Collision-Free Motion Planning of a Six-Link Manipulator Used in a Citrus Picking Robot

Zuoliang Tang, Lijia Xu *, Yuchao Wang, Zhiliang Kang and Hong Xie

Abstract: This paper presents the results of a motion planning algorithm that has been used in an intelligent citrus-picking robot consisting of a six-link manipulator. The real-time performance of a motion planning algorithm is urgently required by the picking robot. Within the artificial potential field (APF) method, the motion planning of the picking manipulator was basically solved. However, the real-time requirement of the picking robot had not been totally satisfied by APF because of some native defects, such as the large number of calculations used to map forces into torques by the Jacobian matrix, local minimum trap, and target not reachable problem, which greatly reduce motion planning efficiency and real-time performance of citrus-picking robots. To circumvent those problems, this paper proposed some novel methods that improved the mathematical models of APF and directly calculates the attractive torques in the joint space. By using the latter approach, the calculation time and the total joint error were separately reduced by 54.89% and 45.41% compared with APF. Finally, the novel algorithm is presented and demonstrated with some illustrative examples of the citrus picking robot, both offline during the design phase as well as online during a realistic picking test.

Keywords: citrus picking robot; manipulator; motion planning; artificial potential field

1. Introduction

Citrus fruit with rich nutritional value and good taste is loved by people all over the world. According to the statistics of the U.S. Department of Agriculture in April 2020 [1], China is the second-largest citrus producer in the world. According to the statistical data of the National Bureau of Statistics of China [2], the annual output of citrus in China shows an increasing trend from 2011 to 2020, and the total output of citrus in 2020 was 5.219 million t. However, at present, citrus fruit is mainly hand-picked, which is inefficient and costly. According to the investigation, the labor force of citrus picking operations accounts for 33–50% of the whole production process. Thus, a picking robot becomes an effective way to alleviate this situation [3–5]. Picking robots have long been a research hotspot in the field of intelligent agricultural equipment, and China, the United States, Japan, and other countries have made many breakthrough achievements in this field [6–11]. Due to the high operability and precision of the manipulator, most studies have adopted a manipulator as the main execution component of a picking robot.

If a manipulator is used as the main executive part of a citrus-picking robot, the motion planning of the manipulator needs to be solved first. Motion planning refers to the trajectory planning of each joint in the process of moving the manipulator from the initial configuration to the target configuration. During this process, it must be ensured that the manipulator does not collide with any obstacle in the workspace or itself. Existing motion planning algorithms include rapidly random-exploring tree (RRT) [12–15], probabilistic roadmap method (PRM) [16,17], A* [18], neural network [19], and artificial potential field (APF) [20–23]. RRT is the most popular algorithm among scientists all over the world to
process motion planning problems, and the related algorithms include RRT*, Informed-RRT, RRT-Connect, and so on. RRT gives a path in the workspace by randomly walking and continuously expanding its leaf nodes. This process will not end till the leaf nodes spread throughout the whole workspace, so this algorithm usually takes a long time and is not suitable for a picking robot because of the real-time requirement. Generally, RRT has a good effect on the motion planning problem of mobile vehicles, but it is not feasible to directly use RRT for a manipulator in Cartesian space, because a manipulator cannot be regarded as a single particle. Therefore, some scholars proposed to adopt RRT to carry out motion planning in the configuration space (C-space) of manipulators, in which a point represents a single configuration instance. RRT divides C-space into free and obstructed space and carries out motion planning in the free space. Because a picking robot works in an unstructured and greatly complex environment with many obstacles, the C-space construction of a manipulator would take an extremely long time. From the existing literature, PRM and A* algorithms also need to build C-space, so RRT, PRM, and A* cannot meet the real-time requirements of a picking robot. The neural network algorithm not only needs a long time to train itself, but also has higher requirements on hardware, so it cannot be used for real-time motion planning of a picking robot as well.

APF was firstly proposed by O. Khatib [23] in 1986 and was applied to real-time dynamic obstacle avoidance motion planning of a manipulator, PUMA-560, and a mobile vehicle. APF plans the motion by building a virtual potential field in the workspace which is a Cartesian space, in which moving objects are affected by two kinds of forces—attractive force, and repulsive force—and avoid crashing into obstacles during the process of moving from the initial to the target position. Different from global planning algorithms such as RRT and PRM, APF is a local motion planning algorithm with low algorithm complexity, low dependence on computer resources, and strong real-time performance. However, APF has problems such as local minimum trap (LMT) [24,25] and target not reachable problem (TNRP) [26]. LMT means that when the object has not reached the target position, the gradient of the potential field of APF disappears, then the resultant force becomes zero, and the object in the potential field no longer can reach the target position, which is the main problem and shortcoming of APF. TPNR refers to that when there is an obstacle near the target position, the attraction is far less than the repulsive force, and the object in the potential field cannot overcome the repulsive force to reach the target position under the action of the attractive force. TPNR is caused by the gradual loss of attractive force near the target position which is another main problem and shortcoming of APF. Matouf, F. et al. [27] designed a non-minimum speed algorithm to improve APF, and the improved APF can effectively avoid the occurrence of the LMT. Jeon, G. et al. [28] proposed a water sink model for path planning of planar robots aiming at solving LMT. Although the above studies have effectively avoided the LMT, the application object is a mobile vehicle rather than a manipulator, so it cannot be directly applied to a picking robot for real-time motion planning. A large number of scholars have also carried out lots of relevant studies on the application of APF in manipulators. Zhou, H. et al. [26] improved the repulsive force calculation function of APF and combined it with the cosine adaptive genetic algorithm. Motion planning for the end effector of a 5-DOF manipulator needed 8.72 and 8.31 s respectively in conventional and LMT conditions, and the time was reduced by 51.6% and 53.85% compared with the original APF. Zhang, H. et al. [20] took 103.421 s to carry out path planning for a dual-manipulator in a static environment based on the potential field of velocity, and the time was 16.85% less than that of APF. Park, S. O. et al. [29] improved the APF algorithm based on the Jacobian matrix and carried out path planning for a 7-DOF manipulator to avoid spherical and cylindrical obstacles respectively, which both took 7 s, and the single iteration took 1.925 and 2.5 ms respectively. Zhang, L. [30] combined RRT with APF and proposed a velocity potential field method for obstacle avoidance motion planning of manipulators based on virtual target points. It could effectively avoid falling into LMT, but the path planning for a planar 3-DOF manipulator took at least 4.15 s. Wang, W. et al. [31] improved the attractive potential field calculation formula of APF by
Introducing a Cartesian space boundary, but it took more than 5 s for path planning of self-designed 9-DOF manipulator to get very close to the target configuration. Although the above studies successfully applied APF to path planning of manipulators and effectively avoided the emergence of LMT, LMT, and TPNR were not considered at the same time. In addition, due to the high real-time requirements of a picking robot on the motion planning algorithm, the above research results are all time-consuming and cannot meet the real-time requirements of picking robots.

