SMOOTH-PLANNER: A ROBUST GRADIENT-BASED LOCAL TRAJECTORY PLANNING METHOD FOR NON-HOLONOMIC ORCHARD ROBOTS
/
SMOOTH-PLANNER:一种面向非完整约束果园机器人基于梯度的鲁棒局部轨迹规划方法

Fanjun MENG, Fa SUN, Mengmeng NI, Zhisheng ZHAO, Lili YI*
School of Agricultural Engineering and Food Science, Shandong University of Technology, Zibo / China
Tel: +86 18553308656; E-mail: yili0001@sdut.edu.cn
DOI: https://doi.org/10.35633/inmateh-79-27

Keywords: *Obstacle avoidance, trajectory optimization, B-spline, non-holonomic constraints*

ABSTRACT
This study proposes Smooth-Planner, a robust gradient-based local trajectory planning method for non-holonomic orchard robots operating in narrow row environments. The method employs a B-spline–based optimization framework that explicitly considers kinematic feasibility and collision avoidance without relying on dense global maps. Field experiments conducted in a standardized orchard demonstrate stable and repeatable performance, with lateral deviations effectively constrained and a minimum safety clearance of 0.90 m consistently maintained from crop rows. The results validate the practical safety and reliability of the proposed approach for autonomous agricultural operations.

摘要
本文提出了一种面向果园非完整约束移动机器人的稳健局部轨迹规划方法——Smooth-Planner。该方法基于 B 样条轨迹表示与梯度优化框架，在不依赖高精度全局地图的条件下，显式约束运动学可行性与避障安全性。标准化果园实地试验结果表明，该方法运行稳定、重复性良好，横向偏差得到有效控制，并始终保持不小于 0.90 m 的作物行安全间距。实验结果验证了该方法在自主农业作业中的安全性与实用性。

INTRODUCTION
With the rapid development of smart agriculture and digital orchard management, ground-based inspection robots are increasingly employed for tasks such as tree growth monitoring, precision spraying, and environmental data collection *(Dou et al., 2024; Hua et al., 2025)*. Unlike structured roads or open outdoor environments, orchard scenarios exhibit distinct spatial and perceptual characteristics, such as GNSS-denied canopy areas, narrow inter-row passages, and irregular branch protrusions *(Bayar et al., 2016)*. These features pose significant challenges to reliable autonomous navigation. Specifically, the complexity of the unstructured environment demands that the robot possesses high autonomy and strict safety awareness.

Local path planning is the core enabling technology for ensuring the safe and stable operation of these robots *(Cai et al., 2007)*. During routine inspection, robots must navigate continuously within confined corridors while coping with dynamic uncertainties, such as wind-blown branches and seasonal canopy variations. Crucially, the stability of local planning directly impacts operational efficiency and system safety. Frequent oscillations or abrupt stops caused by poor planning not only reduce task efficiency but also increase energy consumption and mechanical wear on the chassis *(Hu et al., 2015; Bai et al., 2019)*. Therefore, developing a local planner that satisfies requirements for real-time performance, robustness, and trajectory feasibility is of paramount importance.

A variety of local planning approaches have been applied in agricultural robotics. Classical sampling-based or search-based methods, such as the Dynamic Window Approach (DWA) *(Guo et al., 2024; Wang et al., 2024)* and Timed Elastic Band (TEB) *(Zai et al., 2023; Zheng et al., 2023)*, rely heavily on grid-based cost maps. In orchard environments characterized by sparse and noisy point clouds, maintaining a high-resolution grid map is computationally expensive, and discretization errors often lead to trajectory jitters or overly conservative behaviors *(Xie et al., 2018)*. More recently, gradient-based trajectory optimization methods *(Zhou et al., 2021)* have demonstrated superior smoothness and computational efficiency. However, most existing gradient-based planners are originally designed for holonomic aerial vehicles (UAVs). They often neglect the non-holonomic kinematic constraints of ground robots. Directly applying these methods to orchard robots typically results in infeasible trajectories that the vehicle hardware cannot execute, leading to tracking errors or collisions.

To address these challenges, this paper proposes a robust local trajectory planning method tailored for differential-drive orchard inspection robots, referred to as Smooth-Planner. The proposed approach adopts a gradient-based continuous trajectory optimization framework. Unlike traditional methods that depend on constructing explicit Euclidean Signed Distance Fields (ESDF) *(Han et al., 2019)* or grid maps *(Li et al., 2014)*, Smooth-Planner directly optimizes local trajectories by jointly considering smoothness, collision avoidance, and motion feasibility. By embedding geometry-aware obstacle avoidance constraints and non-holonomic motion limits into a continuously differentiable optimization formulation, the method generates high-quality executable trajectories even with incomplete perception.

