LOCAL PATH PLANNING METHOD BASED ON SMOOTH TIME ELASTIC BAND ALGORITHM FOR ORCHARD ROBOTIC LAWN MOWER

/

基于 S-TEB 算法的果园割草机人局部路径规划研究

Minhui ZHANG[1)], Pengcheng LV[1)], Jie LIU[1)], Lei LIU[1)], Huibin ZHU[1)], Lili YI *[1)]
[1)] Shandong University of Technology, Collage of Agricultural Engineering and Food Science / China
Tel: +86 18553308656; E-mail: yili0001@sdut.edu.cn
DOI: https://doi.org/10.35633/inmateh-73-21

Keywords: *Local path planning, S-TEB, Orchard Robotic Lawn Mower*

ABSTRACT

This paper proposes a local path planning algorithm method named S-TEB (Smooth Time Elastic Band), aimed at fulfilling the requirement of full coverage for ORLMs (Orchard Robotic Lawn Mowers) during mowing operations. Firstly, by analyzing the tracking control mode of ORLMs in operational scenarios, control points are selected reasonably. Subsequently, utilizing B-spline curves, the path is optimized to generate the optimal trajectory and speed for ORLMs that satisfy multiple objectives and constraints. Finally, multiple simulations and field experiments were conducted in actual operational environments, with a speed of 0.6 m/s. Experimental results show that in scenarios involving obstacle avoidance, the minimum distance between the automatic lawnmower and the outer contour of obstacles is 4 cm. Compared to the traditional TEB planning algorithm, there is a 4.23% increase in mowing coverage area. These findings provide theoretical and technical support for local path planning in the operational scenarios of ORLMs.

摘要

本文提出了一种名为 S-TEB 的局部路径规划方法，以满足果园割草机器人在割草作业中的全覆盖需求。首先，通过分析果园割草机器人在作业场景下的追踪控制模式，合理选取控制点。然后，利用 b 样条曲线对路径进行优化，生成满足多目标、多约束条件的自动割草机最优轨迹和速度。最后，在实际作业环境中进行了多次仿真和实车试验，速度为 0.6 m/s。试验结果显示，在绕行障碍物场景中，果园割草机器人与障碍物外轮廓的最小距离为 4cm。相比传统 TEB 规划算法，割草面积覆盖率提升了 4.23%，为果园割草机器人作业过程中的局部路径规划提供了理论和技术支持。

INTRODUCTION

As a consequence of the long-term decline in the number of people of working age, countries around the world are exploring the use of unmanned agricultural machinery driving technologies *(Zhao et al., 2023; Zhong et al., 2020)*. Grass cutting is an inevitable aspect of agricultural operations. Conducting local path planning and autonomous obstacle avoidance for ORLMs (Orchard Robotic Lawn Mowers) has significant theoretical and practical implications. The use of ORLMs in agricultural production is becoming increasingly prevalent *(Huang et al., 2023)*. Among the essential technologies ensuring the safe operation of ORLMs is real-time obstacle detection and local path planning after encountering obstacles *(Wen et al., 2022)*. This research aims to explore and implement the S-TEB (Smooth Time Elastic Band) algorithm for local path planning in ORLMs, contributing to the advancement of autonomous agricultural machinery technology.

The local path planning algorithms can be divided into graph-based local path planning, sample-based local path planning, curve interpolation fitting-based local path planning, and reinforcement learning-based local path planning *(Weixin et al., 2021)*. *Yang et al., (2024),* introduced a risk assessment-based local path planning algorithm, which extracted and reconstructed active lane-changing scenarios and longitudinal collision prevention scenarios from high-dimensional datasets for validation and evaluation. The proposed algorithm reduced risks and improved driving efficiency, especially during speed and trajectory changes.

In the ORLMs field, *Shi et al.* (*2023)* proposed a local path planning method based on Bezier curves improving the problem of discontinuous curvature in planned paths.

[1] *Minhui Zhang, M.S. Stud. Eng; Pengcheng Lv, M.S. Stud. Eng; Jie Liu, M.S. Stud. Eng; Lei Liu, M.S. Stud. Eng; Huibin Zhu, M.S. Stud. Eng; Lili Yi, Prof. Ph.D.*

Liu et al. (2019), presented an obstacle avoidance path planning algorithm that utilized fifth-degree polynomial functions on the basis of improved shortest path method, addressing the issue of discontinuous curvature in the shortest path method. *Zhang et al., (2019),* based on the basis of original work *(Qiu et al., 2020),* introduced dynamic identification zones and improved the selection range of control points for Bezier curves, this results in a smoother curvature of the plan. *Guo et al., (2022),* proposed a local path planning algorithm based on B-spline curves and an improved rapidly exploring random tree algorithm, enhancing the efficiency and smoothness of the algorithm.

Existing agricultural machinery path planning algorithms are mostly designed for field operations, whereas the requirements for ORLMs operations are distinct, necessitating a certain level of coverage in path planning to maximize weed cutting efficiency *(Yang et al., 2015)*. In the operational scenarios of ORLMs, various types of obstacles exist, including temporarily parked agricultural machinery, taller fruit trees or other traffic participants. The characteristics of ORLMs dictate that they cannot perform complex trajectory movements, thus requiring higher smoothness in trajectory planning *(Wu et al., 2022)*.

In the aspect of environmental perception and localization, environment perception and localization are crucial steps for ORLMs to achieve autonomous movement and task execution. Traditional ORLMs often rely on buried metal wires to demarcate lawn boundaries, which can lead to higher installation costs and inflexibility in adapting to changes in mowing areas *(Chen et al., 2023; Qin et al., 2023)*. This approach is inadequate for dynamic orchard environments. *Yang et al., (2022),* addressed this limitation by equipping ORLMs with an omnidirectional camera and an Inertial Measurement Unit (IMU). They utilized the camera to capture surrounding landmarks, thereby obtaining more precise position and orientation information. The fusion of data from multiple sensors enhances the robot's understanding of the environment and its localization accuracy. In this field, *Huo et al., (2024),* proposed a method that combines data from LiDAR and visual sensors for precise orchard robot localization. Furthermore, *Kang et al., (2020),* introduced a visual perception method based on deep learning, enabling orchard robots to achieve real-time environmental perception and modeling.

In terms of path planning and coverage rate, path planning and coverage rate are central to the execution of mowing tasks by orchard robots. Effective path planning algorithms ensure efficient obstacle avoidance and path optimization, thereby enhancing coverage rates. In related studies, *Wang et al., (2023),* proposed a path planning method based on the propagation wavefront algorithm, achieving favorable results in orchard environments. *Zhang et al., (2022),* investigated the feasibility of using deep learning techniques for path planning, implementing efficient orchard robot path planning through model predictive control. Additionally, *Li et al., (2022),* introduced a path planning strategy combining potential field methods and genetic algorithms, enabling orchard robots to autonomously navigate obstacles and optimize paths. However, it is noted that during obstacle avoidance, the smoothness of the path is poor, making it difficult for the lawnmower to follow its trajectory accurately during actual operations.