Because of the problems mentioned above, this paper improved the calculation model of both potential fields and forces of APF respectively, and directly calculated the attractive torques in the joint space of a manipulator when planning the motion of a manipulator and proposed a novel improved APF (IAPF) which can effectively reduce the calculation time of motion planning for manipulators and avoid LMT and TPNR problems at the same time. Finally, the correctness and effectiveness of IAPF in manipulators were verified by some illustrative experiments both in the design and realistic picking phase.

The rest of this paper is organized as follows: in Section 2, a short description of the material and method in this study is given, and the Elementary Transformation Sequence (ETS) method is used to construct the kinematics model of the EC63M manipulator; in Section 3, the calculation model of potential fields and forces of APF is given, and the method of applying APF on a manipulator is described as well. Furthermore, the optimization strategies in IAPF are introduced carefully. In Section 4, some illustrative experiments are shown and, in Section 5 a short conclusion of the whole paper is given.

2. Materials and Methods

2.1. Material

Figure 1 shows a functional model of the intelligent citrus picking robot (ICPR), which consists of three main components: the automatic mobile vehicle which is responsible for the movement in the citrus orchard and serves as a platform for carrying all the other components; the light industrial six-link manipulator, EC63M, produced by Jiangsu Elite Company in China, which is used for picking citrus fruits; the frame structure mounted on the vehicle which is used for placing robot control cabinet; the IPC; citrus storage box; mobile power supplies; and stereo camera vision systems for detection and localization of citrus fruits. Figure 2 shows the conceptual diagram for motion planning of the EC63M manipulator.

Figure 1. Intelligent citrus picking robot.
2.2. Methods

The establishment of the manipulator’s kinematic model is the premise work of motion planning. Common kinematic modeling methods include Denavit–Hartenberg (DH) method [32] which includes the standard DH method [33] and the improved DH method [34], as well as the ETS method [35]. The DH method describes each link of a manipulator by establishing a DH coordinate frame. Because the DH method has only a few indispensable coordinate frames to describe all links of a manipulator, the description of most points on the manipulator instead of DH coordinate frame origins requires complex geometric transformation and mathematical calculation, which cannot meet the requirements of APF in both real-time performance and complete description of a manipulator. Therefore, ETS is used to conduct kinematics modeling of EC63M in this paper, because of the full description of all the points, as shown in Figure 3. The notations used in the ETS model are explained in Table 1. From the ETS model, the Jacobian matrix of every point on the manipulator can be easily obtained, see the paper [35] for more details.

![Figure 2. Conceptual diagram for motion planning.](image1)

![Figure 3. ETS model of EC63M manipulator.](image2)
Table 1. Notations used in the ETS model.

| Notation | Meaning | Notation | Meaning |
|----------|---------|----------|---------|
| $d_i$    | link offset          | $O_i X_i Y_i Z_i$ | ETS coordinate frame |
| $a_i$    | link length           | $E_j \in \mathbb{R}^{4 \times 4}$ | ETS matrix |
| $a_i$    | joint angle           | $T^{j-1}_{i,j} \in \mathbb{R}^{4 \times 4}$ | transformation matrix form frame $j$ to $j-1$ |
| $\theta_i$ | translational matrix along $Z_i$ | $T_{R_i}(\theta_i) \in \mathbb{R}^{4 \times 4}$ | rotation matrix around $X_i$ |
| $T_{R_i}(d_i) \in \mathbb{R}^{4 \times 4}$ | rotation matrix around $Z_i$ | |

Figure 4 shows that, in this paper, it needs three steps to pick a single citrus fruit. Step 1: the picking manipulator moves from the initial configuration to the picking configuration, and the target citrus fruit is detached by the end-effector. The picking configuration is designed and calculated simultaneously when planning the movement. Step 2: The picking manipulator moves from the picking configuration to the release configuration to put the fruit into the storage box. Step 3: the picking manipulator moves from the release configuration to the initial configuration and waits for the next picking cycle. Both the initial and release configurations are pre-set in the picking experiment to further increase the real-time performance.

Figure 4. Flow diagram of motion planning for the picking manipulator.

3. Theory and Calculation

This section describes the motion planning method for a citrus picking manipulator and its optimization approaches. The notations used in this section are listed in Table 2.

Table 2. Notations used in the motion planning method.

| Notation | Meaning | Notation | Meaning |
|----------|---------|----------|---------|
| $U_a \in \mathbb{R}$ | attractive potential field | $U_r \in \mathbb{R}$ | repulsive potential field |
| $k_a$ | attractive coefficient | $k_r$ | repulsive coefficient |
| $F_a \in \mathbb{R}^3$ | attractive force | $F_r \in \mathbb{R}^3$ | repulsive force |
| $P_c \in \mathbb{R}^3$ | current position | $P_o \in \mathbb{R}^3$ | obstacle position |
| $P_i \in \mathbb{R}^3$ | goal position | $d_0$ | distance threshold |
| $U_{d_r} \in \mathbb{R}$ | resultant potential field | $F_r \in \mathbb{R}^3$ | resultant force |
| $d_r$ | distance to obstacle | $\nabla d_r$ | gradient of $d_r$ |
| $d_s$ | distance to goal | $F_o \in \mathbb{R}^3$ | repulsive force of obstacle |
| $O_i \in \mathbb{R}^3$ | origin of frame $i$ | $d_\Theta$ | gradient of $d_\Theta$ |
| $P_o \in \mathbb{R}^3$ | the nearest point to the obstacle of a manipulator | $\nabla d_\Theta$ | configuration error |
| $\Gamma \in \mathbb{R}^6$ | joint torque | $U_{r,i} \in \mathbb{R}$ | the repulsive potential field of obstacle $i$ |
| $0_J^T(\Theta) \in \mathbb{R}^{6 \times 6}$ | transpose of Jacobian matrix of $O_i$ | $F_{r,i} \in \mathbb{R}^3$ | repulsive force of obstacle $i$ |
| $0_P^T(\Theta) \in \mathbb{R}^{6 \times 6}$ | transpose of Jacobian matrix of $P_m$ | $F_r \in \mathbb{R}^3$ | temporary force |
| $k_{s}\Sigma$ | compensation coefficient of $U_r$ | $U_{r,\max}, k_{s1}, k_{s2}$ | deformation control factors of the Sigmoid function |
3.1. Motion Planning for Manipulator

