Online path tracking with an integrated H∞ robust adaptive controller for a double-Ackermann steering robot for orchard waypoint navigation

by Shamshiri, R.R., Azimi, A., Behjati, M., Ghasemzadeh, A., Dworak, V., Weltzien, C., Karydis, K. and Cheein, F.A.A.

International Journal of Intelligent Robotics and Applications
https://doi.org/10.1007/s41315-024-00379-2

REGULAR PAPER

Online path tracking with an integrated H_∞ robust adaptive controller for a double-Ackermann steering robot for orchard waypoint navigation

Redmond R. Shamshiri[1,2] · Alireza Azimi[3] · Maryam Behjati[1] · Aliakbar Ghasemzadeh[3] · Volker Dworak[1] · Cornelia Weltzien[1,2] · Konstantinos Karydis[4] · Fernando A. Auat Cheein[5]

Received: 12 May 2024 / Accepted: 23 August 2024

Abstract
Navigation of agricultural mobile platforms in small-scale orchards poses challenges due to narrow row-end turning spaces and the need for precise path tracking in the presence of disturbances. The objective of this study is to improve path following and rapid turning maneuvers for a double-Ackermann steering robot by employing a simulation approach for PID-based waypoint following enhanced by learning-based H_∞ robust adaptive control. With the zero-speed turning radius of the robot measured at 2.85 m, the primary question to address is determining the minimum achievable turning radius using the two controllers. For this purpose, a versatile framework for fine-tuning and analyzing of the controllers is presented in MATLAB Simulink blocks interfaced with the virtual replica of the robot in CoppeliaSim. A comparative study between the controllers is carried out involving three experiments: offline path following with a fixed number of predefined waypoints, online path following with continuously updated waypoints forming paths, and path tracking with disturbance rejection using the H_∞ controller to reduce the radius of row-end turnings. Results indicate that while the PID controller achieves a minimum row-end turning radius of 3.0 m, the learning-based H_∞ controller surpasses it with a minimum radius of 2.9 m. It is observed that a minimum of 4 waypoints is required for the PID controller to perform effective row-end turning in the offline experiment, with a higher number of waypoints enabling the robot to navigate through complex geometries and tight turns more effectively. Moreover, by incorporating an actor-critic structure, it has been demonstrated that the learning-based H_∞ controller maintains stability even when facing wheel slippage disturbances, and outperforms the PID controller in online path tracking, particularly when maneuvering along a half-circle path. The framework proposed in this study contributes to improving autonomous navigation, particularly in determining the optimal number of waypoints and path configurations required for navigating agricultural robots with varying dimensions and steering mechanisms.

Keywords Agricultural robotics · Path tracking · H_∞ robust adaptive controller · Simulation · Autonomous navigation

1 Introduction

The integration of mobile platforms with different dimensions, steering mechanisms, and control systems has shown the potential to contribute to higher profits from agricultural fields (Fragapane et al. 2021), particularly in practices such as field mapping (Tiozzo Fasiolo et al. 2023), crop scouting (Virlet et al. 2017), and autonomous spraying (Meshram et al. Mar. 2022) that involves repetitive manual work. One area where these robots face challenges is in navigating within small-scale orchards, where the limited spaces between plant rows and the necessity for precise maneuverability at row-end turnings require precision control solutions. Larger turning radii of robots increase the risk of damaging crops, especially in densely planted orchards where branches may extend into pathways. In addition, since these robots are typically battery-powered, reducing the operation time and energy is crucial for continued navigation in the field, necessitating the implementation of optimized waypoints following (Farooq et al. 2023) and path tracking control algorithms (Kivrak et al. 2022) that aim at minimizing unnecessary movements and optimizing routes.

During actual navigation in dynamic and unstructured orchards with variations in terrain, mobile platforms often exhibit nonlinear dynamics, making the design of an

Extended author information available on the last page of the article

Published online: 20 September 2024

effective navigation controller challenging, particularly when traveling on uneven terrains, slopes, bumps, and the presence of Nonholonomic constraints. In these situations, sensors such as Odometry(Aqel et al. 2016), GPS(Ryu et al. 2015), LiDAR(Raj et al. 2020), or cameras may also suffer from noise, biases, or measurement inaccuracies, affecting the accuracy of path tracking. For example, a fuzzy logic path-tracking controller that benefits from the nonlinear least square method and GNSS-based navigation system (Gyagenda et al. 2022) showed significant tracking errors due to signal reception issues caused by obstructions such as tree branches and leaves, leading to substantial deviations from the reference path. While systems equipped with high-precision RTK GPS (Li et al. 2022) have demonstrated promising results in open field cultivation, their reliability is reduced within enclosed spaces like closed-field crop productions or GPS-denied environments due to potential signal blockage. Integrating IMUs enhance localization accuracy (Chen et al. 2018), however, combining multiple sensor types also increases the complexity and cost of the navigation system. Therefore waypoint-based path following must operate within real-time constraints and employ correction signals to ensure timely responses to these changing conditions.

Waypoint-based path following involves guiding the robot along a predefined route by sequentially navigating through a series of specified locations typically defined as coordinates in the robot's workspace until all waypoints have been reached. Each waypoint represents a key position or landmark that the robot needs to reach during its mission. The path between waypoints can be predefined offline (Weber et al. 2023; Asadi et al. 2020) using a fixed set of waypoints based on prior knowledge of the environment, or generated online (Asadi et al. 2020) using a dynamic mechanism in real-time (Kan et al. 2020) or a defined time frame (Kan et al. 2021). This path may consist of straight lines, curves, or complex trajectories depending on the terrain and mission requirements. As the robot navigates, it continuously compares its current position with the coordinates of the next waypoint and adjusts its trajectory and control inputs to steer toward the target waypoint. Feedback control mechanisms such as Proportional-Integral-Derivative (PID) control (He et al. 2022), have been widely used in agricultural mobile platforms due to their simplicity and effectiveness in continuously adjusting the robot's steering and speed. However, traditional controllers often lack robustness in the presence of nonholonomic constraints, which restrict the robot's motion beyond its position and orientation in space. These constraints, such as limitations on velocity or movement direction, can significantly impact the efficacy of path planning and control algorithms. Experiments with PID controllers have revealed difficulties in handling uncertainties and disturbances inherent in real-world environments, such as wheel slippage on varying soil conditions or unexpected obstacles. These challenges cannot be straightforwardly addressed within basic differential equations solely considering the system's positions and velocities.

To address these limitations, advanced control strategies such as Stanley (AbdElmoniem et al. Nov. 2020), Model Predictive Control (MPC) (Zhou et al. 2023), Sliding Mode Control (SMC) (An et al. 2023), Pure Pursuit (Xu et al. 2022; Samuel et al. 2016), Linear Quadratic Regulator (LQR) (Chen et al. 2023), Artificial Neural Networks (ANNs) (Vulpi et al. 2021), Fuzzy Logic Control (FLC) (Wang et al. Jul. 2021), and Reinforcement Learning (RL) (Chang et al. 2021) have been tested in the navigation systems of agricultural mobile platforms (Wijayathunga et al. 2023). Nonlinear control methods, such as MPC or SMC utilizes a dynamic model of the robot to predict its future trajectory, facilitating proactive adjustments to control inputs and enhancing robustness against disturbances (Deniz et al. 2024; Prado et al. 2020). Other advanced algorithms, particularly H_∞ robust adaptive control, are well-suited for applications like orchard navigation, where precise maneuverability and robust performance are essential for successful operation. For instance, the H_∞ robust adaptive controller reported in Khalaji and Moosavian (2014) offers robustness against uncertainties and disturbances by dynamically adapting control parameters in real time based on feedback from sensors and environmental conditions. To enhance path tracking performance, integrating state estimation techniques such as Kalman filtering (Greenberg and Tan 2020) or observer designs (Guevara et al. 2023) are also used to enable the estimation of unmeasured states, providing valuable insights into the robot's position, orientation, and velocity. Accurate state estimation leads to precise control actions based on comprehensive system information, resulting in improved overall path tracking performance. Nevertheless, experimentation and validation in simulation environments and real-world agricultural settings are essential for assessing the effectiveness and applicability of these control methods in practical agricultural robotics applications. Table 1 provides a summary of the control algorithms used for path tracking, outlining their respective benefits and limitations.