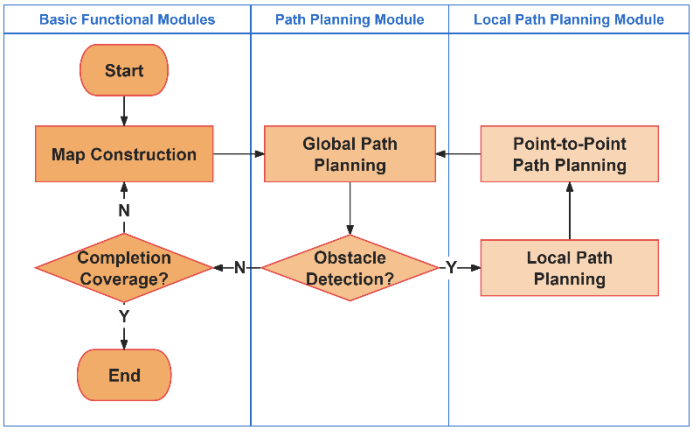

Fig. 1 - The Path planning flow chart for ORLMs

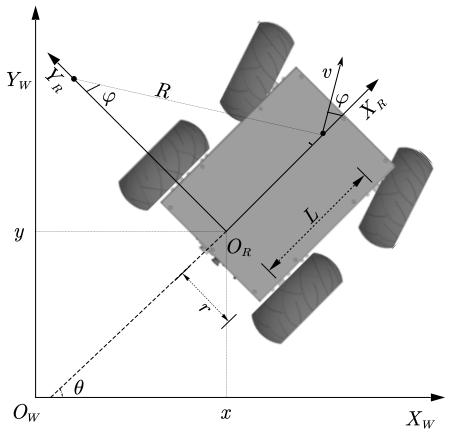

Fig. 2 - The motion model for ORLMs

This paper addresses the characteristics of ORLMs and their operational scenarios, studying the problem of local path planning for ORLMs in various driving environments. Based on the traditional TEB algorithm, the S-TEB algorithm is proposed to adapt to the practical operational requirements of lawnmowers. By selecting control points judiciously and introducing B-spline curves, the proposed algorithm aims to achieve smoother paths while covering a larger area. The objective is to provide technical references for the automation, intelligence, safety, and efficiency of ORLMs navigation.

MATERIALS AND METHODS

In the domain of ORLMs, path planning is a key task because it determines both the action strategy and the safety of ORLMs in complex environments. The proposed S-TEB local path planning algorithm in this paper aims to address the action of ORLMs when encountering obstacles. The primary workflow of the algorithm, as shown in Fig. 1, consists of two crucial steps: first, the TEB algorithm is used to generate a global path based on map information. The TEB algorithm, known for its efficient trajectory generation and optimization capabilities, is widely used in autonomous navigation *(Xia et al., 2022; Zhou et al., 2020)*. Then, when ORLMs encounter obstacles, the S-TEB algorithm is used for replanning. Compared to conventional methods, the S-TEB algorithm shows a significant improvement in obstacle avoidance performance, enabling more adaptive path adjustments to avoid obstacles. By integrating these two steps, our algorithm ensures safe navigation of ORLMs amidst obstacles while striving to achieve maximum area coverage, thereby increasing the efficiency and reliability of path planning.

ORLMs motion mode

In the context of local path planning for ORLMs, the Ackermann motion model plays a central role. This is because ORLMs need to consider their spatial motion characteristics during local path planning to ensure safe and efficient traversal of obstacles and reaching of target points. The Ackermann motion model provides an optimal framework for ORLMs systems to predict and plan their motion trajectories.

To ensure collision-free ORLMs with road edges and obstacles, the ORLMs is abstracted as a rectangular model. The coupled dynamics of the vehicle suspension are not considered. The mass of the ORLMs body is uniformly distributed, and the center of mass is located on the longitudinal geometric line of symmetry of the robot. However, it may not be on the transverse geometric line of symmetry. In addition, the tire sideslip problem has been neglected.

The motion model for ORLMs is shown in Fig. 2. In order to accurately simulate the Ackermann steering mode of the ORLMs, two basic coordinate systems have been introduced: the Cartesian coordinate system W (X_w-O_w-Y_w) , and the frenet coordinate system R (X_R-O_R-Y_R). The coordinates of the ORLMs are $S = (x, y, \theta)$, where (x, y) is the position of the smart car in the world coordinate system, θ is the angle between the longitudinal axis of the ORLMs and the world coordinate system. The steering angle of the front wheels of the ORLMs is denoted as φ, with a maximum steering angle constraint given by $\varphi \leqslant \varphi_{max}$. The wheelbase between the front and rear wheels is denoted by L, and the angular velocity of the rear wheels is denoted by ω. The turning radius of the ORLM is denoted by r.

The relationship between the turning radius r, the wheelbase L, and the steering angle φ of the front wheels should satisfy the following equation *(Li et al., 2023)*:

$$r = \frac{L}{\tan[\varphi(t)]} \tag{1}$$

The relationship between the steering angle of the front wheels and the angular velocity is given by Equation (2).

$$\omega(t) = \frac{v(t)}{r(t)}, \varphi(t) = \arctan\left[\frac{\omega(t)L}{v(t)}\right] \tag{2}$$

In the equation, $\varphi(t)$ represents the real-time steering angle of the ORLMS. The kinematic state equation of the ORLMs is shown in Equation (3).

$$\dot{S}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} \cos[\theta(t)] \\ \sin[\theta(t)] \\ \tan\left[\frac{\varphi(t)}{L}\right] \end{bmatrix} v(t) \tag{3}$$

In the equation, $\dot{S}(t)$ represents the derivative of the pose state of the ORLMs, $[\dot{x}(t), \dot{y}(t), \dot{\theta}(t)]^{\mathrm{T}}$ represents the position state of the ORLMs.

Vehicle control method

Stanley is a trajectory tracking algorithm based on geometry. Its main idea is to use the center of the front wheel as the reference point, without the need for a preview distance, to directly calculate the lateral error and heading error from the center of the front wheel to the target trajectory.

Assume that the current lateral error e=0, so the vehicle only needs to turn through the angle θ_e, and the vehicle can always drive along the target trajectory. In this case, let $\delta(t) = \theta_e(t)$. In order to eliminate the lateral error, construct a triangle, it can be obtained:

$$tan\delta_e = \frac{e}{d^{\omega}} \tag{4}$$

$d = \frac{v}{k}$ into the above formula, the following is obtained:

$$\delta_e = tan^{-1}\left(\frac{ke}{v}\right), \delta(t) \in [\delta_{min}, \delta_{max}] \tag{5}$$

$$\delta(t) = \theta_e(t) + tan^{-1}\left(\frac{ke}{v}\right), \delta(t) \in [\delta_{min}, \delta_{max}] \tag{6}$$