APF plans a trajectory by establishing the virtual potential fields to avoid collision with any obstacle in the workspace. The attractive field and force are calculated as

\[ U_a = \frac{1}{2} k_a (d_a)^2, \]

\[ F_a = -\nabla U_a = k_a d_a (\nabla d_a), \]

The repulsive field and force are calculated as

\[ U_r = \begin{cases} \frac{1}{2} k_r \left( \frac{1}{d_r} - \frac{1}{d_0} \right)^2, & d_r \leq d_0 \\ 0, & d_r > d_0 \end{cases}, \]

\[ F_r = -\nabla U_r = \begin{cases} k_r \left( \frac{1}{d_r} - \frac{1}{d_0} \right) \frac{1}{(d_r)^2} \nabla d_r, & d_r \leq d_0 \\ 0, & d_r > d_0 \end{cases}. \]

The resultant potential field and force are calculated as

\[ U_s = U_a + U_r, \]

\[ F_s = F_a + F_r. \]

Different from the mobile vehicle, the application of APF and the obstacle avoidance motion planning of the manipulator is more complicated. The moving car can be regarded as a point in the workspace for motion planning, but for the manipulator, it is necessary to consider how to carry out an obstacle avoidance motion planning for the whole manipulator, rather than as a single point. Path planning for the manipulator can be done in its workspace or the C-space. Due to a large amount of computation and time-consuming construction of C-space, path planning in the workspace is usually used. Since the motion of each point on the manipulator cannot be individually planned, points on the manipulator must be considered in the context of an overall robotic arm structure. In this research, the attractive forces are applied to the origins of all the ETS coordinate frames, and the repulsive forces are applied to the nearest point to the obstacle on each ETS line segment, as shown in Figure 5. By attractive forces, the manipulator can change from initial configuration to goal configuration. By repulsive forces, it can avoid crashing into any obstacle when moving in the workspace.

When there are multiple obstacles in the workspace, the resultant field force can be calculated as

\[ U_s = U_a + \sum_i U_{r,i}, \]

\[ F_s = F_a + \sum_i F_{r,i}. \]

To control the manipulator, the resultant force is finally mapped to the joint torque through the transpose of the Jacobian matrix. The formula to calculate joint torques is

\[ \Gamma = 0_{0} J_T(\Theta) \cdot F_a + \sum_{i} 0_{0} J_T(\Theta) \cdot F_{r,i}. \]
Different from the mobile vehicle, the application of APF and the obstacle avoidance task are different. In this section, some optimization strategies are proposed.

This phenomenon directly leads to two problems that $F_a$ decreases linearly as $d_a$ decreases and decreases to zero when $d_a$ is equal to zero, but $F_r$ tends to increase with the decrease of $d_r$ and increases to infinite when $d_r$ is equal to zero. First, if there is an obstacle near the target position, the repulsive force will be far greater than the attractive force. When the object approaches the target position, it will be constantly bounced away by the repulsive force, which leads to path oscillation and even failure, which is the TPNR problem. Second, the torque that the joint of the manipulator can bear is limited, and the infinite repulsion force cannot be realized in the motion planning process. To solve the above two problems and the LMT, this paper optimized the models of potential fields and forces.

To avoid the TPNR problem, it is necessary to still have a large gradient close to the target position, to ensure that $F_a$ is large enough to overcome the influence of $F_r$ to continuously guide the object to the target position. At the same time, to make the path planning algorithm converge, when $d_a$ is equal to zero, $F_a$ should also trend to approach zero. For this reason, IAPF adds the Sigmoid function component based on the original attractive potential field calculation model of APF. The attractive potential field and attractive force of IAPF are calculated as

$$U_a = \frac{U_{a,\text{max}}}{1 + e^{-(k_{a1}d_a - k_{a2})}} + \frac{1}{2}k_{a3}d_a^2,$$

$$F_a = -\nabla U_a = \left(\frac{-k_{a1}U_{a,\text{max}}}{(1 + e^{-(k_{a1}d_a - k_{a2})})}e^{-(k_{a1}d_a - k_{a2})} - k_{a3}d_a\right)\nabla d_a.$$

To avoid the problem of uncontrollable increase of $F_r$ with the decrease of $d_r$, IAPF changed the original repulsive potential field model. The repulsive potential field model of APF is a piecewise function. It not only needs to be continuous and differentiable at the piecewise junction but also a gradient explosion will occur when $d_r$ is equal to zero. To
ensure that the repulsive force of IAPF does not increase infinitely and is still continuous and differentiable with the decrease of \( d_r \), the repulsive force of IAPF is changed into an anti-S Sigmoid function in this paper. Thus, the function of repulsive force of IAPF not only approaches zero when it is far from the obstacle, but also increases with the decrease of \( d_r \), and is still continuous and differentiable. The Fr of IAPF will gradually increase and tend to the maximum repulsive force, and there will not be the problem of repulsive force explosion. The repulsive potential field and repulsive force of IAPF are calculated as

\[
U_r = -U_{r,max}d_r + \frac{U_{r,max}\log\left(e^{k_3d_r-k_2}+1\right)}{k_{r1}} + k_{r3},
\]

\[
\mathbf{F}_r = -\nabla U_r = \begin{cases} \frac{U_{r,max}}{1+e^{-k_3d_r-k_2}}\nabla d_r, & d_r \leq d_0 \\ 0, & d_r > d_0 \end{cases},
\]