The main contributions of this work are summarized as follows: (1) A map-free local trajectory optimization framework is established to handle the challenges of narrow orchard corridors and sparse perception, significantly reducing the dependency on high-precision environmental mapping. (2) A novel formulation of continuous collision and kinematic penalties is introduced, which strictly enforces non-holonomic constraints to ensure the generated trajectories are dynamically feasible for tracked ground robots. (3) Extensive field experiments in real orchard scenarios validate that the proposed method achieves high stability and maintains sufficient safety margins, proving its practical value for autonomous orchard inspection.

MATERIALS AND METHODS

System overview and problem formulation

The experimental platform used in this study is an electrically driven tracked robot designed for autonomous orchard inspection (Fig.1). To accurately account for collision risks in narrow inter-row environments, the robot geometry is abstracted as a bounding box with length L and width W. The perception system consists of an Ouster 3D LiDAR for local environment perception and a GNSS-RTK module for global pose estimation, while all planning algorithms are executed on an onboard NVIDIA AGX Orin computing platform. Under nominal operating conditions, the robot demonstrates high-precision trajectory tracking and reliable localization. This consistent baseline ensures that the deviations observed during obstacle avoidance are solely attributable to the proposed planning strategy rather than sensing noise or control instability.

Considering the differential steering mechanism, the robot is subject to non-holonomic kinematic *(Zhang et al., 2025)* constraints that prohibit instantaneous lateral motion. Under these constraints, the local trajectory planning problem is formulated as finding an optimal trajectory Φ within a receding horizon that minimizes a compound cost function J, while satisfying collision avoidance and kinematic feasibility requirements. Detailed formulations of the cost function and constraints are presented in the following sections.

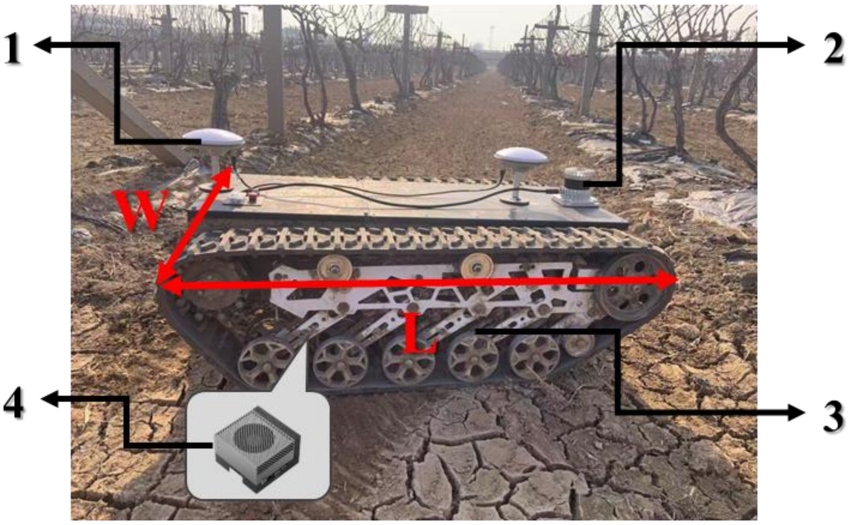

Fig. 1 -Tracked mobile platform for orchard robots
1. GNSS-RTK; 2. 3D LiDAR; 3. Crawler-Type Mobile Chassis; 4. Jetson AGX Orin.

Trajectory representation and motion modeling

To ensure smooth motion on uneven orchard terrain, the local trajectory is parameterized using a uniform B-spline curve *(Sun et al., 2024)*. The trajectory is uniquely defined by a set of control points $Q_i = \{Q_1, Q_2, \ldots, Q_{Nc}\}$, the spline degree P_b, and a uniform knot interval Δt, which enables direct adjustment of the trajectory during optimization.

Fig. 2 demonstrates the key geometric characteristics leveraged in this study **(1) Local Support**, which ensures that each trajectory segment is determined by only P_b+1 neighboring control points, allowing optimization constraints to be decoupled locally; **(2) Convex Hull**, implying that the continuous curve lies within the polygon formed by its control points, which simplifies collision checking; and **(3) Derivative Consistency**, allowing velocity and acceleration to be expressed as linear combinations of control points, enabling kinematic constraints to be enforced directly without explicit integration.

At low speed (< 1 m/s), lateral slip is neglected and the robot is modeled using a kinematic non-holonomic approximation *(Mínguez et al., 2002)*. Under this assumption, discrete velocity and acceleration associated with adjacent control points are approximated using finite differences with a constant time interval Δt as follows:

$$V_i = \frac{Q_{i+1} - Q_i}{\Delta t}, \; A_i = \frac{V_{i+1} - V_i}{\Delta t} \tag{1}$$

where:

V_i and A_i denote the discrete velocity and acceleration, respectively; Q_i denotes the $i\text{-}th$ control point of the B-spline trajectory; Δt is the uniform time interval between adjacent control points.

