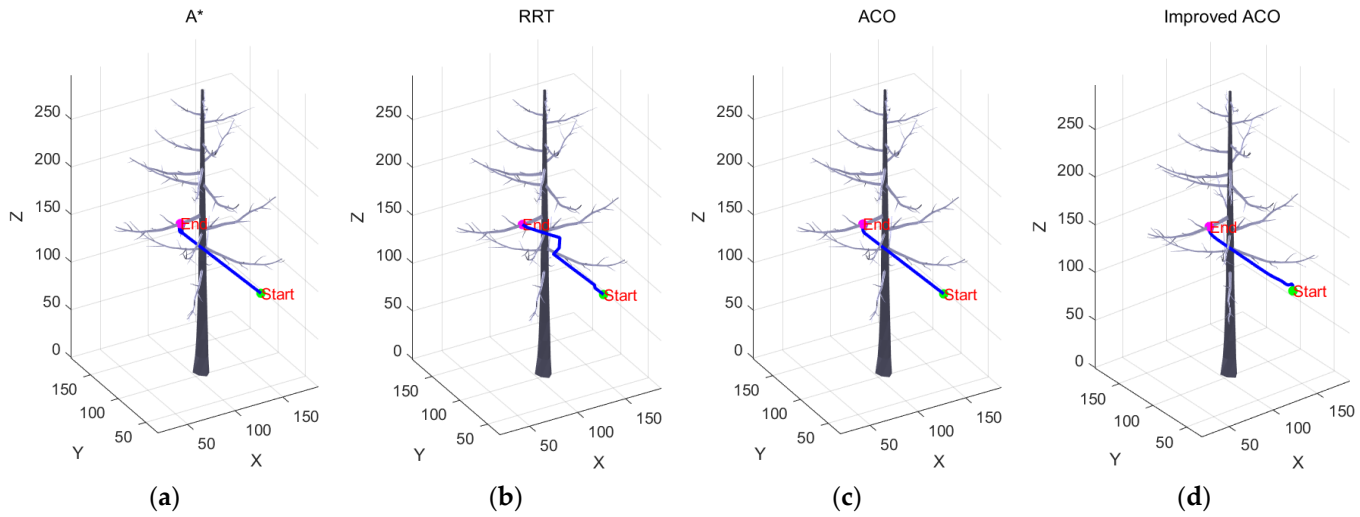


Figure 13. Visualization experiment results of three-dimensional path planning for slender-spindle-shaped apple tree: A* (**a**), RRT (**b**), traditional ACO (**c**), and improved ACO (**d**).

Overall, compared to other algorithms, the improved ACO algorithm had significant advantages for high-spindle-shaped trees with dense branch trunks. The planned path had fewer redundant points, shorter path length, and smoother paths, making it more suitable for robot harvesting operations with densely distributed branch obstacles. In path planning experiments for free-spindle and slender-spindle apple trees, although the A* algorithm and RRT algorithm had faster planning speeds compared to those of ACO algorithms, the former two algorithms had fewer path nodes compared to the improved ACO algorithm. For high-spindle and slender-spindle tree shapes, the improved ACO algorithm had a similar number of nodes to the A* algorithm, and almost 7.7 times the number of RRT algorithm path nodes. The presence of fewer nodes in the path indicates that the path can be quickly planned; however, it may lead to inaccurate path planning results since, in complex three-dimensional environments, fewer nodes may not fully reflect the details and features of the environment, resulting in the planned path not being optimal. In complex environments such as those with dense obstacles or narrow channels, fewer nodes may not be able to effectively represent the information, which may result in path planning algorithms being unable to find feasible paths or finding paths with significant operational

risks. In addition, in the obstacle avoidance experiment on high-spindle apple trees, the A* algorithm and RRT algorithm had relatively low planning success rates, and the RRT algorithm did not have advantages in path length, so neither one or the other can guarantee the effectiveness of path planning.