The selection of an agricultural mobile platform based on kinematic parameters (i.e., wheelbase, width, maximum steering angle, ground clearance, wheel diameter, and center of gravity), alongside the choice of path tracking controller, influences the robot's ability to navigate (Campbell et al. 2021), crucially affecting row-end turning. Longer wheelbases offer enhanced stability, smoother maneuverability, and increased payload capacity, making them ideal for navigating rough terrain and carrying heavier loads. However, they may struggle with tight turns and have larger turning radii, potentially with lower efficiency in confined spaces. Conversely, shorter wheelbases are preferred for greater maneuverability, agility, and adaptability to diverse terrains,

Table 1 Summary of the control methods used for path tracking of mobile platforms, with agricultural applications

Control methods	Advantages and disadvantages	Application example
pure pursuit	Streamlined layout for precise vehicle position control. Challenging to implement in high-speed and large road curvature scenarios	MAFV robot with differential drive used for wheat seeding (Yan et al. 2024)
PID	Simplicity and ease of engineering application. Poor versatility and challenging parameter tuning	Husky A200 with Skid-steering used in apple orchard (Blok et al. 2019)
Model-Free control	Simple control structure, but often perceived as a black box, posing challenges to stability analysis	Used in apple-picking robot platform (Chen et al. 2019)
LQR	Facilitates achievement of closed-loop optimal control objectives. Lack of robustness due to reliance on linear models	Used in skid-steering mobile robot for outdoor fields (Amertet et al. 2024)
Feedforward and Feedback	Capable of handling external disturbances, modeling errors, and sensor noise. Necessitates expensive sensors for vehicle data acquisition	Komodo-01 with Crawler Differential Drive for outdoor fields (Huang et al. 2018)
MPC	Handling system constraints and future prediction. Struggling with stability analysis and computational cost	SPIDO with Double steering off-road for outdoor fields (Fnadi et al. 2019)
H_∞ Control	Easily establishes hard constraints with strong robustness. Requires complex solutions, intricate theory, and only handles bounded disturbances	Yanmar VP6E rice transplanter for paddy field (Majdoubi et al. 2023)
SMC	Fast response, resistant to parameter changes and disturbances Path tracking with SMC requires an adaptive mechanism to reduce chatter	AgRover with 4WD/4W steering for outdoor fields (Tu et al. 2019)
Robust MPC	Handles system states and constraints with robustness. Faces challenges in stability analysis and incurs high computational costs	AROS with skid- steering used in apple orchard (Wen Zhu et al. 2023)

making them well-suited for navigating narrow rows and responding quickly to obstacles. Yet, they may sacrifice stability and have limited payload capacity, necessitating careful load distribution and terrain assessment. Wider robots offer greater stability, traction, and higher payload capacity, ensuring reliable performance in harsh field conditions, however, they may face challenges maneuvering in tight spaces and during transportation due to their increased width.

This study was motivated by the need for a simulation framework to design and experiment with different path following control strategies, to be implemented on agricultural mobile platforms with varying sizes and steering mechanisms. The aim was to evaluate their performance in accurate path following and determine the minimum space required for row-end turnings. Therefore, the main objective of the study was to design and compare two controllers for guiding a simulated double-Ackerman steering mobile platform along various waypoints and reference paths within a simulated orchard environment with narrow row-ends. Knowing the zero-speed turning radius of the robot, the primary question addressed in this study is focused on determining the minimum turning radius at which the proposed controllers could maintain stability during row-end maneuvers. To achieve this, a PID controller for offline and online waypoint-based path following, and a learning-based H_∞ robust adaptive controller for path tracking, were designed and evaluated based on their performances in disturbance rejection, tracking error, and the ability to achieve the smallest radius during row-end turns. This evaluation was conducted using a customized simulation framework that integrates MATLAB Simulink (MathWorks Inc, MA, USA) interfaced with the CoppeliaSim robotic simulator environment (Coppelia Robotics AG, Zurich, Switzerland), which enabled the implementation and visualization of the proposed control strategies' performance, allowing the robot to adapt to varying row-end turning patterns and uncertainties characteristic of real-world orchard settings.

2 Materials and methods

2.1 Kinematic of the robot platform

The kinematic equations derived for this study are based on the four-wheel-drive (4WD) double-Ackerman steering mobile platform developed by Irus (IRUS Motorgeräte GmbH, Burladingen, Germany), as illustrated in Fig. 1a. The platform features a track width of $w = 1.4$ m, a wheelbase of $L = 2.0$ m, a maximum steering angle of $\delta_{max} = 25°$, a maximum forward speed of $V_{max} = 10$ km/h, and an approximate weight of 475 kg that utilizes a 2-cylinder petrol engine with 20.1 kW (27 HP) output power. A schematic representation of the robot is provided in Fig. 1.b to describe the kinematic model by analyzing the instantaneous motion of the robot's center of mass (COM)

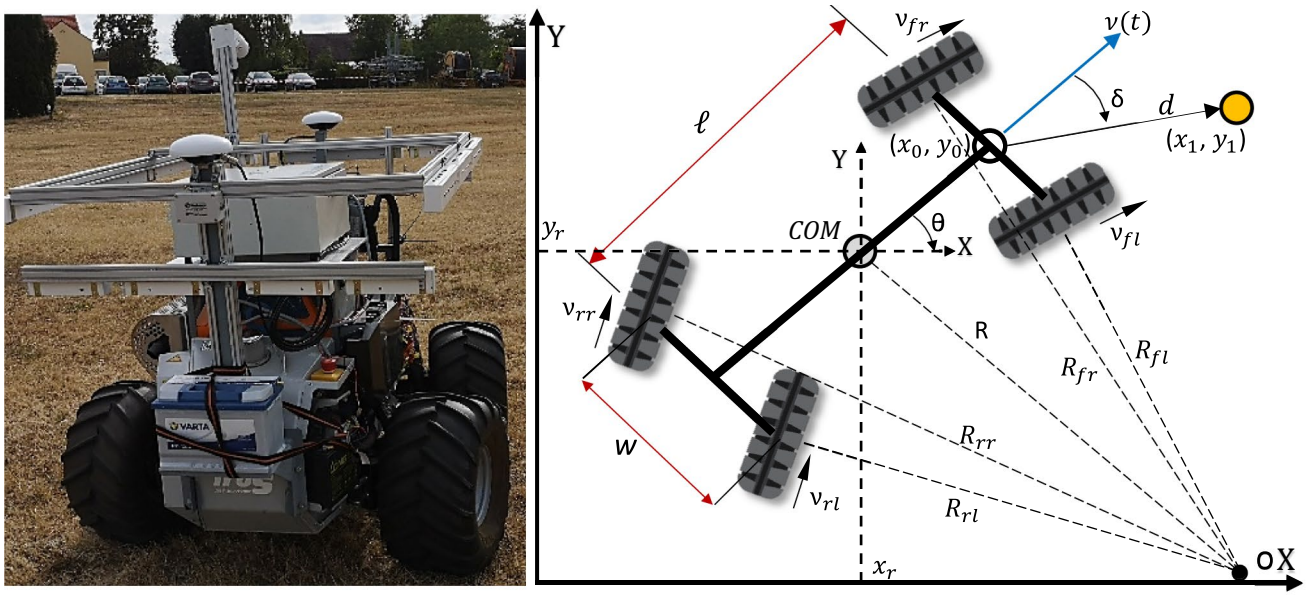

Fig. 1 The double-Ackerman steering vehicle platform used for path tracking controller design (left) and the kinematic representation of the robot (right)

while considering the constraints imposed by the Ackermann steering geometry. Here x_r and y_r denote the Cartesian coordinates of the COM, θ is the heading angle calculated using $arctan^2\left(y-y_0, x-x_0\right)$, and δ is the steering angle of the wheels. The state vector description of the robot is given by $q=\left[x, y, \theta, \delta\right]^T$. Additionally, v and ω represent the linear and angular velocities of the COM, respectively.