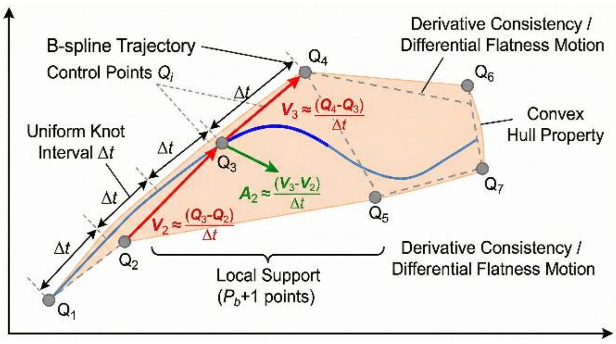

Fig. 2 - Illustration of B-spline trajectory properties

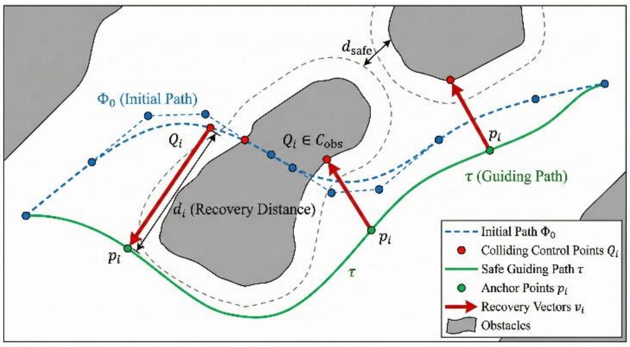

Fig. 3 --Front-end Logic - Optional

Collision-free Initial Trajectory Generator

To incorporate environmental awareness into the local planner, a collision-free initial trajectory generator is adopted as the front-end of the proposed method. Fig. 3 Schematic of the collision-free initial trajectory generation. The navigation layer initially outputs a globally smooth B-spline path Φ_0 that satisfies boundary constraints. When a colliding segment is detected, A^* search *(Hart et al., 1968)* is employed to generate a collision-free guiding path τ based on the topological structure of the environment. Unlike methods that rely on unstable gradients from irregular obstacle surfaces, τ serves as a safe topological attractor to guide the trajectory generation.

To precisely quantify collision risks, a safety threshold d_{safe} is first defined based on the robot's circumscribed radius *(Howell et al., 2022)*:

$$d_{safe} = \sqrt{\left(\frac{L}{2}\right)^2 + \left(\frac{W}{2}\right)^2} + \delta \tag{2}$$

where δ is a safety buffer.

For each control point Q_i, its distance to the nearest obstacle $d_{obs}(Q_i)$ is queried from the local obstacle cloud. Control points satisfying $d_{obs}(Q_i)<d_{safe}$ are identified as colliding points and grouped into a collision set C_{obs}.

For every colliding index i in C_{obs}, a unique anchor point p_i is assigned on the guiding path τ. Based on this correspondence, a collision recovery distance d_i and a recovery gradient vector $\boldsymbol{v_i}$ are defined as follows:

$$d_i = \|Q_i - p_i\| \tag{3}$$

$$\boldsymbol{v_i} = p_i - Q_i \tag{4}$$

where:

d_i denotes the Euclidean distance measuring the deviation of the colliding point from the safe path; $\boldsymbol{v_i}$ represents the gradient direction vector that pulls the control point Q_i back towards the safe anchor p_i.

Through this procedure, the feasible geometric references (p_i) and the associated recovery metrics (d_i, $\boldsymbol{v}_i$) are successfully generated, providing the essential collision-free initialization terms required for the subsequent gradient-based trajectory optimization.

Cost Function Formulation

To generate safe and executable trajectories, the local planning problem is formulated as a weighted multi-objective optimization (Fig. 4). The total cost function J is defined as:

$$\min_{Q} J = \lambda_s J_s + \lambda_c J_c + \lambda_k J_k \tag{5}$$

where:

J_s, J_c, and J_k represent smoothness, collision, and kinematic feasibility penalties, respectively. The coefficients λ_s, λ_c and λ_k are weighting factors that balance the trade-off between trajectory quality, safety, and feasibility.