In terms of the planning time, in the three-dimensional obstacle map of the free-spindle apple tree, the average planning time of the improved ACO algorithm proposed in the study was reduced by 49.38% compared to that of the traditional ACO algorithm. On the 3D obstacle map of the high0spindle apple tree, which was more complex than that of the free-spindle apple tree, the average planning time of the improved ACO algorithm was reduced by 46.33% compared to that of the traditional ACO algorithm. On the three-dimensional obstacle map of slender-spindle-shaped apple trees, the improved ACO algorithm reduced the average planning time by 51.03% compared to the traditional one. Although the two ACO algorithms still have room for improvements in planning time compared to the A* and RRT algorithms, they satisfy application requirements. The main reason for the improvement in the ACO algorithm compared to the traditional ACO algorithm in terms of search speed is introduction of an initial pheromone non-uniform distribution strategy and a pheromone decay rate dynamic change strategy, which ensured the efficiency of path searching while avoiding obstacle branches.

Overall, the improved ACO algorithm has the ability to search for optimal paths with higher success rates on the three-dimensional obstacle map of spindle-shaped apple trees. Furthermore, its obstacle avoidance advantage is more prominent in free-spindle-shaped apple trees with densely distributed branches. On the other hand, in the path planning experiments on three different apple tree shapes, the smoothness of the path planned by the improved ACO algorithm was higher, indicating that the improved ACO algorithm is feasible and effective in obstacle avoidance fruit picking three-dimensional path planning for spindle-shaped apple trees.

5. Discussion

From the data from the comparative experiment results for three-dimensional path planning, it can be seen that although the proposed improved algorithm does not significantly improve the success rate of path planning compared to the original ACO algorithm, the planning speed of the proposed planning algorithm is nearly twice that of the original algorithm, and the number of nodes included in the planned path is greater than that of the original ACO algorithm, indicating the higher smoothness of the path planned by the improved algorithm. These improvements allow the robotic arm to move more smoothly along a path, showing the advantages of the improved algorithm.

On the other hand, increasing path length will increase picking time. Compared with the original ACO algorithm, for the free-spindle-shaped apple tree, the path length increased by 0.1423 cm. For the high-spindle-shaped apple tree, the length of the path decreased by 0.0556 cm. For the slender-spindle-shaped apple tree, the path length increased by 5.7668 cm. However, the path planning time of the proposed algorithm was nearly 3 s faster than that of the original algorithm. Based on the fact that the time required for the robotic arm to run a length of 6 cm was much less than 3 s, the proposed algorithm has significant advantages from a comprehensive performance perspective.

In addition, the path planning algorithm does not involve the function of detecting apple fruits and branches. The core function of the three-dimensional obstacle avoidance harvesting path planning method is to avoid obstacle points on the path on which it carries out picking and to automatically plan the end-effector running path of the robotic arm from the picking starting point to picking end point without collision, based on the obtained obstacle positions. Therefore, in the application of robot intelligent picking, it is necessary to combine apple/branch recognition algorithms to jointly carry out robot picking operations. Based on the fruit and branch detection algorithm, to obtain the spatial position of apples and branches, the obstacle avoidance path planning algorithm proposed in the study can

be used to choose a collision-free picking path, which can achieve obstacle avoidance in the picking of apples.

Our research proposed an obstacle avoidance picking path planning algorithm for the end-effector running path of an apple-picking robot. Due to the unique tree shape of apple trees, larger industrial robotic arms are not suitable for intelligent apple picking by robots. Regardless of the path planned to reach the fruit to be picked, oversized robotic arms will inevitably collide with tree branches during their movement into the tree crown. Therefore, the currently advanced flexible snake-like robotic arm is a nice choice for intelligent apple picking tasks.

6. Conclusions

(1) A three-dimensional spatial model for apple picking from spindle-shaped apple trees was established based on the three typical tree structures and planting characteristics of spindle-shaped apple trees. Inspired by the grid method, after calculating the three-dimensional spatial coordinates of tree trunks, the duplicate feature points of the model were removed. Utilizing the grid environment representation method, obstacle maps of apple tree trunk models were established.