The robot can achieve a minimum row-end turning radius of 2.84 m (d = 5.68 m), as calculated by Eq. 1. The simplified robot's kinematic model is based on the assumption of pure rolling without slippage, which can be expressed as $\dot{q}=S(q)U$, where the matrix S is given by Eq. 2 and is used in conjunction with the vector $U=[v, \omega]^T$. The linear velocity v and angular velocity ω are related to the steering angle δ and the wheel speeds through the Ackermann steering geometry constraints. The kinematic equations governing the motion of the robot in the absence of disturbances are expressed by $\dot{x}=v\cos\theta$, $\dot{y}=v\sin\theta$, $\dot{\theta}=v/L(\tan\delta)$, and $\dot{\delta}=\omega$ (Corke et al. 2023). The superscript dot denotes the time derivative.

$$d=\frac{L}{\tan\left(\delta_{max}\right)}+w \tag{1}$$

$$S=\begin{bmatrix}\cos(\theta) & 0\\ \sin(\theta) & 0\\ \frac{\tan\delta}{L} & 0\\ 0 & 1\end{bmatrix} \tag{2}$$

It should be noted that these kinematic equations capture the non-holonomic constraints imposed by the ideal Ackermann steering geometry, wherein the robot cannot move laterally without changing its orientation. To account for real-world conditions, such as external disturbances to the system such as slippage and wind in an orchard environment, the non-ideal kinematics of the system (Bai et al. 2022) was considered as $\dot{q}=S(q)U+k(x)\beta$, where $k(x)$ is the disturbance matrix, and β is a three-dimensional vector representing the disturbance and β_{amp} is the amplitude of it. Consequently, the non-ideal kinematics of the system are given in Eq. 3, where T_1 and T_2 denote the specific time points when disturbances occur (i.e. the slippage occurs continuously from T_1 to T_2), with $H(.)$ representing the unit step function (Bai et al. 2022). A mathematical examination of this formulation reveals that the disturbance matrix ($k(x)$) is a 4 × 4 identity matrix. Hence the disturbance vector β is precisely defined by Eq. 4. The derived kinematic model served as the foundation for the waypoint following and path tracking controller design, as well as the simulation analysis presented this study.

$$\begin{aligned}\dot{x}&=v\cos\theta+\beta_{amp}H\left(t-T_1\right)\sin\theta\\ \dot{y}&=v\sin\theta-\beta_{amp}H\left(t-T_1\right)\cos\theta\\ \dot{\theta}&=v\frac{\tan\delta}{L}+\beta_{amp}H\left(t-T_1\right)\frac{\tan\delta}{L}\\ \dot{\delta}&=\omega+\beta_{amp}H\left(t-T_1\right)-H\left(t-T_2\right)\end{aligned} \tag{3}$$

$$\beta = \beta_{amp} H(t - T_1) \times \left[\sin\theta, \cos\theta, \tan\delta/L, 0\right]^T + \left[0, 0, 0, H(t - T_1) - H(t - T_2)\right]^T \quad (4)$$

2.2 Design of waypoint following with PID controller

The control strategy for waypoint following involved guiding the robot along a path segment that connects its current position to the next waypoint. This was achieved by simultaneously adjusting both the robot's steering and speed using two PID controllers that generates two control signals, steering $\delta(t)$, and speed $\vartheta(t)$. The first controller minimizes orientation errors to align the robot with the next waypoint, while the second PID controller regulates the robot's linear velocity, ensuring a desired speed as it approaches waypoints. Moreover, the gains of both controllers were tuned using a trial-and-error approach within our simulation framework. By iterating and making adjustments, we managed to achieve satisfactory performance (as will be shown in the results section). The kinematic equation of the robot was discretized with time steps of Δt as described in Eq. 5, using Euler integration method to update the robot's position, allowing the controller to calculate new steering angles and linear velocities while enforcing constraints, ensuring that the robot's motion stays within safe operational bounds and performs smooth and accurate navigation towards the waypoints. The kinematic equations provided in Eqs. 5, 6, 7, along with the two PID controllers described in Eqs. 8, 9, 10, 11, were implemented in MATLAB as a simulation loop that iterates over a predefined number of time steps Δt, during which the robot continuously updates its position, orientation, and control inputs. Here v_{max} and v_{min} are the maximum and minimum linear velocity of the robot, ω is the angular velocity, L is the wheelbase, R is the wheel radius, δ_{max} is the maximum allowable steering angle, and (x_t, y_t) and θ_t are respectively the robot current position and orientation. In this scheme, the controller first calculates the desired orientation θ_d that the robot needs to align with the line connecting its current position (x_t, y_t) to the next desired waypoint (x_d, y_d), generating an error signal $\acute{e}_\theta(t)$ based on its current orientation θ_t. This error is adjusted to ensure it falls within the range -π to π, resulting the final error signal $e_\theta(t)$. The controller also calculates the integral of this error as $\int_0^t e_\theta(\tau)d\tau$ to track the accumulated error over time, and the derivative of this error as $\frac{de(\theta)}{dt}$ to determine the rate of change of the error signal and generate steering control $\delta(t)$ that influence the robot's trajectory towards the target point. A similar approach is used for the speed control by comparing the robot distance to the next waypoint with a reference distance $dist_{ref}$, resulting in a speed error signal of $e_v(t)$.

Finally, in each time step of the simulation, the change in the x–y coordinate of the robot's position over a small time interval Δt is calculated using Eq. 5. The pseudo-code of the controller is provided in Table 2.

$$\left[x_{t+1}, y_{t+1}\right] = \left[x_t + v_t.\cos(\theta_t).\Delta t, y_t + v_t.\sin(\theta_t).\Delta t\right] \quad (5)$$

$$\omega_t = \frac{v}{L}.\tan(\delta(t)) = \frac{\theta_{t+1} - \theta_t}{\Delta t} \quad (6)$$

$$\theta_{t+1} = \theta_t + \left(\frac{v}{L}\right).\tan(\delta(t)).\Delta t \quad (7)$$

$$\delta(t) = K_p.e_\theta(t) + K_i.\int_0^t e_\theta(\tau)d\tau + K_d.\frac{de_\theta(t)}{dt} \quad (8)$$

$$u_v(t) = Kp_v.e_v(t) + Ki_v.\int_0^t e_v(\tau)d\tau + Kd_v.\frac{de(t)}{dt} \quad (9)$$

$$\acute{e}_\theta(t) = \theta_d - \theta_t \quad (10)$$

$$e_\theta(t) = a\tan 2\left(\sin\left(\acute{e}_\theta(t)\right), \cos\left(\acute{e}_\theta(t)\right)\right) \quad (11)$$

$$e_v(t) = \sqrt{(x_d - x)^2 + (y_d - y)^2} - dist_{ref} \quad (12)$$

The reference inputs, including sets of waypoints (2, 3, 4, 5, and 10), and paths (square, triangle, trapezoid, and half-circle row-end turning shapes), as shown in Fig. 2 were respectively used for the offline and online path following experiments to assess the effectiveness of the proposed PID controller. The waypoints for offline experiments were predefined based on known environmental coordinates. For online experiments, waypoints were continuously updated in real time to form dynamic paths, facilitating navigation through various geometries and turning patterns. The outputs of the implemented PID controllers in MATLAB were subsequently integrated with a dynamic model of the robot within the CoppeliaSim simulation environment

Table 2 PID Controller Pseudo code

Initialize PID gains: Kp, Ki, Kd
For each time step:
Calculate orientation error $e_\theta(t)$
Calculate speed error $e_v(t)$
Update steering control $\delta(t)$
Update speed control $u_v(t)$
Update robot position and orientation
End