To eliminate the influence brought by LMT, IAPF chooses to temporarily add a force \( \mathbf{F}_t \) to break the balance of repulsion and attraction when the gradient of the resultant potential field disappears. \( \mathbf{F}_t \) not only determines the quality of the result of IAPF but also determines whether IAPF will fall into LMT again. Through the simulation experiments, \( \mathbf{F}_t \) has a good effect when it is perpendicular to both \( \mathbf{F}_r \) and \( \mathbf{F}_n \). In this paper, when IAPF is about to get into LMT, \( \mathbf{F}_t \) will appear, and its value is equal to \( \mathbf{F}_n \), and its direction is shown in Figure 6.

\[
\mathbf{F}_r = \mathbf{F}_n = 0
\]

\[
\mathbf{F}_r = \mathbf{F}_t
\]

(a) (b)

Figure 6. Diagram of temporary force to avoid LMT. (a) No temporary force; (b) With a temporary force.

It still requires a large amount of calculation to directly map the forces in the workspace to torques in the joint space through the transpose of the Jacobian matrix, so for further optimization, IAPF directly calculates the attractive torques in the joint space. The joint torques are calculated as

\[
\mathbf{\Gamma} = \mathbf{\Gamma}_a + \sum_i^0 p_{ai} J_i^T(\Theta) \cdot \mathbf{F}_{r,i},
\]

\[
\mathbf{\Gamma}_a = \left( -k_{a1}U_{a,max} + \frac{U_{a,max}}{1+e^{-(k_{a2}d_\Theta-k_2)}} e^{-(k_{a1}d_\Theta-k_2)} - k_{a1}d_\Theta \right) (\nabla d_\Theta),
\]

4. Experiments and Results

All the experiments mentioned in this paper are implemented in the Visual Studio Code through Python 3 in Windows 10 operating system. The hardware conditions are as follows: CPU is I7-9700F, the mainboard is B365M-PXiU, 16GB memory, and the graphics card is a NVIDIA Quadro P2000.
4.1. Field Surfaces and Force Curves

The potential fields’ surfaces and forces’ curves of both APF and IAPF are shown in Figures 7 and 8 respectively. Table 3 shows the concrete parameters of the potential field and force model of APF and IAPF.

Table 3. The concrete parameters used in experiment Section 4.1.

| Parameter | Parameter | Parameter | Parameter |
|-----------|-----------|-----------|-----------|
| $x, y \in [0, 10]$ | $k_r = 1$ | $P_c = [x, y]$ | $k_{a1} = 5$ |
| $P_g = [0, 0]$ | $k_{a2} = 10$ | $P_d = [10, 10]$ | $U_{r,\text{max}} = 50$ |
| $P_o = [3, 3]$ | $k_a = 1$ | $d_0 = 5$ | $k_{r1} = 4$ |
| $k_a = 1$ | $k_{r2} = 8$ | $U_{a,\text{max}} = 50$ | $k_{r3} = 100$ |

Figure 7. Potential fields’ surfaces. (a) Attractive potential field of APF; (b) Repulsive potential field of APF; (c) Resultant potential field of APF; (d) Attractive potential field of IAPF; (e) Repulsive potential field of IAPF; (f) Resultant potential field of IAPF.

The gradient of APF’s attractive potential field gradually disappears near the target position, while the gradient of IAPF does not disappear but increases. Therefore, IAPF has stronger resistance to TPNR and a faster convergence rate than APF. Due to the phenomenon of gradient explosion in the repulsive potential field of APF, the gradient of the repulsive potential field of APF is much larger than that of the attractive potential field when it is close to an obstacle, resulting in the lack of features of the surface of the attractive potential field of the resultant potential field. However, IAPF can effectively overcome the above problems. Its resultant potential field surface has the common characteristics of both attractive and repulsive potential field surface, which makes IAPF not only have a faster convergence rate but also have the strong obstacle-avoidance ability.
4.2. TPNR Verification

When there are obstacles near the target position, APF has a TPNR problem. Therefore, the following simulation experiment is conducted to verify that IAPF can effectively avoid TPNR. The simulation area is a 10 × 10 rectangle and the concrete parameters are given in Table 4, and the action threshold of each obstacle is equal. The simulation results are shown in Figure 9, and the simulation curve is shown in Figure 10.

As can be seen from Figure 9, when there is an obstacle near the target position, the object keeps approaching the target position, but is pushed away by the repulsive force of the obstacle and oscillates back and forth, unable to reach the target position by APF. Therefore, APF cannot converge. As can be seen from Figure 10, IAPF approaches the target position near the 20th iteration and obtains an enhanced attractive force, avoiding the occurrence of TPNR. Simulation results show that the improved attractive force of the IAPF algorithm can effectively avoid the occurrence of TPNR and ensure the success rate of path planning.

Table 4. The concrete parameters used in experiment Section 4.2.

| Parameter | Parameter | Parameter | Parameter |
|-----------|-----------|-----------|-----------|
| \( P_s = [0, \ 0] \) | \( P_t = [10, \ 10] \) | \( P_o1 = [2, \ 4] \) | \( P_o2 = [5, \ 7] \) |
| \( P_o3 = [2, \ 2] \) | \( d_o = 2 \) | | |

Figure 8. Force curves. (a) Attractive force curve of APF; (b) Repulsive force curve of APF; (c) Attractive force curve of IAPF; (d) Repulsive force curve of IAPF.
The gradient of APF’s attractive potential field gradually disappears near the target position, while the gradient of IAPF does not disappear but increases. Therefore, IAPF has stronger resistance to TPNR and a faster convergence rate than APF. Due to the phenomenon of gradient explosion in the repulsive potential field of APF, the gradient of the repulsive potential field of APF is much larger than that of the attractive potential field when it is close to an obstacle, resulting in the lack of features of the surface of the attractive potential field of the resultant potential field. However, IAPF can effectively overcome the above problems. Its resultant potential field surface has the common characteristics of both attractive and repulsive potential field surface, which makes IAPF not only have a faster convergence rate but also have the strong obstacle-avoidance ability.