(2) To address the problems of slow convergence speed and high search randomness in the early stage of 3D obstacle avoidance path planning using traditional ant colony algorithms, an improved ant colony algorithm was proposed in the study based on the original ACO algorithm. The initial pheromones were unevenly distributed, resulting in a higher concentration of pheromones in areas closer to the endpoint, which guided the ant colony to search for a target point in the early stage of path planning, solving the problem of large randomness in the early stage of planning. Secondly, the update rules for pheromones were improved, and thus the decay rate of pheromones was dynamically adjusted based on the length and height of the path.

(3) To address the problem of traditional ant colony algorithms being prone to getting stuck in local optima, the biomimetic optimization mechanism was integrated with the beetle antenna algorithm to perform quadratic planning on the basic global path obtained, in order to the improve speed and stability of path searching. After integrating the beetle antenna algorithm with the improved-design ant colony algorithm, not only did it ensure the global nature of the path, but the search efficiency was also improved. Finally, a cubic B-spline curve was utilized to smoothen the path composed of node connections, eliminating duplicate nodes in the path and preserving the effective path length. The path was made smoother, avoiding unnecessary pauses or turns during the harvesting process of the robotic arm, and the harvesting efficiency was improved.

(4) Three types of spindle-shaped apple trees were utilized for simulation experiments using the improved ACO algorithm for obstacle avoidance 3D path planning. The results showed that for free-spindle-shaped trees, the success rate of obstacle avoidance path planning was above 96%. For high-spindle-shaped trees, the success rate of path planning was above 86%. For slender-spindle-shaped trees, the success rate was above 92%.

(5) The 3D path planning performance of the improved ACO algorithm was compared with that of the traditional ACO algorithm, A* algorithm, and RRT algorithm for three types of spindle-shaped apple trees. The planning results of the algorithms were compared and analyzed from five perspectives: path length, planning success rate, planning time, path smoothness, and number of path nodes. The simulation experiments results showed that, compared with the traditional ACO algorithm, the improved ACO algorithm proposed in this study reduced the average planning time by 49.38% on the three-dimensional obstacle map of free-spindle apple trees. On the 3D obstacle map of high-spindle apple trees, the average path length was reduced by 0.06%, and the average planning time was reduced by 46.33%. On the three-dimensional obstacle map of slender-spindle-shaped apple trees, the average planning

time was decreased by 51.03%. On the other hand, compared with the traditional ACO algorithm, the average planning success rate of the proposed algorithm was improved, and the smoothness of the planned path was higher. Contrasting the other two algorithms, the improved ACO algorithm showed better overall performance in path planning experiments for the three different spindle-shaped apple trees. Overall, the performance of the proposed ACO algorithm was proven to be effective in three-dimensional path planning during obstacle avoidance fruit harvesting for spindle-shaped apple trees.

Author Contributions: All authors contributed to this manuscript. B.Y. and J.Q. contributed to the development of the algorithm, obtaining data, programming, and writing. B.Y. also contributed to conceptualization, supervision and funding acquisition. W.Y. contributed to original draft preparation and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the Key Research and Development Plan Project of Shaanxi Province (Grant No. 2021GY-084) and the Doctoral Research Project of Xi'an University of Technology (Grant No. 256082311).