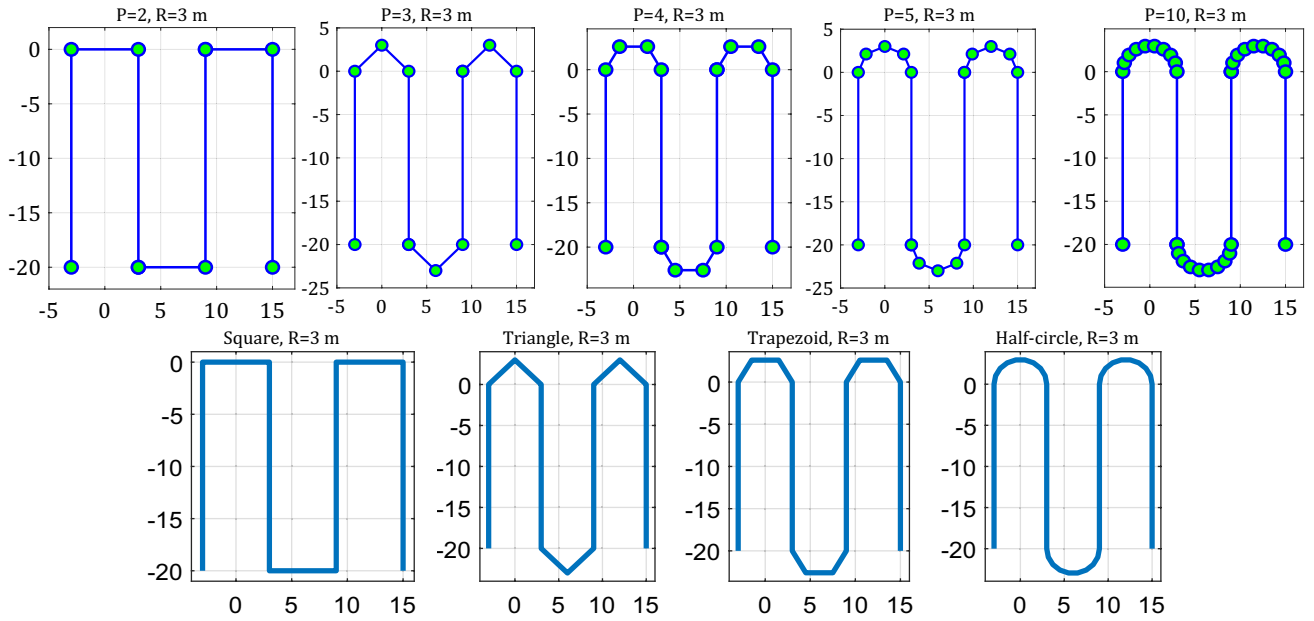

Fig. 2 Reference paths with different numbers of waypoints were used for the evaluation of the proposed PID controller, and for different row-end turning shapes, including square, triangle, trapezoid, and half-circle as shown below

using API functions as shown in Fig. 3, creating a versatile framework for visualization of the robot's behavior across various reference inputs and different terrains with uneven surfaces. It should be noted that for this study, experiments were exclusively conducted on flat terrains. For the simulation parameters in CoppeliaSim, the predefined time step of 0.05 s was used to ensure accurate results along with the Bullet Physics engine which is known for its robustness and good performance, and is suitable for general-purpose simulations. For the MATLAB Simulink, specific details like the step size and duration are embedded within the simulation loop, where the robot continuously updates its position, orientation, and control inputs at each time step (Δt) (Lines 200–210, Fig. 5 and Fig. 6).

To ensure the accuracy and reliability of the manually created waypoints and paths, several verification steps were implemented. The paths were first manually checked against the high-resolution orthoimage and 3D model to ensure they align accurately with the physical layout of the field, and cross-referencing was done using the RTK-GNSS measurements to validate the positions of key waypoints along the paths. The paths were then tested within the simulation environment to identify any potential issues such as collisions or navigation errors, and simulated runs were performed to ensure the paths are navigable by the robot, adjusting as

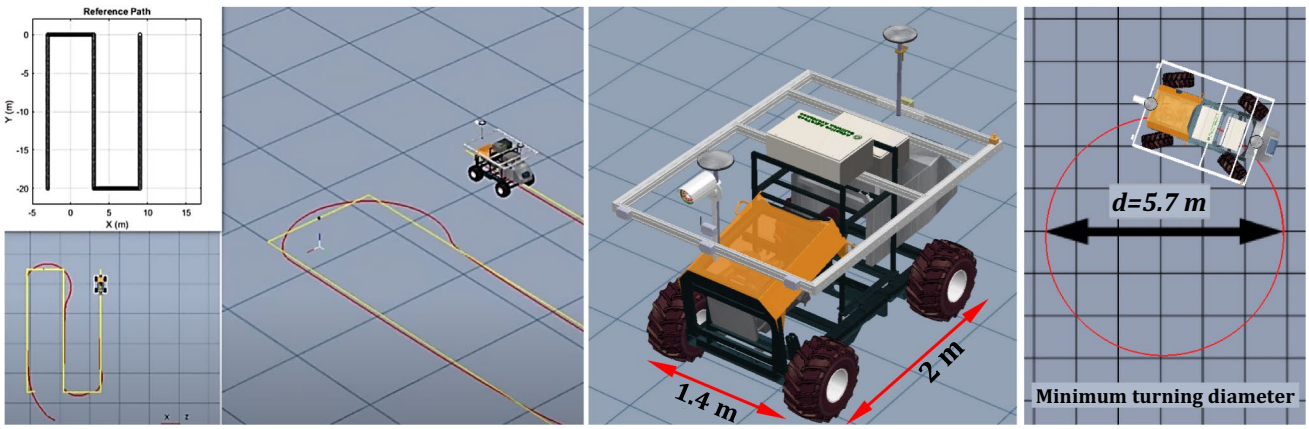

Fig. 3 The simulated robot in CoppeliaSim environment, interfaced with MATLAB for experimenting with the proposed controllers

necessary based on the results. An automated path validation algorithm was implemented in the simulation to check for common human errors, such as paths that pass through obstacles or exceed the operational limits of the robot, and redundant paths were created where necessary to provide alternative routes in case of unforeseen obstacles or errors in the primary path. A video demo of the path validation algorithm can be found here: https://youtu.be/IhjEKPQ7w88. Using these verification and failsafe mechanisms, we managed to maintain the integrity of the manually generated paths and minimize the risk of human error and ensuring reliable navigation through the berry rows.

2.3 Design of path tracking with learning-based H_∞ robust adaptive control

This section presents the details of the proposed learning-based H_∞ robust adaptive controller and its integration with the waypoint following algorithm that was specifically designed to improve the performance of the orchard robot navigation in the presence of uncertainties, disturbances, and time-varying dynamics. The objective was to increase the stability and performance bounds, even in the presence of nonlinearities and constraints. An overview of the controller is illustrated in Fig. 4, demonstrating its aim to achieve optimal tracking while maintaining stability by employing approximate dynamic programming techniques to solve the Hamilton–Jacobi-Isaacs (HJI) equation, balancing tracking error minimization and disturbance maximization in a zero-sum game formulation.

As shown in Fig. 4, a reference path with a half-circle row-end shape is first generated using a set of mathematical equations. Within the neural estimator block, a proper estimation of the HJI equation is then derived by incorporating inputs such as velocity, the difference between the reference path and the robot's position, and their derivatives. Based on this initial estimation and the system states, the adaptive robust controller (actor) generates the suitable velocity control signal for the robot to track the desired path. This control loop ultimately enables the robot to follow the reference path with minimal deviation, even in the presence of slip disturbances. (Eqs. 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30) offer a detailed, step-by-step guide for the control design procedure. Considering the nonlinear continuous-time affine system given in Eq. 13. Here $x = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^n$ denotes the state vector of the system, $u = [u_1, u_2, \ldots, u_m] \in \mathbb{R}^m$ is the control input vector, and $\beta = [\beta_1, \beta_2, \ldots, \beta_p] \in \mathbb{R}^p$ is the external disturbance vector. Furthermore, $f(x) \in \mathbb{R}^n$ represent the system dynamics, $g(x) \in \mathbb{R}^{n\times m}$ the input dynamics, and $k(x) \in \mathbb{R}^{n\times m}$ the disturbance dynamics. It is assumed that both u and β satisfy the condition $L_2 \in [0, \infty)$, where $L_2[0, \infty)$ denotes the L_2 norm on the interval $[0, \infty)$. Functions of $f(.)$, $g(.)$, and $h(.)$ are local Lipschitz continuous, ensuring the existence and uniqueness of solutions, and $f(0) = 0$, implying that the origin is an equilibrium point of the unforced system. The system given in Eq. 13 is presumed to be stable and robustly controllable within a compact set $\Omega \in \mathbb{R}^n$ of the state space (Lian et al. 2024). In order to find an optimal solution, the