4.2. TPNR Verification

When there are obstacles near the target position, APF has a TPNR problem. Therefore, the following simulation experiment is conducted to verify that IAPF can effectively avoid TPNR. The simulation area is a 10 × 10 rectangle and the concrete parameters are given in Table 4, and the action threshold of each obstacle is equal. The simulation results are shown in Figure 9, and the simulation curve is shown in Figure 10.

Table 4. The concrete parameters used in experiment Section 4.2.

| Parameter | Value |
|-----------|-------|
| $g$       | $P_{10,10}$ |
| $i$       | $P_{5,5}$ |
| $o$       | $P_{2,2}$ |
| $d$       | $a \times b$ |

Figure 9. Simulation results of TPNR. (a) Results of APF; (b) Results of IAPF.

As can be seen from Figure 9, when there is an obstacle near the target position, the object keeps approaching the target position, but is pushed away by the repulsive force of the obstacle and oscillates back and forth, unable to reach the target position by APF. Therefore, APF cannot converge. As can be seen from Figure 10, IAPF approaches the target position near the 20th iteration and obtains an enhanced attractive force, avoiding the occurrence of TPNR. Simulation results show that the improved attractive force of the IAPF algorithm can effectively avoid the occurrence of TPNR and ensure the success rate of path planning.

4.3. LMT Verification

To verify that the IAPF algorithm can avoid getting into the LMT problem, the following simulation experiment is designed in this section: It is easy to get into the LMT problem when there are obstacles on the line between the target and the initial position. Therefore, APF and IAPF carry out path planning in this case simultaneously. The simulation area is a 10 × 10 rectangle, and the concrete parameters are given in Table 5 and the action threshold of each obstacle is equal. The simulation results are shown in Figure 11, and the simulation curve is shown in Figure 12.

Table 5. The concrete parameters used in experiment Section 4.3.

| Parameter | Value |
|-----------|-------|
| $g$       | $P_{10,10}$ |
| $i$       | $P_{5,5}$ |
| $d$       | $a \times b$ |

Figure 10. Simulation curve of TPNR. (a) Distance curve to goal; (b) Distance curve to the obstacle; (c) Attractive force curve.; (d) Repulsive force curve.
4.3. LMT Verification

To verify that the IAPF algorithm can avoid getting into the LMT problem, the following simulation experiment is designed in this section: It is easy to get into the LMT problem when there are obstacles on the line between the target and the initial position. Therefore, APF and IAPF carry out path planning in this case simultaneously. The simulation area is a $10 \times 10$ rectangle, and the concrete parameters are given in Table 5 and the action threshold of each obstacle is equal. The simulation results are shown in Figure 11, and the simulation curve is shown in Figure 12.

Table 5. The concrete parameters used in experiment Section 4.3.

| Parameter | Parameter | Parameter | Parameter |
|-----------|-----------|-----------|-----------|
| $P_g = [0, 0]$ | $P_i = [10, 10]$ | $P_o = [5, 5]$ | $d_0 = 2$ |

![Figure 11. Simulation results of LMT. (a) Results of APF; (b) Results of IAPF.](image)

![Figure 12. Simulation curve of LMT. (a) Distance curve to goal; (b) Distance curve to the obstacle; (c) Attractive force curve.; (d) Repulsive force curve.](image)
As can be seen from the simulation results, APF quickly falls into the LMT and remains in this state until the path planning fails. IAPF did not fall into the LMT and quickly reached the target position. IAPF reached its target location near iteration 22, while APF remained in the LMT. The simulation results show that IAPF can overcome the LMT problem effectively and ensure the success rate of path planning.

### 4.4. Application in EC63M Manipulator

In this section, APF and IAPF algorithms were used to carry out path planning for the EC63M manipulator. The experimental parameters were shown in Table 6, in which the initial configuration was $\Theta_i$, the target configuration was $\Theta_g$, and the obstacle coordinate was $P_{Oi}$, the obstacle radii were all 120 mm. The simulation results are shown in Table 7. The joint angle curves and the distance curves between each ETS line segment of the EC63M manipulator and the obstacles are shown in Figure 13.

The simulation results show that IAPF is better than APF. The joint angles’ curves of IAPF are smoother and the change rate is almost the same. The joint angles’ curves of the APF algorithm change more obviously, and the change rate is quite different. The joint angle $\theta_3$ begins to converge after a reverse process, while the joint angle $\theta_2$ shows a wave peak. The convergence speed of APF is very slow. Compared with IAPF, it directly reaches the target configuration at the 238th time, but APF converges at the 549th time. Compared with APF, IAPF reduces the operation time by 54.89% and the total joint error by 45.41% in terms of running speed and accuracy.

### Table 6. The concrete parameters used in experiment Section 4.4.

| Parameter | Parameter | Parameter |
|-----------|-----------|-----------|
| $\Theta_i = \left[ \begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]$ | $\Theta_g = \left[ \begin{array}{cccc} \frac{\pi}{2} & -\frac{\pi}{2} & \frac{\pi}{2} & 0 & 0 & 0 \end{array} \right]$ | $P_{O1} = \left[ \begin{array}{c} -400 \ 100 \ 300 \end{array} \right]$ |
| $P_{O2} = \left[ \begin{array}{c} -400 \ 100 \ 400 \end{array} \right]$ | $P_{O3} = \left[ \begin{array}{c} 400 \ 100 \ 450 \end{array} \right]$ | $P_{O4} = \left[ \begin{array}{c} 400 \ 100 \ 500 \end{array} \right]$ |
| $P_{O5} = \left[ \begin{array}{c} 400 \ 100 \ 550 \end{array} \right]$ | $P_{O6} = \left[ \begin{array}{c} 400 \ 100 \ 300 \end{array} \right]$ | $P_{O7} = \left[ \begin{array}{c} 400 \ 100 \ 500 \end{array} \right]$ |

### Table 7. The path planning results of IAPF and APF applied to EC63M.

| Algorithm | Iterations | Runtime/s | Joint Error/° | Total Error/° |
|-----------|------------|-----------|---------------|---------------|
| APF       | 549        | 2.35      | $[-0.03, 0.1, -0.30, 0.30, -1.79, 0.00]$ | 1.85 |
| IAPF      | 238        | 1.06      | $[-0.39, 0.8, -0.39, 0.02, 0.00, 0.00]$ | 1.01 |
reached the target position. IAPF reached its target location near iteration 22, while APF remained in the LMT. The simulation results show that IAPF can overcome the LMT problem effectively and ensure the success rate of path planning.