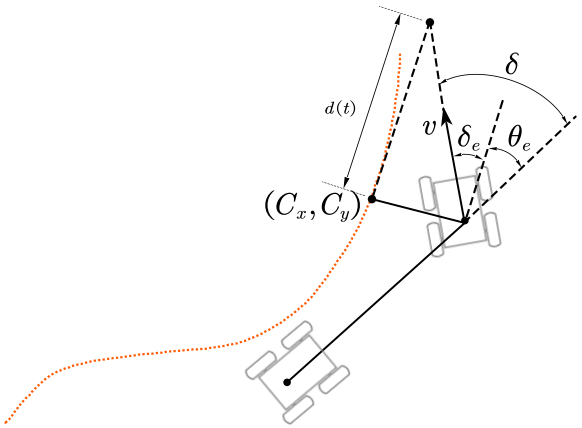

Fig. 3 - Schematic diagram of Stanley trajectory tracking control algorithm

ORLMs generally operates at low speed. In order to solve the noise problem, a positive constant k_s is added. The final corner is:

$$\delta(t) = \theta_e(t) + tan^{-1}\left(\frac{ke}{k_s + v}\right) \tag{7}$$

Principle of TEB algorithm

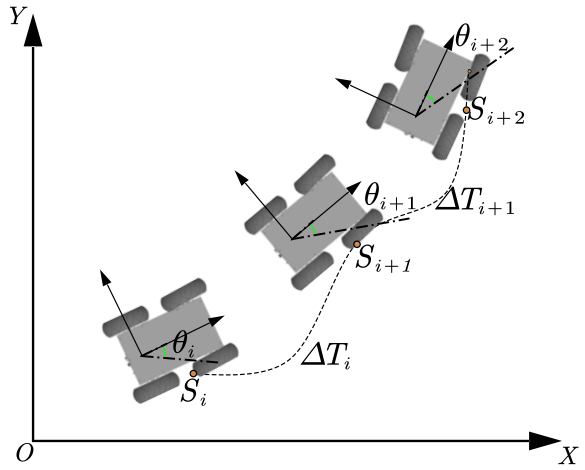

Fig. 4 - The relationship between bit position and time interval with TEB algorithm

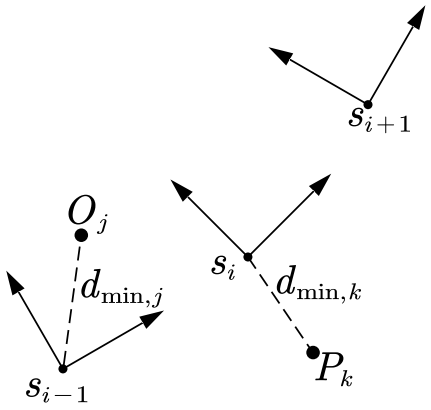

Fig. 5 - Abstract graph of pose and time interval relationship in TEB algorithm

The global path of the TEB algorithm consists of poses relative to discretized pose points $S_i(x_i, y_i, \theta_i)^{\mathrm{T}}$. In this context, (x_i, y_i) represents the position of discrete path points, and θ_i denotes the angle between the direction of motion of the ORLMs and the X-axis of the world map, as shown in Fig. 4. At this point, the pose sequence can be represented as:

$$Q = \{S_i\}_{i=0,\cdots,n} \tag{8}$$

The time interval $\boldsymbol{\Delta T}_i$ denotes the time required for the ORLMs to move from the current position point S_i to the next position point S_{i+1} in the sequence Q. The time interval $\boldsymbol{\Delta T}_i$ is the time required for the ORLMs to move from the current position point S_i to the next position point S_{i+1} in the sequence *Q (Dai et al., 2022)*. Therefore, the trajectory formed by the TEB algorithm consists of two sets of time interval sequences and

position point sequences. Therefore, the trajectory formed by the TEB algorithm consists of two sets of time interval sequences and position point sequences. The time interval sequence can be expressed as:

$$\tau = \{\Delta T_i\}_{i=0,1,\cdots,n-1} \tag{9}$$

The optimization sequence consisting of the position and time of the intelligent cart in the TEB algorithm can be expressed as follows:

$$\begin{aligned} \mathrm{B} &= (\mathrm{Q}, \tau) \\ &= \{\mathrm{s}_1, \Delta\mathrm{T}_1, \mathrm{s}_2, \Delta\mathrm{T}_2, \ldots, \mathrm{s}_{\mathrm{n}-1}, \Delta\mathrm{T}_{\mathrm{n}-1}, \mathrm{s}_{\mathrm{n}}\} \end{aligned} \tag{10}$$

The central concept behind the TEB algorithm is to determine the optimal sequence of vehicle poses within specified time intervals. This optimization problem is formulated as a nonlinear least-squares cost function, aiming to aggregate the weighted sum of B* values.

In the objective function B constituted by the optimization sequence, the relationship between each position point and time is affected by the TEB constraints. Usually, the TEB constraints contain four types of constraints: incomplete kinematics constraints, acceleration and velocity constraints, obstacle point and global path point constraints, and shortest time constraints *(Wu et al., 2021)*.

In most cases, ORLMs needs to follow the global path points to navigate around obstacles and complete local path planning *(Jiang et al., 2022)*. The global path planning algorithm treats ORLMs as a point mass, and the planned global path is an ideal trajectory. However, in real-world scenarios, due to constraints such as its own dynamics, ORLMs cannot completely follow the global path *(Huang et al., 2023)*. Therefore, it is necessary to further consider the information of global path points and obstacles, establish reasonable constraints, and plan paths that conform to reality.

The distance between the local path point and the global path point is expressed as $d_{min,k}$, and the distance between the local path point and the obstacle is expressed as $d_{min,j}$, its simplified diagram is shown in Fig. 5, point O_j represents an obstacle, and point P_k represents the path point of the global path. The obstacle avoidance penalty function of TEB is as follows:

$$f_{\text{obstacle}} = e_\tau\left(-d_{\min,j}, -r_{O_{\min}}, k, S, n\right) \tag{11}$$

In order to ensure that the ORLMs do not collide with road edges and obstacles, the ORLMs are abstracted as a rectangular model and a kinematic model. In the formula, $r_{O_{min}}$ represents the lower limit of the minimum distance set between the smart car and the obstacle. Similarly, the number of penalty terms following the global path can be expressed as:

$$f_{\text{path}} = e_\tau\left(d_{min,j}, r_{P_{max}}, k, S, n\right) \tag{12}$$

$r_{P_{max}}$ is the upper limit of the allowed deviation from the global path points. The penalty function takes effect when the distance of the smart trolley from the obstacle is less than the set lower limit distance, or the distance from the path point is greater than the set upper limit distance. If it is within the allowed range, the penalty function outputs 0.

Optimize TEB algorithm

Incomplete Kinematic Constraints are constraints that contain generalized coordinate derivatives of the system and are not integrable. The turning radius of the ORLMs is limited by its own turning radius, which belongs to incomplete kinematic constraints. Fig. 6 shows the relationship between bit position and time interval with TEB algorithm when turning.

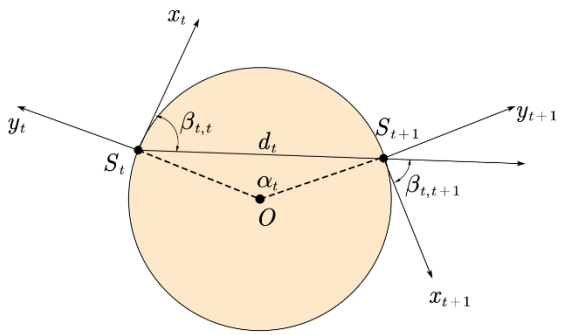

Fig. 6 - Schematic diagram of ORLMS turning motion

The displacement of the non-complete kinematic constraint of the AMR between two neigh boring attitude points is a path composed of a segment of arcs. In Fig. 6, the displacements of the ORLMs from point S_{i} to point $S_{\mathrm{i}+1}$ are on arcs with constant curvature, thus they satisfy the following equations.

$$\beta_{t,t} = \beta_{t,t+1} \tag{13}$$

$$d_{t,t+1} = \begin{pmatrix} x_{t+1} - xt \\ y_{t+1} - y_t \\ 0 \end{pmatrix} \tag{14}$$

$$\begin{bmatrix} \cos\beta_{t,t} \\ \sin\beta_{t,t} \\ 0 \end{bmatrix} d_{t,t+1} = d_{t,t+1} \begin{bmatrix} \cos\beta_{t,t+1} \\ \sin\beta_{t,t+1} \\ 0 \end{bmatrix} \tag{15}$$