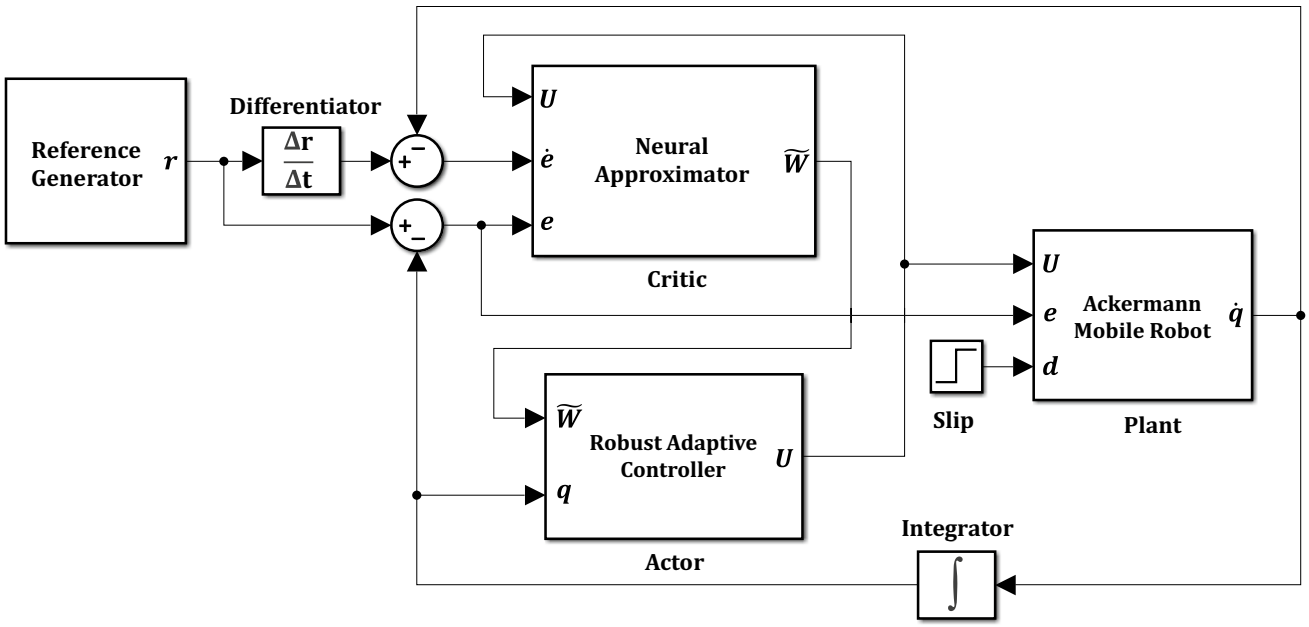

Fig. 4 Schematic block diagram of the H_∞ Robust Adaptive kinematic controller

cost function given in Eq. 14 (Zhao et al. 2022) was used to be minimized by the control input (u):

$$\dot{x} = f(x) + g(x)u + k(x)\beta \tag{13}$$

$$V(x(t)) = \int_{t}^{\infty} e^{-\gamma(\tau-t)}(x^T(\tau)Qx(\tau) + u^T(\tau)Ru(\tau) - {\gamma_d}^2(x))d\tau \tag{14}$$

The weighting matrices Q and R were chosen to be positive definite diagonal matrices. The positive constant of γ was also considered to guarantee the boundedness of the cost function with γ_d denoted as the upper bound of uncertainties (Zhao et al. 2022). Solving the H_∞ control problem bears a close relationship to the concept of zero-sum games within the differential game framework (Xue et al. 2022). Consequently, the task of synthesizing a robust controller can be formulated as the search for a saddle point that satisfies the Nash condition in the context of zero-sum games, expressed by Eq. 15 as described in "DIFFERENTIAL GAMES" (2012). Here $V^*(x)$ is the optimal value function. Differentiating $V(x(t))$ along system trajectories, yields the Bellman equation given in Eq. 16. The Hamiltonian is then defined by Eq. 17. Subsequently, through further partial differentiation steps described in Eq. 18 and Eq. 19, the optimal values of the control input and disturbance, representing the two players in the zero-sum game, were computed. Substituting Eq. 18 and Eq. 19 into Eq. 17 yields Eq. 20.

$$V^*(x) = J(x, u^*, \beta^*) = \min_u \max_d J(x, u, \beta) = \max_d \min_u J(x, u, \beta) \tag{15}$$

$$\frac{dv}{dx}\left[f(x) + g(x)u + k(x)\beta\right] + x^T Qx + u^T Ru - \gamma V(x) + {\gamma_d}^2(x) \tag{16}$$

$$\begin{aligned} H(x, V, u, \beta) = {} & \frac{dv}{dx}\left[f(x) + g(x)u + k(x)\beta\right] \\ & + x^T Qx + u^T Ru - \gamma V(x) + {\gamma_d}^2(x) \end{aligned} \tag{17}$$

$$u^* = \frac{\partial H}{\partial u} = 0 \rightarrow u^* = -\frac{1}{2}R^{-1}g(x)^T\frac{dV^*}{dx} \tag{18}$$

$$\beta^* = \frac{\partial H}{\partial d} = 0 \rightarrow \beta^* = \frac{1}{2\gamma^2}k(x)^T\frac{dV^*}{dx} \tag{19}$$

$$\begin{aligned} V(x) = {} & x^T Qx + \nabla {V^*}^T f(x) - \frac{1}{4}\nabla {V^*}^T g(x)R^{-1}g^T(x)\nabla V^* \\ & + \frac{1}{4\gamma}\nabla {V^*}^T k(x)k^T(x)\nabla V^* = 0 \end{aligned} \tag{20}$$

Here $\nabla V^*(x) = dV^*/dx$ denotes the gradient of the optimal value function V^*. Equation 20 represents an HJI partial differential equation. Obtaining an analytical solution to this nonlinear equation is generally intractable. In this regard, an Adaptive Dynamic Programming (ADP) algorithm was employed to approximate its solution in actor-critic structure. Following the approach proposed by Zhao et al. (2022) and (Zhao et al. 2020), a critic neural network was employed to estimate the value function ($\widehat{V}$) given in Eq. 21, where $\widehat{W} \in \mathbb{R}^l$ and $\psi(x) \in \mathbb{R}^l$ are estimated neural network weight and regressor, respectively. Substituting Eq. 21 into Eq. 18 and Eq. 19, yields the approximated signals of u and β, given in Eq. 22 and Eq. 23

$$\widehat{V}(x) = \widehat{W}^T\psi(x) \tag{21}$$

$$\widehat{u} = -\frac{1}{2}R^{-1}g(x)^T(\nabla\psi(x))^T\widehat{W} \tag{22}$$

$$\widehat{\beta} = \frac{1}{2\gamma^2}k(x)^T(\nabla\psi(x))^T\widehat{W} \tag{23}$$

For online updating of the neural network weights ($\dot{\widehat{W}}$), we employed an adaptive law based on a novel method outlined in Zhao et al. (2022). To this aim, Eq. 24 and Eq. 25 were used to arrive at the adjusted version of the HJI equation as given in Eq. 26, where W is an unknown parameter estimated using the adaptive law $\dot{\widehat{W}} = -\xi \mathrm{N}$. Here ξ represents the positive adaptive learning gain (Zhao et al. 2022) and $\mathrm{N} \in R^l$ is an auxiliary vector calculated as $\mathrm{N} = -\mathrm{P}\widetilde{W} + \rho$. According to Lv and Ren (2019) and (Luan et al. 2019) $\mathrm{P} \in R^{l\times l}$ and $\Upsilon \in R^l$ are filtered regressor matrices that are calculated using Eq. 27., and Eq. 28. We assumed that η is a positive constant and the initial values of P and Υ are equal to zero in the calculation of $\dot{\mathrm{P}} = -\eta\mathrm{P} + \Psi\Psi^T$ and $\dot{\Upsilon} = -\eta\Upsilon + \Psi\Phi$. It should be noted that $\widetilde{W} = W - \widehat{W}$ is the error of weight estimation and $\rho = \int_0^t e^{-\eta(t-\tau)}\varepsilon_{HJI}\Psi(\tau)d\tau$ is a bounded variable. The adaptive law can retain the convergence of the online learning algorithm by incorporating estimation errors, as described in Luan et al. (2019). Finally, the error-based signals of $e = q - q_r$, and $\dot{e} = S(q)U - \dot{q}_r$, were defined along with an analysis of the preliminaries of adaptive robust control. A comparative analysis between $\dot{e}$ and Eq. 13 reveals that $S(q)U = g(e)u$ and $\dot{q}_r = f(e)$, where the functions $g(e)$ and $f(e)$ represent the reformulated system dynamics expressed in terms of the error state e. With the system dynamics recast in the nonlinear affine formulation, Eq. 22 is used to produce the kinematic control input (u) as formulated in Eq. 29. The pseudo-code description of the controller is also provided in Table 3.