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: We sincerely thank the editors and reviewers for their detailed comments and efforts toward improving our study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
2. Zhang, J.; Karkee, M.; Zhang, Q.; Zhang, X.; Yaqoob, M.; Fu, L.; Wang, S. Multi-class object detection using faster R-CNN and estimation of shaking locations for automated shake-and-catch apple harvesting. *Comput. Electron. Agric.* **2020**, *173*, 105384. [CrossRef]
3. Kang, H.; Chen, C. Fast implementation of real-time fruit detection in apple orchards using deep learning. *Comput. Electron. Agric.* **2020**, *168*, 105108. [CrossRef]
4. Kang, H.; Chen, C. Fruit detection, segmentation and 3D visualisation of environments in apple orchards. *Comput. Electron. Agric.* **2020**, *171*, 105302. [CrossRef]
5. Gene-Mola, J.; Sanz-Cortiella, R.; Rosell-Polo, J.R.; Morros, J.-R.; Ruiz-Hidalgo, J.; Vilaplana, V.; Gregorio, E. Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry. *Comput. Electron. Agric.* **2020**, *169*, 105165. [CrossRef]
6. Gao, F.; Fu, L.; Zhang, X.; Majeed, Y.; Li, R.; Karkee, M.; Zhang, Q. Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN. *Comput. Electron. Agric.* **2020**, *176*, 105634. [CrossRef]
7. Fu, L.; Majeed, Y.; Zhang, X.; Karkee, M.; Zhang, Q. Faster R-CNN-based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting. *Biosyst. Eng.* **2020**, *197*, 245–256. [CrossRef]
8. Fu, L.; Gao, F.; Wu, J.; Li, R.; Karkee, M.; Zhang, Q. Application of consumer RGB-D cameras for fruit detection and localization in field: A critical review. *Comput. Electron. Agric.* **2020**, *177*, 105687. [CrossRef]
9. Zhao, D.; Wu, R.; Liu, X.; Zhao, Y. Apple positioning based on YOLO deep convolutional neural network for picking robot in complex background. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 164–173. [CrossRef]
10. Wang, D.; He, D. Recognition of apple targets before fruits thinning by robot based on R-FCN deep convolution neural network. *Trans. Chin. Soc. Agric. Eng.* **2019**, *35*, 156–163. [CrossRef]
11. Kang, H.; Chen, C. Fruit detection and segmentation for apple harvesting using visual sensor in orchards. *Sensors* **2019**, *19*, 4599. [CrossRef]
12. Yan, B.; Fan, P.; Wang, M.; Shi, S.; Lei, X.; Yang, F. Real-time apple picking pattern recognition for picking robot based on improved YOLOv5m. *Trans. CSAM* **2022**, *53*, 28–38+59. [CrossRef]
13. Guo, Z.; Wu, X.; Yin, C.; Chen, Q.; Wang, J.; Zhou, H. Path planning of apple picking robot arm based on informed RRT*. *For. Mach. Woodwork. Eouipment* **2024**, *52*, 59–65. [CrossRef]
14. Dong, J.; Yi, B.; Li, Y. Six-axis robot apple picking path planning based on improved RRT* algorithm. *For. Mach. Woodwork. Eouipment* **2023**, *51*, 31–34. [CrossRef]
15. Zhang, C.; Wang, H.; Fu, L.; Pei, Y.; Lan, C.; Hou, H.; Song, H. Three-dimensional continuous picking path planning based on ant colony optimization algorithm. *PLoS ONE* **2023**, *18*, e0282334. [CrossRef]