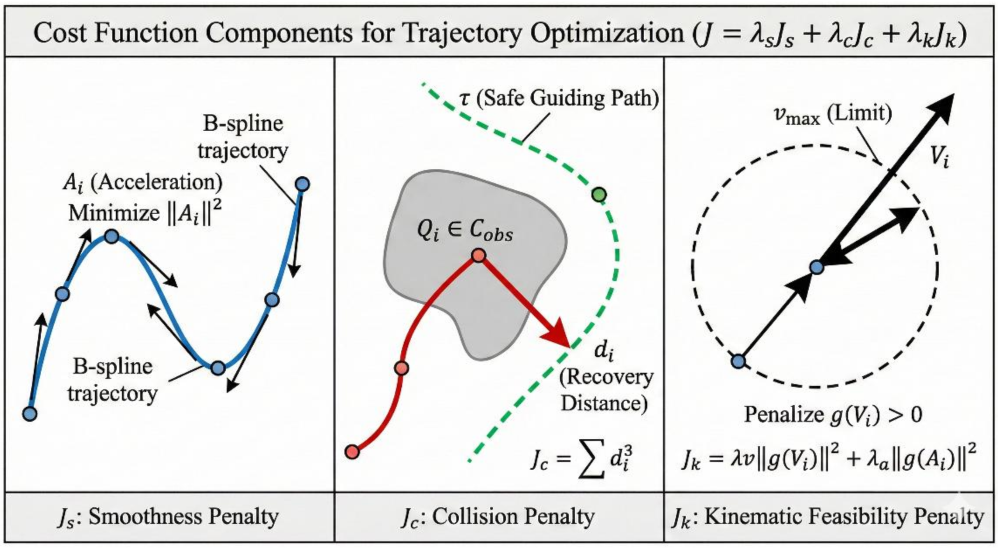

Fig. 4 - Cost function components

- Smoothness Penalty

Orchard environments are characterized by uneven terrain and varying soil conditions. Excessive acceleration can lead to chassis vibration, track slippage, or damage to onboard inspection sensors. To ensure trajectory smoothness and minimize control effort, a penalty on the control point acceleration is applied:

$$J_s = \sum_{i=1}^{N_c-2} \| A_i \|^2 \tag{6}$$

where:

N_c is the total number of control points in the B-spline trajectory, and A_i is the discrete acceleration vector derived in Equation (1).

The use of the squared L_2 norm serves two critical purposes. Mathematically, it provides a convex and continuously differentiable cost landscape, which facilitates rapid convergence for gradient-based optimization. Physically, the quadratic form heavily penalizes abrupt acceleration spikes, ensuring that the generated motion is gentle enough to prevent track slippage on soft orchard ground.

- Collision Penalty

The collision penalty strictly enforces the safety constraints. Unlike traditional methods that rely on computing gradients from irregular obstacle surfaces, this study leverages the recovery distance d_i derived from the safe guiding path. The penalty term J_c is formulated as:

$$J_c = \sum_{i \in C_{obs}} (d_i)^3 \tag{7}$$

This cubic formulation is applied exclusively to the set of colliding control points C_{obs} identified by the front-end. It ensures a continuous gradient that grows steeply as the control point deviates further from the safe anchor p_i, generating a strong restoring force that effectively pulls the trajectory back into the collision-free corridor defined by the guiding path.

- Kinematic Feasibility Penalty

To guarantee that the planned trajectory is executable by the hardware, dynamical limits are imposed as soft constraints. Velocities and accelerations exceeding the system limits (v_{max}, a_{max}) are penalized to ensure the generated commands remain within the robot's physical capabilities. The kinematic feasibility penalty J_k is formulated as:

$$J_{kinematic} = \sum_{i=1}^{N_c-1} \lambda_v \cdot g(V_i, v_{max}) + \sum_{i=1}^{N_c-2} \lambda_a \cdot g(A_i, a_{max}) \tag{8}$$

where:

$g(\cdot)$ is a squared penalty function activated only when constraints are violated:

$$g(x, x_{lim}) = \begin{cases} (\|x\|^2 - x_{lim}^2)^2 & \|x\| > x_{lim} \\ 0 & \|x\| \le x_{lim} \end{cases} \tag{9}$$

where:

V_i and A_i are the discrete velocity and acceleration vectors, v_{max} and a_{max} denote the maximum linear velocity and acceleration allowed by the chassis, and λ_v and λ_a are the weighting coefficients.

This formulation provides a continuously differentiable gradient that grows quadratically with the violation magnitude, effectively pulling the trajectory back into the feasible state space without introducing unnecessary computational complexity.

Gradient-based Trajectory Optimization

Given the composite cost function J, the trajectory generation is resolved by finding the optimal control points Q_i that minimize this cost. To ensure rapid convergence and avoid infeasible local minima, the optimization is initialized using the collision-free geometric reference. This initialization strategy provides a high-quality initial guess that naturally lies within the safe topological corridor, allowing the gradient-based solver to focus on local smoothing and refinement rather than global search.

Let x denote the decision variable vector formed by flattening the control points Q_i. The optimization is solved using an iterative gradient update rule:

$$x_{k+1} = x_k - \alpha_k \mathrm{H}_k^{-1} \nabla \mathrm{J}(x_k) \tag{10}$$

where:

$\nabla J(x_k)$ is the gradient of the cost function, α_k is the step size determined by line search, and H_k^{-1} represents the inverse Hessian matrix. To solve this efficiently within the limited computing budget, the L-BFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno) *(Liu et al., 1989)* algorithm is adopted. Unlike standard Newton methods, L-BFGS approximates H_k^{-1} using a limited history of past updates, significantly reducing memory consumption and computational complexity for the high-dimensional B-spline problem.