$$\Psi = -\gamma\varphi + \nabla\varphi(x)[f(x) + g(x)u] \tag{24}$$

$$\Phi = \gamma_d{}^2(x) + x^T Q x + u^T R u \tag{25}$$

$$\Phi = -W^T \Psi - \varepsilon_{HJI} \tag{26}$$

$$\mathrm{P} = \int_0^t e^{-\eta(t-\tau)} \Psi(\tau) \Psi^T(\tau) d\tau \tag{27}$$

$$\Upsilon = \int_0^t e^{-\eta(t-\tau)} \Psi(\tau) \Phi(\tau) d\tau \tag{28}$$

$$\hat{u} = -\frac{1}{2} R^{-1} S(q)^T (\nabla \psi(x))^T \hat{W} \tag{29}$$

To assess the proposed control scheme's efficacy and superiority, a numerical simulation study was carried out utilizing MATLAB-Simulink with the corresponding blocks shown in Fig. 5 and 6. During the time interval from 15 to 100 s, wheel slippage with the amplitude of $\beta_{amp} = 0.2$ occurred, characterized by the dynamic equation given in Eq. 14. The robot's initial posture was specified as $q = [0,0,\pi/2,0]^T$. $Q = I_4, R = 0.01 \times I_2$ and $\gamma = 0.1$ are constant parameters utilized in the value function defined in (Eq. 9). For the best adaption and robustness, other constants related to the adaptive learning method were assigned values of $\xi = 50$ and $\eta = 100$ using a trial-and-error process. Altering the matrices of R and its coefficients, as well as adjusting the initial values of W, enhances the robustness and efficacy of the controller. The regressor employed in the critic neural network is in Eq. 30. The critic neural network used a regressor involving the input signals' cross-product. This provided a nonlinear mapping of the error signals for the trajectory tracking of each state vector. The details of this can be found in the NN-Approximator block in Fig. 6, where the error signals e_1, e_2, e_3, and e_4 correspond to e_x, e_y, e_δ, and e_θ respectively. Also, the initial values of estimated critic NN weights are $\widehat{W} = 20 \times [5,1,1,1,1,1,5,1,5]$.

$$\psi = \left[e_1^2, e_1 e_2, e_1 e_3, e_1 e_4, e_2^2, e_2 e_3, e_2 e_4, e_3^2, e_3 e_4, e_4^2\right]^T \tag{30}$$

Table 3 Learning-based H_∞ Robust Adaptive Controller Pseudo code

Initialize HJI equation parameters and neural network weights
For each time step:
  Generate reference path
  Estimate HJI equation through neural estimator
  Update control signal using adaptive robust controller
  Apply control signal to robot
  Update robot position and orientation
End

3 Results and discussions

3.1 Performance of the waypoint-based path-following controller

The gains of the PID controller for waypoint-based path following were fine-tuned using a trial and error approach

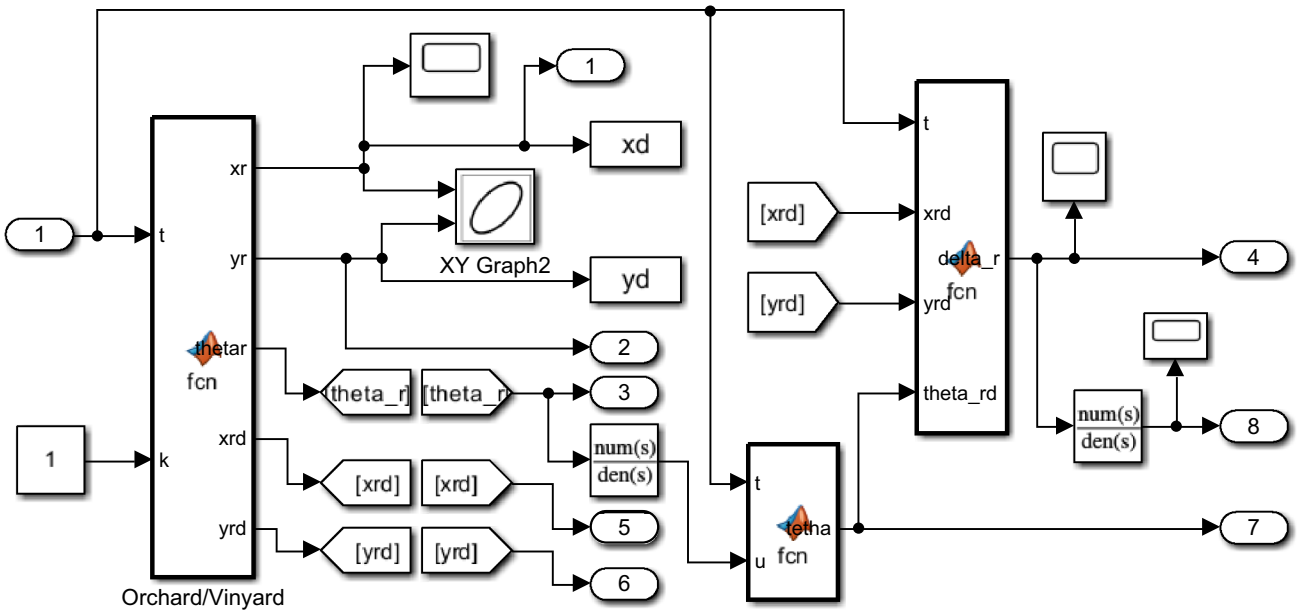

Fig. 5 Implementation of the reference trajectory path planner in MATLAB Simulink