16. Zhuang, M.; Li, G.; Ding, K. Obstacle avoidance path planning for apple picking robotic arm incorporating artificial potential field and A* algorithm. *IEEE Access* **2023**, *11*, 100070–100082. [CrossRef]
17. Li, H. A Visual Recognition and Path Planning Method for Intelligent Fruit-Picking Robots. *Sci. Program.* **2022**, *2022*, 1–9. [CrossRef]
18. Gao, R.; Zhou, Q.; Cao, S.; Jiang, Q. Apple-picking robot picking path planning algorithm based on improved PSO. *Electronics* **2023**, *12*, 1832. [CrossRef]
19. Kang, M.; Chen, Q.; Fan, Z.; Yu, C.; Wang, Y.; Yu, X. A RRT based path planning scheme for multi-DOF robots in unstructured environments. *Comput. Electron. Agric.* **2024**, *218*, 108707. [CrossRef]
20. Zheng, C.; Gao, P.; Hao, G.; Tian, Y.; Zhao, Y. Trajectory planning method for apple picking manipulator based on stepwise migration strategy. *Trans. CSAM* **2020**, *51*, 15–23. [CrossRef]
21. Zhang, Q.; Yue, X.; Li, B.; Jiang, X.; Xiong, Z.; Xu, C. Motion planning of picking manipulator based on CTB-RRT* algorithm. *Trans. CSAM* **2021**, *52*, 129–136. [CrossRef]
22. Wang, H. Research on task scheduling model of ant colony optimization cloud computing platform for online practical customer-training application. *IEIE Trans. Smart Process. Comput.* **2024**, *13*, 243–253. [CrossRef]
23. Starzec, G.; Starzec, M.; Rutkowski, L.; Kisiel-Dorohinicki, M.; Byrski, A. Ant colony optimization using two-dimensional pheromone for single-objective transport problems. *J. Comput. Sci.* **2024**, *79*, 102308. [CrossRef]
24. Shan, D.; Zhang, S.; Wang, X.; Zhang, P. Path-planning strategy: Adaptive ant colony optimization combined with an enhanced dynamic window approach. *Electronics* **2024**, *13*, 825. [CrossRef]
25. Prado-Rodriguez, R.; Gonzalez, P.; Banga, J.R.; Doallo, R. Improved cooperative Ant Colony Optimization for the solution of binary combinatorial optimization applications. *Expert Syst.* **2024**, *41*, e13554. [CrossRef]
26. Pourtousi, Z.; Babanezhad, M.; Ghanizadeh, A. Finding the link between iranian efl teacher motivation and engagement via ant colony optimization algorithm and fuzzy decision mode. *Integr. Psychol. Behav. Sci.* 2024; *Online ahead of print*. [CrossRef]
27. Liu, B.; Li, W.; Su, X.; Xu, X. An improved ACO based service composition algorithm in multi-cloud networks. *J. Cloud Comput. Adv. Syst. Appl.* **2024**, *13*, 17. [CrossRef]
28. Liang, W.; Lou, M.; Chen, Z.; Qin, H.; Zhang, C.; Cui, C.; Wang, Y. An enhanced ant colony optimization algorithm for global path planning of deep-sea mining vehicles. *Ocean Eng.* **2024**, *301*, 117415. [CrossRef]
29. Li, J.; He, P.; Li, H.; Li, S.; Xu, L.; He, K.; Yang, J.; Yang, J.; Wang, L. Multi-objective cutting parameter optimization method for the energy consumption and machining quality of computerized numerical control lathes. *Appl. Sci.* **2024**, *14*, 905. [CrossRef]
30. Cui, J.; Wu, L.; Huang, X.; Xu, D.; Liu, C.; Xiao, W. Multi-strategy adaptable ant colony optimization algorithm and its application in robot path planning. *Knowl. Based Syst.* **2024**, *288*, 111459. [CrossRef]
31. Ye, F.; Su, E.; Wei, Y.; Xu, C.; Liang, X. Investigation of esthetic evaluation and its influencing factors for a tunnel portal based on dynamic vision. *PLoS ONE* **2020**, *15*, e0238762. [CrossRef] [PubMed]
32. Fuyin, N.; Jian, H. Research on UPQC harmonic control strategy based on optimized QPIR controller of beetle antennae search algorithm in microgrid. *Electr. Eng.* **2024**, *106*, 2357–2369. [CrossRef]
33. Chen, C.; Cao, L.; Chen, Y.; Chen, B.; Yue, Y. A comprehensive survey of convergence analysis of beetle antennae search algorithm and its applications. *Artif. Intell. Rev.* **2024**, *57*, 141. [CrossRef]
34. Li, H.; Shen, J.; Zheng, L.; Cui, Y.; Mao, Z. Cost-efficient scheduling algorithms based on beetle antennae search for containerized applications in Kubernetes clouds. *J. Supercomput.* **2023**, *79*, 10300–10334. [CrossRef]
35. Kim, J.; Jo, H.; Ri, J.; Han, K. Automatic fabric defect detection using optimal Gabor filter based on hybrid beetle antennae search–gravitational search algorithm. *J. Opt.* **2023**, *52*, 1667–1675. [CrossRef]