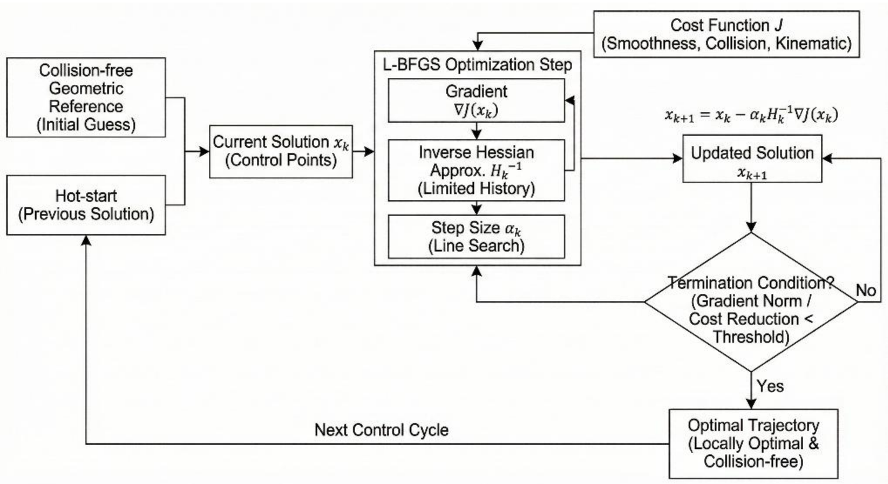

Fig. 5 - Flowchart of numerical optimization algorithm

In the dynamic context of orchard inspection, the optimizer employs a hot-start strategy, utilizing the solution from the previous control cycle as the initial guess for the current frame. The iteration terminates when either the gradient norm or the relative cost reduction falls below a predefined threshold, ensuring that a stable, locally optimal, and collision-free trajectory is regenerated in real-time.

RESULTS

Experimental Setup and Baseline Validation

Field experiments were conducted in a standard orchard environment in Shandong Province, China. The test site featured typical inter-row characteristics with an average row spacing of 4.5 m and a plant spacing of 1.5 m. The robot was tasked to follow predefined global paths, with all sensor data and trajectory logs recorded at 0.2 s intervals. throughout the trials, a constant forward speed of 0.6 m/s was maintained to ensure consistent motion dynamics.

Prior to the obstacle avoidance tests, the baseline path tracking accuracy was verified. The results showed that the robot maintained a lateral tracking error of less than 3.0 cm on straight paths. This high-precision baseline ensures that the significant deviations observed in subsequent experiments were solely driven by the active obstacle avoidance algorithm rather than control instability.