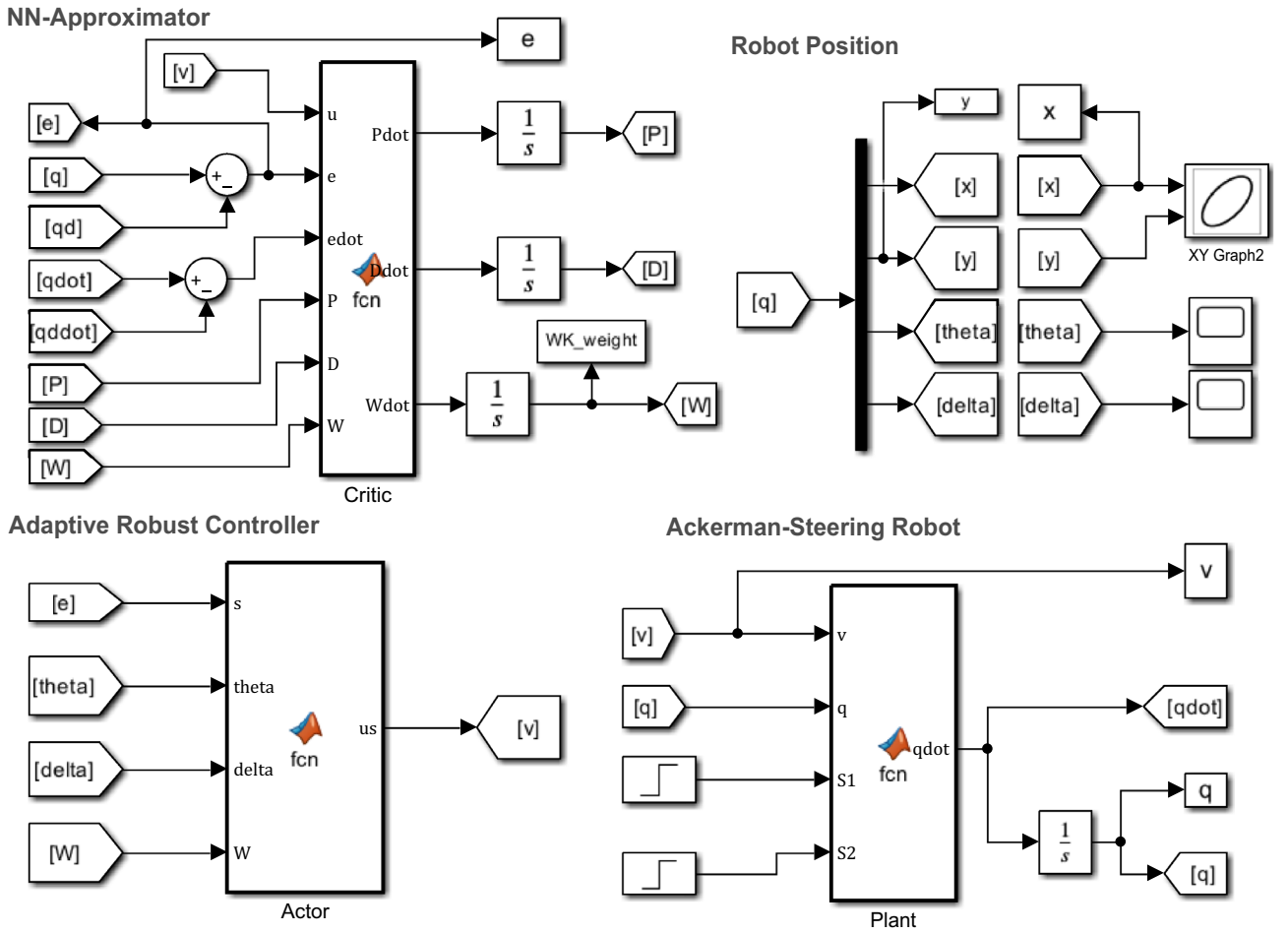

Fig. 6 Implementation of the Adaptive robust controller in MATLAB Simulink

within the simulation framework. After multiple iterations and adjustments, the PID controller achieved satisfactory performance for a row-end turning diameter of 6.0 m, an initial robot orientation (θ_0) of 90 degrees, and a maximum linear speed of 2.8 m/s, as shown in Fig. 7.a. Here, the robot traveled a path closely aligned with the reference input, demonstrating minimal tracking error and effective alignment with the desired waypoints. Screenshots of preliminary experiments with other scenarios resulting in unstable or unsatisfactory responses such as row-end turning diameter below 6.0 m, initial orientation of $\theta_0 = 0$, maximum linear speed of $V_{max} > 2.8$ m/s, or untuned PID gains are also provided in Fig. 7(b-e). It can be observed that when the PID gains were not tuned, slight deviations from the reference path could still be observed, depending on the diameter of the row-end turning or in scenarios with higher linear velocity of the robot and introduction of disturbances.

Results of the experiments for offline path following with PID controller using different predefined numbers of waypoints (P = 2, 3, 4, 5, and 10, labeled as (a) to (e) respectively) are provided in Fig. 8. This includes visualizations of the robot's motion, alongside plots of the robot's steering (δ) and orientation (θ). The primary objective of this experiment was to determine the minimum number of waypoints required by the controller to enable stable row-end turnings within a 6.0 m distance between two tracks. This minimum number of waypoints is particularly crucial not only for robot teleoperation from simulation environments, leveraging long-range wireless connectivity (i.e., using a LoRa network) to transmit target waypoints to the robot as explained in Shamshiri, et al., (2023), but also for autonomous modes of operation.

It can be observed that as the number of waypoints increases, the alignment between the robot's traveled path and the reference path improves noticeably, particularly with 4, 5, and 10 waypoints, where the controller demonstrates excellent performance, which is desired for navigating through narrow row-end turning spaces. This improvement

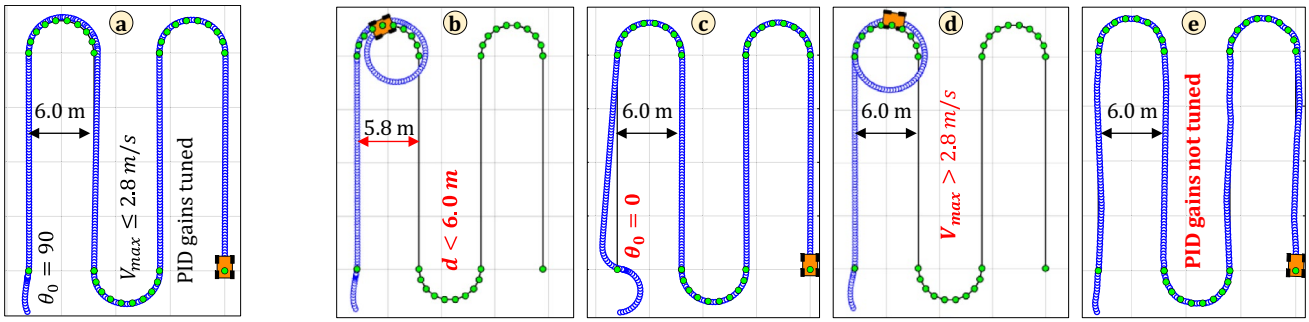

Fig. 7 Demonstration of the trial and error experiments for tuning the gains of the PID controller and visualization of the waypoint-based path following performance

is attributed to reduced tracking errors, enhanced curvature negotiation, and improved path coverage. With a higher density of waypoints, the robot receives more frequent and precise guidance points along the path, which allows for finer adjustments in steering commands and velocity control, resulting in smoother trajectory tracking and orientation adjustments as shown in Fig. 8. Moreover, as the number of waypoints at the row-end turning increases, the distance between consecutive waypoints decreases, leading to shorter segments of the path, which minimizes the potential for cumulative tracking errors, resulting in a more accurate reproduction of the reference path. This higher number of waypoints enables the robot to navigate through complex geometries and tight turns more effectively around curvature changes, hence the controller can anticipate trajectory adjustments earlier, and reduce the likelihood of overshooting or undershooting, contributing to smoother and more controlled maneuvering. The plots of steering angle shown in Fig. 8 represent the control signal sent to the robot to adjust its orientation along the path. It can be seen that as the number of waypoints increases from 2 to 10, the steering angle plot tends to exhibit smoother transitions and smaller fluctuations at the row-end turnings. This is also visible in the plot of robot orientation (smoother and more consistent orientation). The relationship between the robot's dimensions (wheelbase of 2 m, width of 1.4 m), maximum allowable steering angle (25 degrees), and the row-end turning radius emphasizes the importance of considering kinematic constraints in path planning and controller design. These results clearly show that while the proposed PID controller can successfully follow waypoints within certain bounds (specifically, a minimum of 4 waypoints), it encounters difficulties with maneuvers that exceed the robot's physical limitations, particularly turning radii below 3 m (as depicted in Fig. 7.b), or if the initial orientation of the robot is not properly aligned.

The results of online path following experiments with the PID controller for various row-end turning shapes are demonstrated in Fig. 9, showing plots of the robot's motion overlaid on the reference path, steering angle, linear speed, as well as the robot's lateral and longitudinal trajectories. By continuously receiving 1000 waypoints every 0.1 s, the robot navigates through paths of varying geometries, including square, triangle, trapezoid, and half-circle shapes, all with a fixed row-end turning radius of 3.0 m. It can be observed that the PID controller has demonstrated satisfactory performance in path following for half-circle and trapezoid shape row-end turnings, whereas the results are less favorable for square and triangle shapes. Here, this difference in performance is also attributed to the integral constraints imposed by the robot's kinematics and the geometry of the path. The gradual curvature of the half-circle shape facilitates smoother trajectory tracking, allowing the PID controller to make subtle steering adjustments within the constraints of the robot's steering mechanism. In contrast, the sharp corners of square and triangle shapes present challenges for the controller, requiring more abrupt changes in steering angle (as shown in the plots of steering angle) that may exceed the maximum allowable limits or result in suboptimal path tracking. In scenarios where the PID controller exhibits satisfactory performance, such as with a half-circle shape row-end turning, the steering angle plot typically shows smooth, gradual changes, indicative of precise steering adjustments within the allowable limits. On the other hand, in cases where the controller struggles, such as with square or triangle shapes, the steering angle plot may exhibit abrupt changes or oscillations, suggesting challenges in maintaining desired trajectory tracking. In successful scenarios, the speed plot at row-end turnings shows consistent velocity profiles, with smooth transitions between acceleration and deceleration phases. However, in cases where the controller faces difficulties, variations and fluctuations in the speed plot are observed, indicating challenges in maintaining optimal speed control, particularly during sharp turns or complex path geometries. Further analysis of the tracking paths in the X and Y directions provides evidence for the controller's