4.4. Application in EC63M Manipulator

In this section, APF and IAPF algorithms were used to carry out path planning for the EC63M manipulator. The experimental parameters were shown in Table 6, in which the initial configuration was $\Theta_i$, the target configuration was $\Theta_g$, and the obstacle coordinate was $P_O$, the obstacle radii were all 120 mm. The simulation results are shown in Table 7. The joint angle curves and the distance curves between each ETS line segment of the EC63M manipulator and the obstacles are shown in Figure 13.

Table 6. The concrete parameters used in experiment Section 4.4.

| Process  | Iterations | Runtime/s | Joint Error/$^\circ$ | Total Error/$^\circ$ |
|----------|------------|-----------|---------------------|-------------------|
| Picking  | 113        | 0.40      | [0.29, −0.55, 0.04, 1.16, −0.20, 0.04] | 1.33 |
| Releasing | 315        | 1.12      | [−0.11, 0.09, 0.19, −0.15, 1.06, 0.00] | 1.10 |
| Initialization | 299 | 1.04 | [0.02, 0.08, −0.22, −0.21, −1.09, −0.01] | 1.13 |

Figure 13. Experiments results of motion planning of the EC63M manipulator. (a) The joint trajectory by APF; (b) The joint trajectory by IAPF; (c) Distance to the obstacle of APF; (d) Distance to the obstacle of IAPF.

4.5. Application in Citrus Picking Robot

To further verify the effectiveness and feasibility of IAPF, in this section, the motion planning experiment was carried out on the ICPR and experimental results were shown in Table 8. The target citrus tree and its simplified model are shown in Figure 14. Firstly, the target citrus tree was modeled by the partial spherical envelope (PSE) method which was like a bounding sphere algorithm [36,37]. PSE method only includes sphere and line segment models, which can greatly reduce the amount of calculation in the process of collision detection, to further ensure that IAPF can meet the real-time requirements of ICPR. PSE envelops all non-current picking parts of citrus trees through spheres and models branches in the current picking part through straight segments. The PSE model of picking three oranges on the top of the target citrus tree is presented in Figure 15, the picking simulation of the ICPR and the trajectory curves are shown in Figure 16, the experiment results of IAPF applied to the ICPR are shown in Figure 17.

Table 8. Simulation experiment results of IAPF applied to ICPR.

| Process  | Iterations | Runtime/s | Joint Error/$^\circ$ | Total Error/$^\circ$ |
|----------|------------|-----------|---------------------|-------------------|
| Picking  | 113        | 0.40      | [0.29, −0.55, 0.04, 1.16, −0.20, 0.04] | 1.33 |
| Releasing | 315        | 1.12      | [−0.11, 0.09, 0.19, −0.15, 1.06, 0.00] | 1.10 |
| Initialization | 299 | 1.04 | [0.02, 0.08, −0.22, −0.21, −1.09, −0.01] | 1.13 |
The simulation results show that IAPF is better than APF. The joint angles’ curves of IAPF are smoother and the change rate is almost the same. The joint angles’ curves of the APF algorithm change more obviously, and the change rate is quite different. The joint angle $\theta_3$ begins to converge after a reverse process, while the joint angle $\theta_2$ shows a wave peak. The convergence speed of APF is very slow. Compared with IAPF, it directly reaches the target configuration at the 238th time, but APF converges at the 549th time. Compared with APF, IAPF reduces the operation time by 54.89% and the total joint error by 45.41% in terms of running speed and accuracy.

Table 7. The path planning results of IAPF and APF applied to EC63M.

| Algorithm | Iterations | Runtime/s | Joint Error/° | Total Error/° |
|-----------|------------|-----------|---------------|--------------|
| APF       | 549        | 2.35      | $[-0.03, 0.1, -0.30, 0.30, -1.79, 0.00]$ | 1.85         |
| IAPF      | 238        | 1.06      | $[-0.39, 0.8, -0.39, 0.02, 0.00, 0.00]$ | 1.01         |

4.5. Application in Citrus Picking Robot

To further verify the effectiveness and feasibility of IAPF, in this section, the motion planning experiment was carried out on the ICPR and experimental results were shown in Table 8. The target citrus tree and its simplified model are shown in Figure 14. Firstly, the target citrus tree was modeled by the partial spherical envelope (PSE) method which was like a bounding sphere algorithm [36,37]. PSE method only includes sphere and line segment models, which can greatly reduce the amount of calculation in the process of collision detection, to further ensure that IAPF can meet the real-time requirements of ICPR. PSE envelops all non-current picking parts of citrus trees through spheres and models branches in the current picking part through straight segments. The PSE model of picking three oranges on the top of the target citrus tree is presented in Figure 15, the picking simulation of the ICPR and the trajectory curves are shown in Figure 16, the experiment results of IAPF applied to the ICPR are shown in Figure 17.

Figure 14. Target citrus tree and its PSE model. (a) Target citrus tree; (b) PSE model of the target citrus tree.

Figure 15. Picking part of target citrus tree and its PSE model. (a) Amplification 1 of picking part; (b) Amplification 2 of picking part; (c) Amplification 3 of picking part; (d) Picking part; (e) Fitting the branches; (f) PSE model of picking part.
Figure 15. Picking part of target citrus tree and its PSE model. (a) Amplification 1 of picking part; (b) Amplification 2 of picking part; (c) Amplification 3 of picking part; (d) Picking part; (e) Fitting the branches; (f) PSE model of picking part.

Figure 16. The trajectory curves in picking simulation of the ICPR. (a) Joint trajectory of EC63M during picking stage; (b) Joint trajectory of EC63M during releasing stage; (c) Joint trajectory of EC63M during initialization stage.