Local Obstacle Avoidance Performance

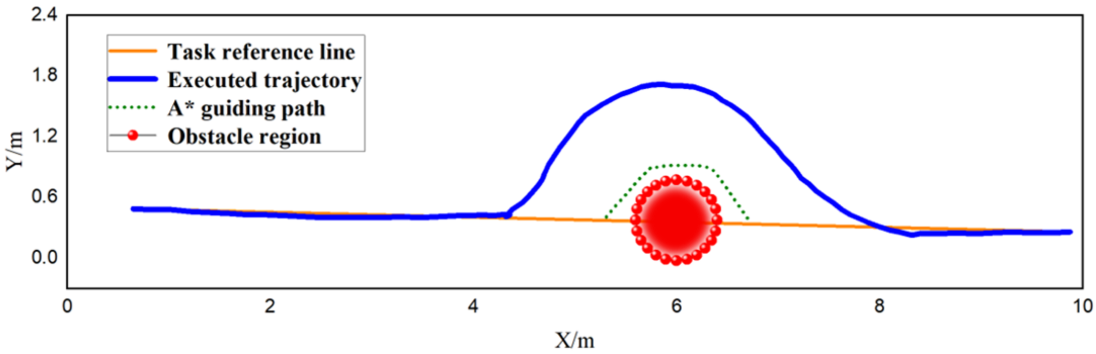

Fig. 6 -Obstacle avoidance trajectory

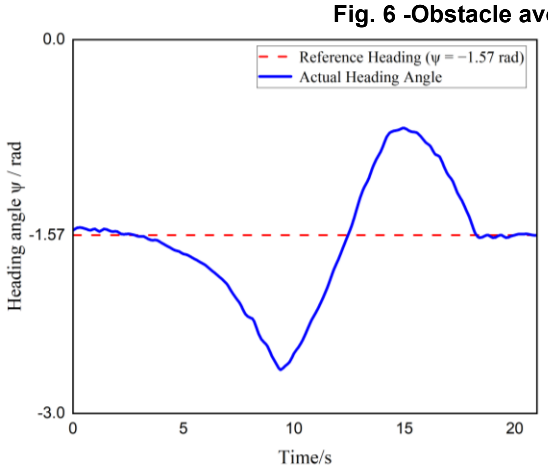

Fig. 7 - The curve of the heading angle over time

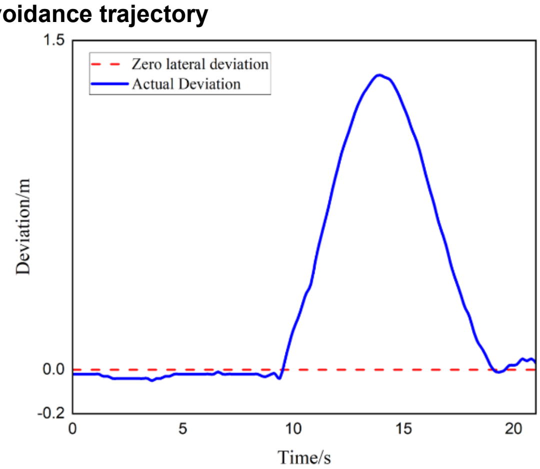

Fig. 8 - The curve of lateral deviation over time

Fig. 6 illustrates the spatial trajectory of the robot during a representative avoidance maneuver. The local avoidance process is initiated at approximately x = 4.35 m, where the robot smoothly deviates from the task path guided by the optimized B-spline trajectory. The maneuver is completed at around x = 8.32 m. Although a slight overshoot occurs immediately after obstacle clearance, it is rapidly attenuated by the feedback controller, allowing the trajectory to converge asymptotically back to the global reference path without persistent oscillations. The corresponding heading angle response is presented in Fig. 7. It is noteworthy that while the reference heading remains constant, the actual heading exhibits an anticipatory deviation as the robot perceives the obstacle, reaching a maximum difference of approximately 1.07 rad at t = 9.4 s. This heading-first response precedes significant lateral displacement, demonstrating the kinematic feasibility of the smooth transition into the avoidance maneuver. Fig. 8 quantifies the lateral deviation relative to the global path. The deviation profile exhibits a single-peak structure with a maximum amplitude of 1.34 m. Given the orchard row spacing of 4.5 m, the operational boundary lies at 2.25 m from the path centerline. Consequently, the maximum deviation of 1.34 m preserves a substantial safety margin of approximately 0.9 m from the vine rows. This result confirms that the proposed Smooth-Planner effectively utilizes the available free space to bypass obstacles while strictly adhering to the agronomic safety constraints of the orchard environment.

Stability and Robustness Validation

To further evaluate the practical applicability of the proposed method under realistic field conditions, additional experiments were conducted. Fig. 9 visualizes the onboard local point cloud perception during the obstacle avoidance maneuver, alongside corresponding real-world snapshots. As observed, despite the unstructured environmental noise typical of orchards, the obstacle geometry is distinctly captured by the LiDAR. The robot successfully executes a smooth bypass within the 4.5 m inter-row space. The strong consistency between the point cloud data and the physical environment confirms the reliability of the perception-planning pipeline.

To quantitatively assess the repeatability and stability of the system, the obstacle avoidance experiment was replicated in ten independent trials under identical conditions. Key performance indicators were extracted from each trial, and the statistical results are summarized in Table 1.