Where: $\beta_{t,t}$ and $\beta_{t,t+1}$ are the angles between the traveling direction of the ORLMS and the X-axis of the world coordinates, so the cost equation of the incomplete kinematic constraints is shown as follows.

$$f(x_t, x_{t+1}) = \left\| \left[\begin{pmatrix} \cos\beta_{t,t} \\ \sin\beta_{t,t} \\ 0 \end{pmatrix} + \begin{pmatrix} \cos\beta_{t,t+1} \\ \sin\beta_{t,t+1} \\ 0 \end{pmatrix} \right] d_{t,t+1} \right\|^2 \tag{16}$$

When the ORLMs with Ackermann structure turns, the turning radius must not be smaller than its minimum turning radius, therefore, its constraint is shown as follows.

$$r(x_{t+1}, x_t) = \left| \frac{v_t}{\omega_t} \right| \geqslant \left| \frac{d_t}{2\sin\frac{\alpha_t}{2}} \right| = r_{\min} \tag{17}$$

The smart trolley needs to satisfy the incomplete kinematics constraints, and also needs to satisfy that the turning radius of the smart trolley is not less than its minimum turning radius, $r_t \geq r_{\min}$.

- **Velocity and acceleration constraints**

The constraints of velocity and acceleration are composed of the Euclidean distance between two neigh boring attitude points S_{i} and $S_{\mathrm{t+1}}$ and the time ΔT_i needed for the motion between the two points, and the constraints are formed by obtaining the rotational and translational velocities of the intelligent trolley, whose solution formulas are shown as follows.

$$v \approx \frac{1}{\Delta T_i} \left\| \begin{pmatrix} x_{t+1} - x_t \\ y_{t+1} - y_t \end{pmatrix} \right\| \tag{18}$$

$$\omega_t \approx \frac{\beta_{t+1} - \beta_t}{\Delta T_i} \tag{19}$$

Similarly, the average linear acceleration and average angular acceleration can be obtained from the linear and angular velocities of two consecutive postures, shown as follows.

$$a_t = \frac{2(v_{t+1} - v_t)}{\Delta T_i + \Delta T_{i+1}} \tag{20}$$

$$a_\omega = \frac{2(\omega_{t+1} - \omega_t)}{\Delta T_i + \Delta T_{i+1}} \tag{21}$$

- **Minimum time constraint**

The TEB algorithm combines the time information between the neighboring position points of the ORLMs, different from the traditional local path planning algorithm which takes the shortest distance as the optimization criterion, the TEB algorithm takes the shortest time as the optimization criterion.

The objective function takes the square of the sum of the time intervals between all the position points as the optimal index, and its expression is shown as follows.

$$f_{\text{short\_time}} = \left(\sum_{i=0}^{n} \Delta T_i \right)^2, i \in N \tag{22}$$

Constraints on Obstacle Points and Global Paths

Usually, the ORLMs needs to follow the global path points to avoid obstacles and complete the local path planning. The global path planning algorithm treats the ORLMs as a mass point, and the planned global path is an ideal path. However, in reality, the ORLMs cannot follow the global path completely due to its own dynamics and other constraints. Therefore, it is necessary to further consider the global path points and obstacle information to construct reasonable constraints and plan a realistic path.

- **The trajectory planning constraints**

For a curve composed of discrete points with equal lateral spacing, the magnitude of the first derivative of the trajectory points is positively correlated with the length of the curve. That is, the smaller the first derivative of the trajectory points, the smoother the curve. Hence, the cost function C_{smooth} for smoothness can be derived as:

$$C_{\text{smooth}} = W_{\text{lenth}} \sum_{i=1}^{n} (l'(s_i))^2) \tag{23}$$

where W_{lenth} is the distance weight, and $l'(s_i)$ is the lateral displacement of the agricultural machinery at the longitudinal distance s_i.

- **Severity of danger**

The severity of danger is assessed based on the distance of ORLMs from obstacles when circumventing static obstacles. Let d denote the straight-line distance between the ORLMs and the obstacle.

The cost function for the severity of danger is denoted as C_{danger}.

$$C_{\text{danger}} = \begin{cases} 0, d > d_2 \\ \dfrac{W_{\text{danger}}}{d}, d_1 \leqslant d < d_2 \\ +\infty, d \leqslant d_1 \end{cases} \tag{24}$$

In the formula, W_{danger} represents the weight of the severity of danger, d_1 is the risk distance in meters (m), and d_2 is the safety distance in meters (m). Beyond the safety distance, it is assumed that there is no collision risk between the agricultural robot and the obstacle, and the cost is 0. Within the risk distance, it is assumed that there is a collision risk between the agricultural robot and the obstacle, and the cost is considered infinite. Between the risk distance and the safety distance, the severity of danger is inversely proportional to d.

- **Selecting Control Points**

In the actual operation process, the task target point of the lawn mower is abstracted as shown in the Fig. 7. As is commonly understood, curvature denotes the degree of curvature exhibited by a curve. Altering the radius of a circle, smaller radii yield larger curvatures, resulting in more pronounced curvature of the curve. Conversely, larger radii lead to smaller curvatures, resulting in a smoother curve. Illustrated in the diagram are three curves denoted as l_1, l_2, and l_3, respectively. These curves are positioned on circles centered at points O_1, O_2, and O_3, with radii denoted as R_1, R_2, and R_3, respectively. If curvature is denoted by K:

$$K = \frac{1}{r} \tag{25}$$

For a general curve, the degree of curvature varies at different positions. This variation in curvature can be described using curvature. Curvature denotes the degree of curvature of a curve at a certain point and can be obtained by computing the second derivative of the curve.

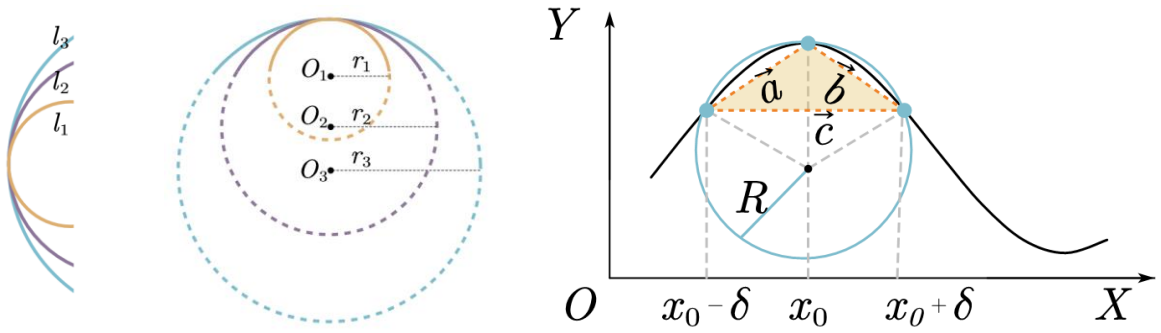

Fig. 7 - Curvature models

In the curve, some positions exhibit significant curvature while others are relatively flat. To compute the curvature of point x_0 on the curve, a point to the left is selected and another to the right of x_0, $x_0 - \delta$ and $x_0 + \delta$ respectively. These three points collectively determine a circle, the radius of which is named R, as δ tends towards zero, this circle becomes the osculating circle of the curve at point x_0. By calculating the radius of the osculating circle, the curvature of the curve at point x_0 can be determined.