IAPF took 0.40 s, 1.12 s, and 1.04 s respectively in the process of path planning for ICPR picking citrus 2 on the top of the target citrus tree, and the total joint errors were 1.33, 1.10, and 1.13 degrees respectively in the stages of picking, releasing, and initialization. ICPR took 2.56 s to plan a path to pick single citrus. Since the time spent on the EC63M mechanical arm’s movement is much longer than the time spent in path planning, the IAPF algorithm can fully plan the next citrus picking path during the ICPR picking process. From the experimental results, it can be concluded that the IAPF algorithm has fast solving speed and accuracy, and can meet the real-time requirements of the ICPR.

5. Conclusions

To improve the efficiency and success rate of the intelligent citrus-picking robot, the motion planning problem of a manipulator—which was the core component of the citrus-picking robot—was studied in-depth, and a novel improved APF algorithm was proposed by improving the models of potential fields and forces of APF and designing some optimization strategies, which greatly reduced the calculation amount of its application in the process of motion planning of a manipulator. In this paper, the ETS method is used to model the kinematics of the EC63M manipulator, and the Jacobian matrix of each point on the manipulator relative to the base coordinate frame can be obtained easily by this model. The improved and original algorithms were both used to carry out the motion of the EC63M manipulator mounted on the intelligent citrus-picking robot. Compared with the original method, the improved one reduced the operation time by 54.89% and the total joint error by 45.41%. Therefore, the improved APF in this paper can not only...
effectively avoid the problems of the local minimum trap but also the target position not reachable problem existing in APF. Concisely, the improved APF has stronger real-time and accuracy performance. Furthermore, because the working environment of a citrus-picking robot is too complex, this paper used the PSE method to model the target citrus tree. The experimental results show that the PSE method can effectively reduce the calculation amount of motion planning for collision detection. Therefore, the improved APF mentioned in this paper is better than the original algorithm, concerning both real-time performances and the accuracy of the citrus picking robot. This paper presents the PSE method of the target citrus tree but does not explain the general situation for any other citrus tree, which will be the next research goal of our project team. In this experiment, we did not consider the impact on the branch directly attached to the fruit when detaching the citrus fruit from it and the optimal picking orientation of the gripper, and these matters will be part of our forthcoming research as well.

**Author Contributions:** Conceptualization, Z.T.; Methodology, Z.T.; Software, Z.T. and H.X.; Validation, Z.T., L.X. and H.X.; Formal analysis, Z.T., Y.W. and H.X.; Investigation, Z.T., Y.W. and Z.K.; Resources, L.X., Y.W. and Z.K.; Data curation, L.X.; Writing—original draft preparation, Z.T.; Writing—review and editing, Z.T., L.X. and Y.W.; Visualization, Z.T.; Supervision, L.X. and Y.W.; Project administration, Z.T. and L.X.; Funding acquisition, X.L and Z.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Key R&D project of Science and Technology Department of Sichuan Province on Research and Development of an Intelligent Citrus Picking Robot, grant number 2020YF00025; Sichuan Provincial Department of Education Research on Key Technology of Manipulator Arm Motion Control of Citrus Picking Robot, grant number 2021JDR0091.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available upon request from the corresponding author. The data are not publicly available due to restrictions of privacy and morality.

**Acknowledgments:** The authors would like to thank the anonymous reviewers and the editor, whose comments and suggestions greatly improved this paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Ross, J.; Davis, V.; Foste, C.; Ray, T. Agricultural Statistics. 2020. Available online: http://www.nass.usda.gov/ (accessed on 24 November 2021).
2. Ning, J.; Xian, Z.; Li, X.; Mao, Y.; Sheng, L.; Zeng, Y.; Xing, Z.; Wen, J.; Liu, A. China Statistical Yearbook. 2020. Available online: https://data.stats.gov.cn/ (accessed on 24 November 2021).
3. Gonzalez-de-Santos, P.; Fernández, R.; Sepúlveda, D.; Navas, E.; Emmi, L.; Armada, M. Field Robots for Intelligent Farms—Inhering Features from Industry. *Agronomy* 2020, 10, 1638. [CrossRef]
4. Mehta, S.S.; MacKunis, W.; Burks, T.F. Robust visual servo control in the presence of fruit motion for robotic citrus harvesting. *Comput. Electron. Agric.* 2014, 102, 146–158. [CrossRef]
5. Mehta, S.S.; Burks, T.F. Vision-based control of robotic manipulator for citrus harvesting. *Comput. Electron. Agric.* 2016, 123, 362–375. [CrossRef]
6. Hayashi, S.; Yamamoto, S.; Tsubota, S.; Ochiai, Y.; Kobayashi, K.; Kamata, J.; Kurita, M.; Inazumi, H.; Peter, R. Automation technologies for strawberry harvesting and packing operations in Japan. *J. Berry Res.* 2014, 4, 19–27. [CrossRef]
7. Sivaraman, B. Design and Development of a Robot Manipulator for Citrus Harvesting. Ph.D. Dissertation, University of Florida, Gainesville, FL, USA, 2006.
8. Ceres, R.; Pons, J.L.; Jiménez, A.R.; Martin, J.M.; Calderón, L. Design and implementation of an aided fruit-harvesting robot (Agribot). *Ind. Robot. Int. J.* 1998, 25, 337–346. [CrossRef]
9. Hayashi, S.; Shigematsu, K.; Yamamoto, S.; Kobayashi, K.; Kohno, Y.; Kamata, J.; Kurita, M. Evaluation of a strawberry-harvesting robot in a field test. *Bio syst. Eng.* 2010, 105, 160–171. [CrossRef]
10. Bu, L.X.; Hu, G.R.; Chen, C.K.; Sugirbay, A.; Chen, J. Experimental and simulation analysis of optimum picking patterns for robotic apple harvesting. *Sci. Hortic.* 2020, 261, 108937. [CrossRef]
11. Barnett, J.; Duke, M.; Au, C.K.; Lim, S.H. Work distribution of multiple Cartesian robot arms for kiwifruit harvesting. *Comput. Electron. Agric.* 2020, 169, 105202. [CrossRef]
12. Wang, X.D.; Luo, X.; Han, B.L.; Chen, Y.H.; Liang, G.H.; Zheng, K.L. Collision-Free Path Planning Method for Robots Based on an Improved Rapidly-Exploring Random Tree Algorithm. *Appl. Sci.* 2020, 10, 1381. [CrossRef]