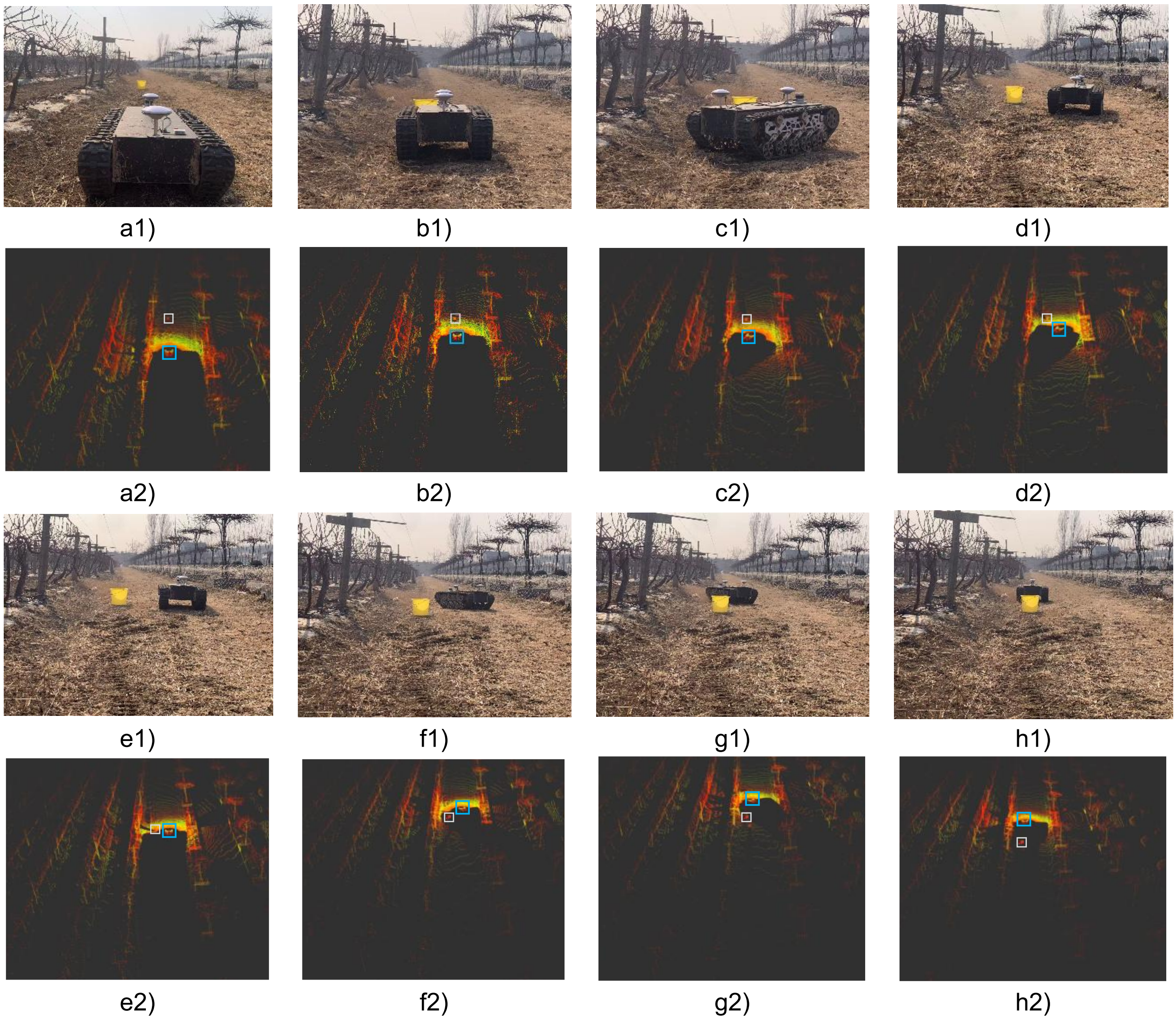

Fig. 9 – Real-world images and point cloud diagrams of obstacle avoidance operation

a1)–h1) – real-world images; a2)–h2) – LiDAR point cloud

Table 1

Statistical Metrics of Obstacle Avoidance Performance (N=10)

Performance Metric	Mean	Std. Dev.	Description
Max. Lateral Deviation(m)	1.32	0.08	Peak deviation from the global path
Min. Safety Margin(m)	0.9	0.07	Distance to the crop row boundary
Avoidance Duration(s)	9.65	0.15	Time elapsed during the maneuver
Total Path Length(m)	4.12	0.05	Length of the executed trajectory

As indicated in Table 1, the system demonstrates high repeatability. The Max. Lateral Deviation averaged 1.32 m with a standard deviation of only 0.08 m. This minimal variance is critical for agricultural automation, as it indicates that the robot's trajectory is highly predictable and robust against sensor noise across different runs.

Furthermore, by correlating these statistics with the 4.5 m row spacing, the Minimum Safety Margin is calculated to be approximately 0.9 m. Even accounting for the variance, the robot consistently maintains a sufficient buffer from the vine rows, thereby effectively mitigating collision risks with the crop canopy. The consistent values for Lateral Deviation RMSE and Avoidance Duration further validate that the proposed Smooth-Planner generates stable, feasible, and robust trajectories suitable for continuous orchard inspection tasks.

CONCLUSIONS

In this study, a robust local trajectory planning method, Smooth-Planner, was developed to address the challenges of autonomous navigation in narrow and unstructured orchard environments. By integrating a collision-free topological guiding path with a gradient-based continuous optimization framework, the proposed method generates smooth trajectories that strictly satisfy the non-holonomic kinematic constraints of differential-drive robots.

Field experiments conducted in a standardized orchard validated the effectiveness of the system. The results demonstrated that the robot could stably bypass obstacles with highly repeatable behavior, exhibiting an average maximum lateral deviation of 1.32 m with a minimal standard deviation of 0.08 m. Crucially, quantitative analysis confirmed that even during avoidance maneuvers, a minimum safety margin of approximately 0.93 m was maintained from the crop rows, ensuring agronomic safety. Overall, this work provides a reliable, map-free, and practical solution for the deployment of inspection robots in complex agricultural scenarios.

ACKNOWLEDGEMENT

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

REFERENCES

[1] Bai, G., Liu, L., Meng, Y., Luo, W., Gu, Q., & Liang, C. (2019). Study of obstacle avoidance controller of agricultural tractor-trailers based on predictive control of nonlinear model (基于非线性模型预测控制的农用拖挂车避障控制器研究). *Transactions of the Chinese Society for Agricultural Machinery*, 50(4), 356–362. https://doi.org/10.6041/j.issn.1000-1298.2019.04.041

[2] Bayar, G., Bergerman, M., Konukseven, E. I., & Koku, A. B. (2016). Improving the trajectory tracking performance of autonomous orchard vehicles using wheel slip compensation. *Biosystems Engineering*, 146, 149–164. https://doi.org/10.1016/j.biosystemseng.2015.12.019

[3] Cai, J., Zhao, J., Rath, T., & Kawollek, M. (2007). Path planning for obstacle avoidance of fruit harvesting robot (水果收获机器人避障路径规划). *Transactions of the Chinese Society for Agricultural Machinery*, 38(3), 102–105. https://doi.org/10.3969/j.issn.1000-1298.2007.3.070325