According to the circumscribed circle formula:

$$R = \frac{abc}{4S} \tag{26}$$

S is the area of the circumcircle determined by three points a, b, and c representing the lengths of the three sides. The directed area of the parallelogram formed by vectors a and b can be represented by the determinant $det(\vec{a}, \vec{b})$.

$$R = \frac{abc}{4S} = \frac{||\vec{a}|| \times ||\vec{b}|| \times ||\vec{c}||}{4 \times \det(\vec{a}, \vec{b})} \tag{27}$$

Dividing both sides by δ^3 it is obtained:

$$R = \frac{\sqrt{[f(x_0+\delta) - f(x_0-\delta)]^2 + 4\delta^2} \cdot \sqrt{[f(x_0+\delta) - f(x_0)]^2 + \delta^2} \cdot \sqrt{[f(x_0) - f(x_0-\delta)]^2 + \delta^2}}{2|\delta f(x_0+\delta) + \delta f(x_0-\delta) - 2\delta f(x_0)|} \tag{28}$$

When δ approaches 0, the radius of the osculating circle is r, thus r is the limit of R, resulting in:

$$r = \lim_{\delta \to 0} R = \frac{\left(1 + \left(f'(x_0)\right)^2\right)^{\frac{3}{2}}}{|f''(x_0)|} \tag{29}$$

As the result:

$$K = \frac{1}{r} = \frac{|f''(x_0)|}{\left(1 + \left(f'(x_0)\right)^2\right)^{\frac{3}{2}}} = \frac{2|(x_1 - x_2)(y_2 - y_3) - (x_2 - x_3)(y_1 - y_2)|}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}^3} \tag{30}$$

Through detailed calculations and analysis, a significant conclusion is reached: when the parameter k undergoes substantial changes, employing a strategy of multiple control points can effectively ensure the safety of autonomous robotic systems (ORLMs) during operation. When k changes from 0 to non-zero, it evaluates to true. When the change in k exceeds 0.26, the evaluation result is true. This can be expressed with the following formula:

$$\begin{cases} \text{true}, & \text{if}(k_{\text{previous}} = 0) \text{ or if}(k = 0) \\ \text{true}, & if(|k_{\text{previous}} - k|) > 0.26 \\ \text{false}, & \text{otherwise} \end{cases} \tag{31}$$

Where k_{previous} represents the previous value of k.

Implementing this strategy enables the ORLMs system to adapt more flexibly to various environments and situations. Moreover, when faced with complex road conditions, it can navigate and plan paths more robustly. Therefore, selecting multiple control points is an effective approach to enhance the safety and stability of ORLMs systems, which holds crucial significance for the advancement of autonomous robotics technology.

- **B-spline curve smoothing strategy**

For the initial path obtained from the TEB algorithm, a cubic uniform B-spline curve smoothing strategy is introduced to optimize the path smoothing. The formula for a B-spline curve can be expressed as:

$$P(u) = \sum_{i=0}^{n} N_{i,p}(u) \cdot P_i \tag{32}$$

Where $P(u)$ is a point on the curve, u is the parameter, n is the number of control points minus 1, P is the degree of the B-spline minus 1, $N_{i,p}(u)$ is the B-spline basis function, and P_i is a control point.

When k=3, the mathematical expression for the cubic uniform B-spline curve is:

$$P(u) = \sum_{i=0}^{n} N_{i,3}(u) \cdot P_i \tag{33}$$

After introducing the smoothing strategy, line segments around the turning points in the path are replaced by curves, resulting in smoother local paths being generated.

- **Smoothness evaluation function**

This paper designs a smoothness function $\mathbf{Q}_\mathrm{k}$ to calculate the overall smoothness of the path during algorithm. This function is represented as:

$$\mathbf{Q}_\mathrm{k} = \sum_{i=2}^{k-1} abs\left(\arccos\frac{D_1^2 + D_2^2 - D_3^2}{2D_1D_2}\right) \cdot \frac{180}{\pi} \tag{34}$$

In the equation, D_1, D_2, and D_3 represent the distances between any three adjacent path nodes. The expression is given by:

$$\begin{cases} D_1 = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \\ D_2 = \sqrt{(x_{i+2} - x_{i+1})^2 + (y_{i+2} - y_{i+1})^2} \\ D_3 = \sqrt{(x_{i+2} - x_i)^2 + (y_{i+2} - y_i)^2} \end{cases} \quad (35)$$

In equation, a larger value of $\mathbf{Q}_k$ indicates that there are more acute angles between adjacent path nodes, meaning the path is more tortuous; conversely, a smaller value represents a smoother path.

Overall, in optimizing the navigation process of ORLMs, the main improvement directions include selecting control points reasonably, adding danger constraints, and increasing velocity constraints. By selecting control points reasonably, the smoothness and coverage area of path planning can be improved. Additionally, the danger penalty function makes ORLMs more cautious when encountering obstacles or complex road conditions, thereby enhancing its navigational safety. Furthermore, increasing velocity constraints helps balance the speed and safety of ORLMs, ensuring stability during the navigation process. These improvement measures collectively influence the path planning and navigation process of ORLMs, aiming to enhance its adaptability, safety, and stability in various environments and conditions.

RESULTS

- **Simulation**

In order to verify the effectiveness of the improved S-TEB local planning algorithm, the algorithm was integrated into the ROS robot operating system for simulation experiments, and then actual experimental tests were conducted on the Ackerman differential robot[6]. The purpose is to verify the improved S-TEB algorithm. There are 3 aspects of performance: first, the effectiveness of planning safe motion trajectories at obstacles; second, the effectiveness of smoothing the planned path; third, the effectiveness of accurately reaching the target point.

In the simulation environment shown in Fig. 8, the left side is the location of the simulation environment and its obstacles, and the right side is the environment map constructed using SLAM. In the map, the innermost circle is the obstacle, the first circle is the expansion distance of the obstacle, and the second circle is the safe distance.

To validate the effectiveness of the S-TEB local path planning algorithm, simulation experiments were conducted in the ROS system. The operating system utilized is Ubuntu 18.04, through the gazebo and RVIZ platform in ROS-melodic system co-simulation, in the gazebo platform to build the simulation environment and set up the physical simulation map, in the RVIZ platform to display the obstacles and the intelligent car model, the ORLMs trajectory through the ROS control.

First, establish a map of the plowed path. The global path consists of smooth curves formed by connecting trajectory points.

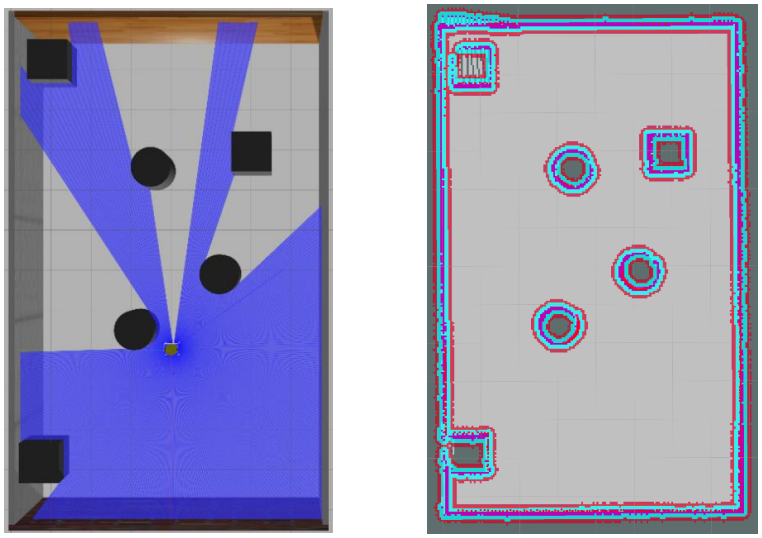

Fig. 8 - Simulation environment and map constructed by slam

Subsequently, employing the aforementioned formula, the process of selecting control points ensues. These meticulously chosen control points are then integrated into the B-spline curve for optimization, resulting in the refinement of the local path. This iterative procedure seamlessly harmonizes mathematical precision with real-world application, culminating in a locally optimized trajectory tailored to the intricacies of the environment.