13. Ali, A.; Lee, J.Y. Integrated Motion Planning for Assembly Task with Part Manipulation Using Re-Grasping. *Appl. Sci.* 2020, 10, 749. [CrossRef]

14. Wei, K.; Ren, B.Y. A Method on Dynamic Path Planning for Robotic Manipulator Autonomous Obstacle Avoidance Based on an Improved RRT Algorithm. *Sensors 2018*, 18, 571. [CrossRef] [PubMed]

15. Kurosu, J.; Yorozu, A.; Takahashi, M. Simultaneous Dual-Arm Motion Planning for Minimizing Operation Time. *Appl. Sci.* 2017, 7, 1210. [CrossRef]

16. Zhao, X.; He, Y.; Chen, X.; Liu, Z. Human–Robot Collaborative Assembly Based on Eye-Hand and a Finite State Machine in a Virtual Environment. *Appl. Sci.* 2021, 11, 5754. [CrossRef]

17. Cheng, Q.; Zhang, W.; Liu, H.S.; Zhang, Y.; Hao, L.N. Research on the Path Planning Algorithm of a Manipulator Based on GMM/GMR-MPRM. *Appl. Sci.* 2021, 11, 7959. [CrossRef]

18. Van Henten, E.J.; Hemming, J.; Van Tuijl, B.A.J.; Kornet, J.G.; Bontsema, J. Collision-free Motion Planning for a Cucumber Picking Robot. *Biosyst. Eng.* 2003, 86, 135–144. [CrossRef]

19. Glasius, R.; Komoda, A.; Gielen, S.C.A.M. Neural Network Dynamics for Path Planning and Obstacle Avoidance. *Neural Netw.* 1995, 8, 125–133. [CrossRef]

20. Zhang, H.; Zhu, Y.F.; Liu, X.F.; Xu, X.R. Analysis of Obstacle Avoidance Strategy for Dual-Arm Robot Based on Speed Field with Improved Artificial Potential Field Algorithm. *Electronics 2021*, 10, 1850. [CrossRef]

21. Luo, L.F.; Wen, H.J.; Lu, Q.H.; Huang, H.J.; Chen, W.L.; Zou, X.J.; Wang, C.L. Collision-Free Path-Planning for Six-DOF Serial Harvesting Robot Based on Energy Optimal and Artificial Potential Field. *Complexity* 2018, 2018, 3563846. [CrossRef]

22. Harik, E.H.C.; Korsaeth, A. Combining Hector SLAM and Artificial Potential Field for Autonomous Navigation Inside a Greenhouse. *Robotics* 2018, 7, 22. [CrossRef]

23. Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. *Auton. Robot. Veh.* 1986, 5, 396–404. [CrossRef]

24. Falco, A.; Hilario, L.; Montes, N.; Mora, M.C.; Nadal, E. A Path Planning Algorithm for a Dynamic Environment Based on Proper Generalized Decomposition. *Mathematics* 2020, 8, 2245. [CrossRef]

25. Sun, J.B.; Zhu, Y.F.; Liu, X.F.; Xu, X.R. Analysis of Obstacle Avoidance Strategy for Dual-Arm Robot Based on Speed Field with Improved Artificial Potential Field Algorithm. *Electronics 2021*, 10, 1850. [CrossRef]

26. Zhou, H.B.; Zhou, S.; Yu, J.; Zhang, Z.D.; Liu, Z.Z. Trajectory Optimization of Pickup Manipulator in Obstacle Environment Based on Improved Artificial Potential Field Method. *Appl. Sci.* 2020, 10, 935. [CrossRef]

27. Matouxi, F.; Boussaid, B.; Abdelkrim, M.N. Distributed path planning of a multi-robot system based on the neighborhood artificial potential field approach. *Simul.-Trans. Soc. Model. Simul. Int.* 2019, 95, 637–657. [CrossRef]

28. Jeon, G.Y.; Jung, J.W. Water Sink Model for Robot Motion Planning. *Sensors 2019*, 19, 1269. [CrossRef] [PubMed]

29. Park, S.O.; Lee, M.C.; Kim, J. Trajectory Planning with Collision Avoidance for Redundant Robots Using Jacobian and Artificial Potential Field-based Real-time Inverse Kinematics. *Int. J. Control. Autom. Syst.* 2020, 18, 2095–2107. [CrossRef]

30. Long, Z. Virtual target point-based obstacle-avoidance method for manipulator systems in a cluttered environment. *Eng. Optim.* 2020, 52, 1957–1973. [CrossRef]

31. Wang, W.R.; Zhu, M.C.; Wang, X.M.; He, S.; He, J.P.; Xu, Z.B. An improved artificial potential field method of trajectory planning and obstacle avoidance for redundant manipulators. *Int. J. Adv. Robot. Syst.* 2018, 15, 1729881418799562. [CrossRef]

32. Corke, P.I. A simple and systematic approach to assigning Denavit–Hartenberg parameters. *IEEE Trans. Robot. Autom.* 2007, 23, 590–594. [CrossRef]

33. Spong, M.W.; Hutchinson, S.; Vidyasagar, M. Robot modeling and control. *Ind. Robot.* 2006, 33, 403.

34. Craig, J.J. *Introduction to Robotics: Mechanics and Control*, 3/E; Pearson Education: Delhi, India, 2009.

35. Haviland, J.; Corke, P. A Systematic Approach to Computing the Manipulator Jacobian and Hessian Using the Elementary Transform Sequence. Available online: https://arxiv.org/abs/2010.08896 (accessed on 24 November 2021).

36. Bac, C.W.; Roorda, T.; Reshef, R.; Berman, S.; Hemming, J.; van Henten, E.J. Analysis of a motion planning problem for sweet-pepper harvesting in a dense obstacle environment. *Biosyst. Eng.* 2016, 146, 85–97. [CrossRef] [PubMed]

37. Van Henten, E.J.; Schenk, E.J.; van Willigenburg, L.G.; Meuleman, J.; Barreiro, P. Collision-free inverse kinematics of the redundant seven-link manipulator used in a cucumber picking robot. *Biosyst. Eng.* 2010, 106, 112–124. [CrossRef]