[4] Dou, H., Chen, Z., Zhai, C., Zhang, B., Ding, X., & Wang, X. (2024). Research progress on autonomous navigation technology for orchard intelligent operation equipment (果园智能化作业装备自主导航技术研究进展). *Transactions of the Chinese Society for Agricultural Machinery*, 55(4), 1–22. https://doi.org/10.6041/j.issn.1000-1298.2024.04.001

[5] Guo, H., Li, Y., Wang, H., Wang, C., Zhang, J., Wang, T., Rong, L., Wang, H., Wang, Z., Huo, Y., Guo, S., & Yang, F. (2024). Path planning of greenhouse electric crawler tractor based on the improved A* and DWA algorithms. *Computers and Electronics in Agriculture*, 227, 109596. https://doi.org/10.1016/j.compag.2024.109596

[6] Han, L., Gao, F., Zhou, B., & Shen, S. (2019). FIESTA: Fast incremental Euclidean distance fields for online motion planning of aerial robots. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4423–4430. https://doi.org/10.1109/IROS40897.2019.8968199

[7] Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107. https://doi.org/10.1109/TSSC.1968.300136

[8] Howell, T. A., Le Cleac'h, S., Singh, S., Florence, P., Manchester, Z., & Sindhwani, V. (2022). Trajectory optimization with optimization-based dynamics. *IEEE Robotics and Automation Letters*, 7(3), 6750–6757. https://doi.org/10.1109/LRA.2022.3152696

[9] Hu, J., Gao, L., Bai, X., Li, T., & Liu, X. (2015). Review of research on automatic guidance of agricultural vehicles (农业机械自动导航技术研究进展). *Transactions of the Chinese Society of Agricultural Engineering*, 31(10), 1–10. https://doi.org/10.11975/j.issn.1002-6819.2015.10.001

[10] Hua, W., Zhang, Z., Zhang, W., Liu, X., Hu, C., He, Y., Mhamed, M., Li, X., Dong, H., Saha, C. K., Khan, W. U., Abid, F., & Abdelhamid, M. A. (2025). Key technologies in apple harvesting robot for standardized orchards: A comprehensive review of innovations, challenges, and future directions. *Computers and Electronics in Agriculture*, 235, 110343. https://doi.org/10.1016/j.compag.2025.110343

[11] Li, H., Tsukada, M., Nashashibi, F., & Parent, M. (2014). Multivehicle cooperative local mapping: A methodology based on occupancy grid map merging. *IEEE Transactions on Intelligent Transportation Systems*, 15(5), 2089–2100. https://doi.org/10.1109/TITS.2014.2309639

[12] Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1–3), 503–528. https://doi.org/10.1007/BF01589116

[13] Mínguez J., Montano L., & Santos-Victor J. (2002). Reactive navigation for non-holonomic robots using the ego-kinematic space. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, 3, 3074–3080. https://doi.org/10.1109/ROBOT.2002.1013699

[14] Sun, Y., & Wang, H. (2024). Research on B-spline curve trajectory planning method considering curvature. *8th International Conference on Robotics, Control and Automation (ICRCA),* 211–216. https://doi.org/10.1109/ICRCA60878.2024.10649358

[15] Wang, Y., Fu, C., Huang, R., Tong, K., He, Y., & Xu, L. (2024). Path planning for mobile robots in greenhouse orchards based on improved A* and fuzzy DWA algorithms. *Computers and Electronics in Agriculture*, 227, 109598. https://doi.org/10.1016/j.compag.2024.109598

[16] Xie, Y., Yin, J., Yu, C., He, K., Hu, X., & Li, R. (2018). Obstacle avoidance navigation algorithm and analog experiment for wheeled AGV running along vineyard ridge road (轮式 AGV 沿葡萄园垄道行驶避障导航算法与模拟试验). *Transactions of the Chinese Society for Agricultural Machinery*, 49(7), 13–22.

[17] Zai, W., Lin, Q., Wang, S., & Lin, X. (2023). Path planning for wheeled robots based on the fusion of improved A* and TEB algorithms. *2023 China Automation Congress (CAC)*, 3257–3261. https://doi.org/10.1109/CAC59555.2023.10451976

[18] Zhang, W., Sun, T., Li, Y., He, C., Xiu, X., & Miao, Z. (2025). Optimal motion planning and navigation for nonholonomic agricultural robots in multi-constraint and multi-task environments. *Computers and Electronics in Agriculture*, 238, 110822. https://doi.org/10.1016/j.compag.2025.110822

[19] Zheng, H., Dai, M., Zhang, Z., Xia, Z., Zhang, G., & Jia, F. (2023). The navigation based on hybrid A star and TEB algorithm implemented in obstacles avoidance. *29th International Conference on Mechatronics and Machine Vision in Practice (M2VIP),* 1–6. https://doi.org/10.1109/M2VIP58386.2023.10413435

[20] Zhou, X., Wang, Z., Ye, H., Xu, C., & Gao, F. (2021). 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