Through the fusion of algorithmic finesse and practical utility, the refined path intricately weaves through obstacles, ensuring optimal traversal efficiency and enhancing the navigational capabilities of the autonomous system. This meticulous optimization not only fosters smoother trajectory planning but also bolsters the system's adaptability and resilience amidst dynamic operational scenarios.

The comparison between paths circumventing square and circular obstacles, as depicted in Fig. 9 and Fig. 10, reveals a notable distinction. The green line illustrates the path planned using the conventional TEB algorithm, while the red line signifies the path charted using the enhanced S-TEB algorithm. It is discernible from these illustrations that the path generated by the improved S-TEB algorithm exhibits enhanced smoothness and encompasses a larger coverage area compared to its traditional counterpart.

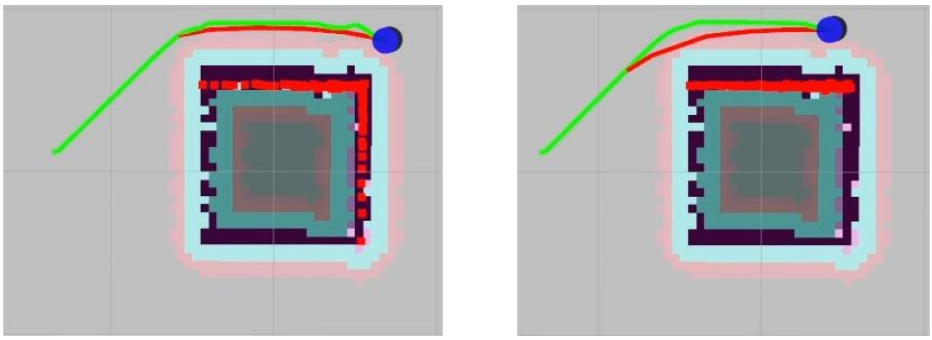

Fig. 9 Comparison chart of path planning under square obstacle scenario

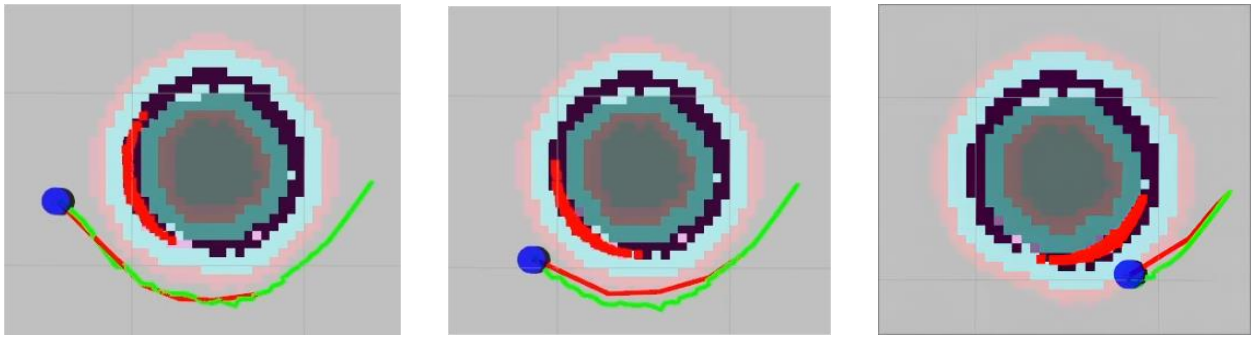

Fig. 10 Comparison chart of path planning under circular obstacle scenario

Experiment

Fig. 11 – Local Path Planning Experiment to ORLMs

Standardized grape vineyards ensure a high degree of consistency in row spacing, plant spacing and pruning methods, which provides an ideal basis for orchard robots to work in and plan paths. Furthermore, the data collection and analysis of the standardized grape vineyard is more convenient. Consequently, the standardized grape vineyard is selected as the experimental environment for the orchard robot.

Through continuous testing and optimizing in this environment, the mature technology solution can be gradually extended to other types of orchards or agricultural fields, thereby promoting the intelligent upgrading of the entire agricultural production process. The experimental test scenario mirrors the real-world operational conditions encountered by ORLMs and is set within a standardized grape vineyard environment. This scenario comprises two cylindrical obstacles and one cuboid obstacle strategically placed to simulate dynamic obstacles commonly encountered in agricultural settings. By replicating these realistic conditions, the performance of the robots in effectively navigating around obstacles can be thoroughly evaluated. This carefully designed experimental setup ensures that the outcomes accurately reflect the capabilities and effectiveness of the robotic systems in real-world agricultural applications.

In this series of scientific research experiments, a variety of sensors and intelligent control modules were integrated into the ORLMs. The RTK positioning system provides centimeter-level ultra-high-precision positioning, while the 64-line 3D LIDAR offers detailed obstacle information through its omni-directional field of view and high-precision detection capability. It is noteworthy that the ROS operating system was implemented on the Nvidia Jetson Orin NX hardware platform, which was paired with the VCU vehicle control unit to serve as the ORLMs's intelligent brain.

Fig. 12 – Local Path Planning Experiment to ORLMs in the spring

The SLAM was adopted to construct a point cloud map and employ RTK (Real-Time Kinematic) devices for online localization. As shown in Fig. 13, a 3D point cloud map of the current scene is opened in Reconstruct Visualization Interface. This map is employed to delineate the effective identification area, which is instrumental in the process of identifying potential obstacles. Subsequently, the 2D Pose Estimate tool is employed. In order to approximate the location and orientation of the current vehicle on the map, it is necessary to remotely move the vehicle by a distance of between one and two meters. Concurrently, it is essential to monitor the alterations in the vehicle's coordinate system within the designated visualization area to ascertain the precision of the initialized vehicle position. The 2D Nav Goal tool is employed to select a target point on the map, and the TEB (Timed Elastic Band) global path planner will generate a global path from the starting point to the target point.

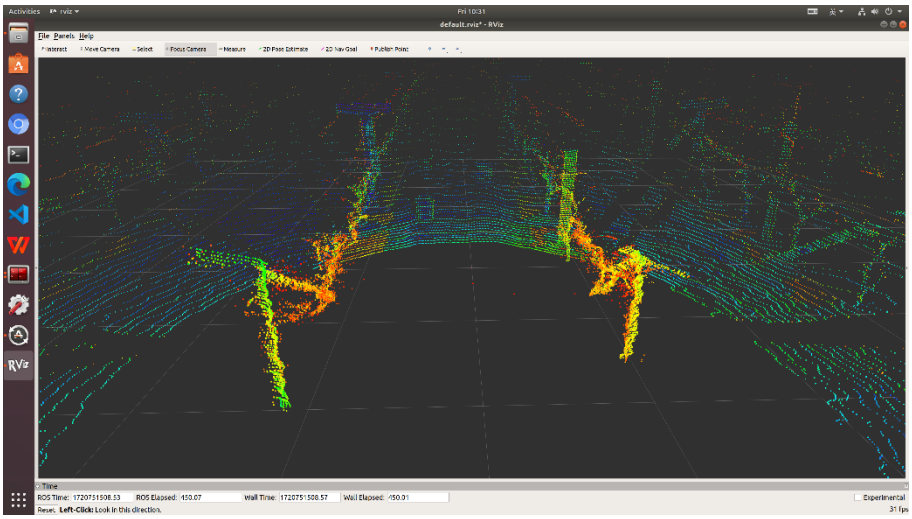

Fig. 13 – The Point Cloud Plots from the experiment

The ORLMs operated at a speed of approximately 0.6 m/s, traversing from a fixed starting point to a fixed endpoint. Within the program, both the traditional TEB and S-TEB algorithms were separately employed to replace the local path planning module, each followed by ten experimental obstacle avoidance tests. The obstacle conditions were based on the model depicted in Fig. 12, with a standardized grape vineyard environment featuring row spacing of 1.5 meters and a minimum distance of 4 cm from obstacles. Experimental parameter data were collected using ROS tools such as rqt and rosbag, and analyzed based on the evaluation metrics that were developed. The results, depicted in Fig. 14. show that the coverage area increased by 4.23%. To facilitate observation of data changes, the numerical values of the x-axis and y-axis coordinates underwent non-uniform scaling.

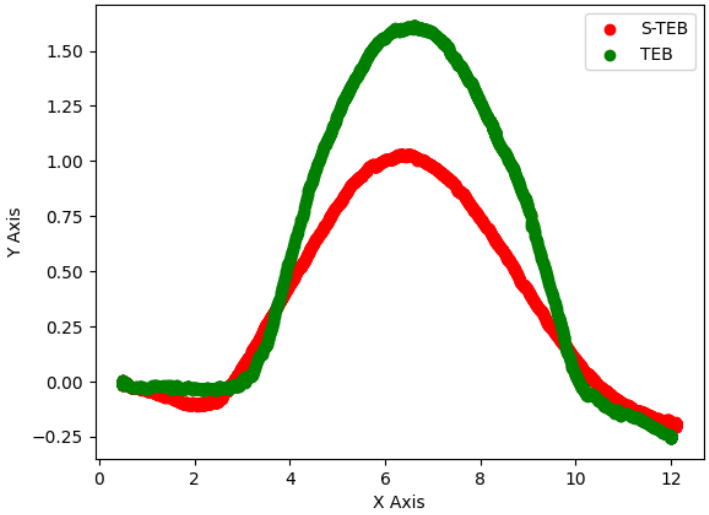

Fig. 14 Comparison diagram of experimental paths

From the Fig. 14, it can be observed that neither of the two path planning methods resulted in collisions with obstacles. Compared to the traditional TEB algorithm, the motion planner of the proposed TEB-CA algorithm maintains a smaller distance relative to obstacles and covers a larger area of weed growth. Additionally, there is minimal fluctuation in speed and angle, ensuring stable control and reducing the probability of rollover during turning and obstacle avoidance. The planned travel path meets the dynamic constraint requirements of the mobile robot, with various state quantities changing accordingly.

1. Obstacle Avoidance from Start to Goal:

The experimental results demonstrate the efficacy of the S-TEB algorithm in enabling the ORLMs to navigate from the starting point to the target point while effectively avoiding obstacles in the environment. Through real-world testing scenarios, the ORLMs successfully circumvented various obstacles encountered along the path, showcasing its robust obstacle avoidance capability.

2. Enhanced Smoothness in Local Path Planning:

Incorporating the S-TEB algorithm into the ORLM's local path planning process yielded notable improvements in trajectory smoothness. By adjusting parameters such as velocity and acceleration profiles, the ORLMs was able to generate smoother paths, minimizing abrupt changes in direction and velocity. This enhanced smoothness not only contributes to the overall navigation efficiency but also ensures a more stable and predictable trajectory for the ORLMs.

3. Mowing Coverage Area Increases:

Analysis of the experimental data reveals that the implementation of the improved S-TEB algorithm has resulted in a notable enhancement in mowing coverage area, with a significant increase of 4.23%.

CONCLUSIONS

This paper optimizes the ORLMs local path planning algorithm by incorporating the B-spline curve time elastic band local path planning algorithm, and proposes the S-TEB algorithm for improved local path planning based on this optimization.

In this paper, the SLAM method was adopted to construct a point cloud map and utilized RTK (Real-Time Kinematic) devices for online localization. In a standardized vineyard environment, two cylindrical obstacles and one cuboid obstacle were set up to evaluate the performance of robots in avoiding dynamic obstacles.

The experimental results demonstrate that the paths planned using the improved S-TEB algorithm exhibit smoother trajectories and cover a larger area compared to those generated by the traditional TEB algorithm, showcasing enhanced navigation capabilities. This research provides valuable insights and guidance for autonomous navigation of agricultural robots in complex environments.

While this study successfully explores the effectiveness of the improved S-TEB algorithm in the local path planning of ORLMs and demonstrates a significant improvement with a 4.23% increase in coverage area, there are still areas that require further investigation. Specifically, this research did not analyze dynamic obstacles, which could be a crucial factor in actual orchard environments. Therefore, future research could consider expanding the scope of experiments to include in-depth analysis and testing of dynamic obstacles to comprehensively assess the performance of ORLMs in real orchard operations. Additionally, further optimization of algorithm parameters and path planning strategies could enhance the navigation efficiency and stability of ORLMs. Overall, while this study provides valuable insights for the development of orchard robot technology, there are still many directions to explore and improve upon to achieve higher levels of autonomy and operational efficiency.

ACKNOWLEDGEMENT

The work was supported by the R&D of low-speed unmanned autonomous navigation controller Project under Grant (No.2022TSGC1175).

REFERENCES

[1] Chen Ruiyun, Tian Wenbin, Bao Haibo, Li Duan, Xie Xinhao, Zheng Yongjun, Tan Yu., (2023). Three-dimensional environment perception technology for agricultural wheeled robots: A review（农业轮式机器人三维环境感知技术研究进展）. *Smart Agriculture*, 5(4), 16-32. https://doi.org/10.12133/j.smartag.SA202308006

[2] Dai Wanyu, Zhang Lijuan, Wu Jiafeng, Ma Xianghua, (2022). Research on Local Path Planning Algorithm Based on Improved TEB Algorithm (改进 TEB 算法的局部路径规划算法研究). *Computer Engineering and Applications*, 58(8), 283-288. https://doi.org/10.3778/j.issn.1002-8331.2108-0290

[3] Guo Chengyang, Liu Meichen, Gao Zeng, Zhou Jianguo, Chen Jun, (2020). Research on agricultural machinery obstacle avoidance method based on improved artificial potential field method. *Journal of Chinese Agricultural Mechanization*, 41(3), 152-157. https://doi.org/10.13733/j.jcam.issn.2095-5553

[4] Jiang H., Pi J., Li A., Yin C. (2022). Dynamic Local Path Planning for Intelligent Vehicles Based on Sampling Area Point Discrete and Quadratic Programming. *IEEE Access*, vol. 10, pp.70279-70294, 2022, https://doi.org/10.1109/ACCESS.2022.3183154

[5] Hang P., Yan, Y., Fu, X., Chen, H., Liu, Y., (2023, April). Research on local path planning of intelligent vehicle based on improved dynamic window approach. In *Third International Conference on Artificial Intelligence and Computer Engineering (ICAICE 2022)* (Vol. 12610, pp. 1326-1334). SPIE. https://doi.org/ 10.1117/12.2671069

[6] Huang X., Yi J., Shen T., (2023). Research on automatic lawn mower system based on differential GPS and IMU fusion. In *2023 3rd Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)*, 26-432, IEEE. https://doi.org/10.1109/ACCTCS58815.2023.00056

[7] Huo, F., Zhu, S., Dong, H., Ren, W., (2024). A new approach to smooth path planning of Ackerman mobile robot based on improved ACO algorithm and B-spline curve. *Robotics and Autonomous Systems*, 175, 104655. https://doi.org/10.1016/j.robot.2024.104655

[8] Kang H., Zhou H., Chen C., (2020). Visual perception and modeling for autonomous apple harvesting. *IEEE Access*, *8*, 62151-62163. https://doi.org/10.1109/ACCESS.2020.2984556

[9] Li Y., Li J., Zhou W., Yao Q., Nie J., Qi X., (2022). Robot Path Planning Navigation for dense planting red jujube orchards based on the joint improved A* and DWA algorithms under laser SLAM. *Agriculture*, 12(9), 1445. https://doi.org/10.3390/agriculture12091445

[10] Li J., Zhang Z., (2023, August). AUV Local Path Planning Based on Fusion of Improved DWA and RRT Algorithms. In *2023 IEEE International Conference on Mechatronics and Automation (ICMA)* (pp. 935-941). IEEE. https://doi.org/10.1109/ICMA57826.2023.10216263

[11] Liu Z., Lü Z., Zheng W., Zhang W., Cheng X., (2019). Design of obstacle avoidance controller for agricultural tractor based on ROS. *International Journal of Agricultural and Biological Engineering*, *12*(6), 58-65. https://doi.org/10.25165/j.ijabe.20191206.4907

[12] Qin J., Sun R., Zhou K., Xu Y., Lin B., Yang, L., ... & Wu, C., (2023). Lidar-Based 3D Obstacle Detection Using Focal Voxel R-CNN for Farmland Environment. *Agronomy*, *13*(3), 650. https://doi.org/10.3390/agronomy13030650

[13] Qiu Z., Zhao N., Zhou L., Wang M., Yang L., Fang H., ... & Liu Y., (2020). Vision-based moving obstacle detection and tracking in paddy field using improved yolov3 and deep SORT. *Sensors*, *20*(15), 4082. https://doi.org/10.3390/s20154082

[14] Shi Yangjie, Cheng Xinhui, Xi Xiaobo, (2023). Research progress on the path tracking control methods for agricultural machinery navigation (农业机械导航路径跟踪控制方法研究进展). *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE),* 39(15), 1-14. https://doi.org/10.11975/j.issn.10026819.202304004

[15] Wang, Z., Li, P., Li, Q., Wang, Z., Li, Z., (2023). Motion Planning Method for Car-Like Autonomous Mobile Robots in Dynamic Obstacle Environments. *IEEE Access*, *11*, 137387-137400. https://doi.org/10.1109/ACCESS.2023.3339539

[16] Weixin, Z., Wang, D., Chen, Z., (2021). Autonomous operation path planning method for unmanned agricultural machinery (无人驾驶农机自主作业路径规划方法). *Transactions of the* Chinese Society of Agricultural Engineering, *37*(16), 1-7. https://doi.org/10.11975/j.issn.1002-6819.2021.16.001

[17] WEN Yu, HUANG Jiang-shuai, JIANG Tao, SU Xiao-jie, (2022). Safe and smooth improved time elastic band trajectory planning algorithm (安全平滑的改进时间弹性带轨迹规划算法). Control and Decision, 37(08), 2008-2016. https://doi.org/10.13195/j.kzyjc.2021.0068.

[18] Wu, J., Ma, X., Peng, T., Wang, H., (2021). An improved timed elastic band (TEB) algorithm of autonomous ground vehicle (AGV) in complex environment. *Sensors*, *21*(24), 8312. https://doi.org/10. 3390/s21248312

[19] Wu, M.H., Yu, J.C., Lin, Y.C., (2022, August). Study of Autonomous Robotic Lawn Mower Using Multi-Sensor Fusion Based Simultaneous Localization and Mapping. In 2022 International Conference on Advanced Robotics and Intelligent Systems (ARIS), Taipei, Taiwan, 2022, pp. 1-4, https://doi.org/10.1109 /ARIS56205.2022.9910445

[20] XIA Changgao, DING Weibing, HAN Jangyi, (2022). Research on path planning algorithm combining B-splines with improved APF (融合 B 样条与改进 APF 的路径规划算法研究). Journal of Chongqing University of Technology (Natural Science), 36(6), 48-54. https://doi.org/10.3969/j.issn.1674-8425(z)

[21] Yang, J., Chung, S. J., Hutchinson, S., Johnson, D., Kise, M., (2015, May). Omnidirectional-vision-based estimation for containment detection of a robotic mower. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 6344-6351). IEEE. https://doi.org/10.1109/ICRA.2015.7140090

[22] Yang Lili, Tang Xiaoyu, Wu Sixian, (2024). Local path planning for autonomous agricultural machinery on farm road (机耕道自动驾驶农机局部路径规划). Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 40(1), 27-36. (in Chinese with English abstract). https://doi.org/10.11975/ j.issn.1002-6819.202307231

[23] Yang, L., Xu, Y., Liang, Y., Qin, J., Li, Y., Wang, X., Wu, C., (2022). Extraction of straight field roads between farmlands based on agricultural vehicle-mounted LiDAR. International Journal of Agricultural and Biological Engineering, 15(5), 155-162. https://doi.org/10.25165/j.ijabe.20221505.6933

[24] Zhang Qiang, Chen Bingkui, Liu Xiaoyong, (2019). Ant colony optimization with improved potential field heuristic for robot path planning. Transactions of the Chinese Society for Agricultural Machinery, 50(5), 23-32. https://doi.org/10.6041/j.issn.1000-1298.2019.05.003

[25] Zhang, X., Zhu, T., Du, L., Hu, Y., Liu, H., (2022). Local path planning of autonomous vehicle based on an improved heuristic bi-rrt algorithm in dynamic obstacle avoidance environment. *Sensors*, *22*(20), 7968. https://doi.org/10.3390/s22207968

[26] Zhao Chunjiang, FAN Beibei, LI Jin, FENG Qingchun, (2023). Agricultural robots: Technology progress, challenges and trends (农业机器人技术进展、挑战与趋势). Smart Agriculture, 5(4), 1-15. https://doi.org/10.12133/j.smartag.SA202312030

[27] Zhong, X., Tian, J., Hu, H., Peng, X., (2020). Hybrid path planning based on safe A* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment. *Journal of Intelligent & Robotic Systems*, *99*(1), 65-77. https://doi.org/10.1007/s10846-019-01112-z

[28] Zhou, X., Wang, Z., Ye, H., Xu, C., Gao, F., (2020). Ego-planner: An ESDF-free gradient-based local planner for quadrotors. IEEE Robotics and Automation Letters, 6(2), 478-485. https://doi.org/10.1109/LRA.2020.3047728