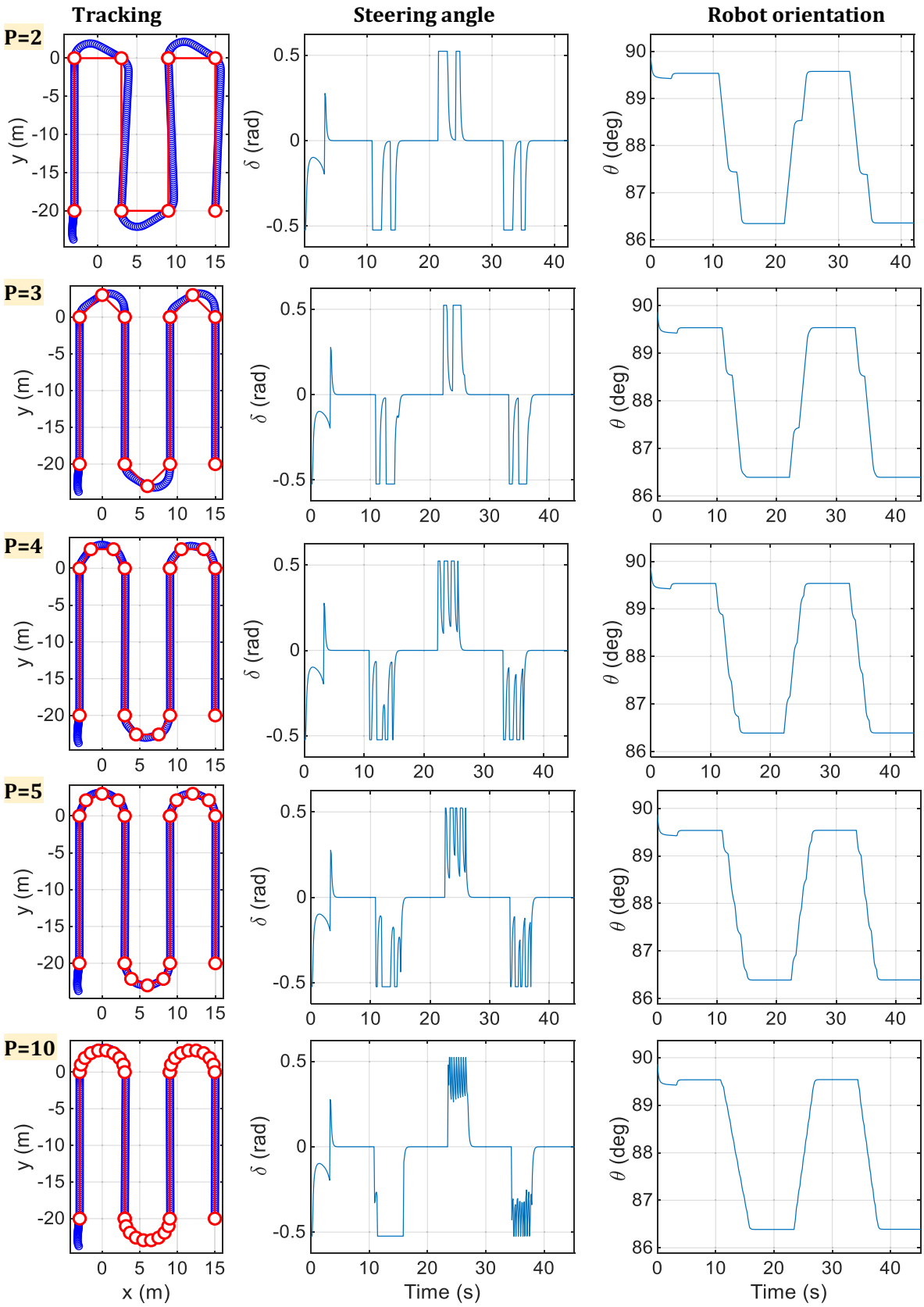

Fig. 8 Results of offline path following experiment with PID controller, showing plots of robot travelled path, control input signal (steering angle), and the robot response (orientation) with P = 2,3,4,5,10 number of waypoints and row-end radius of 3.0 m

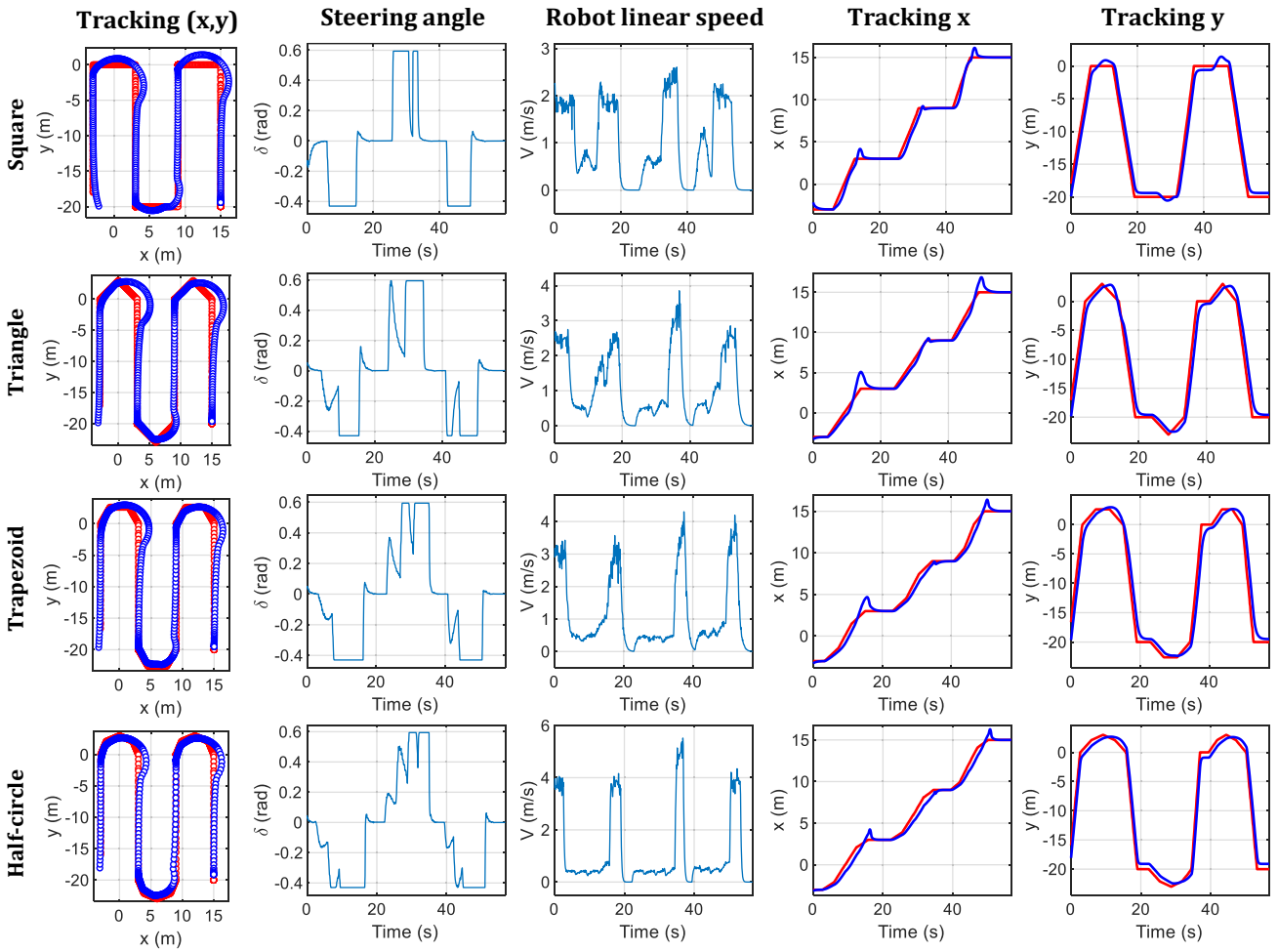

Fig. 9 Results of online path following experiment with PID controller, showing plots of robot travelled path, control input signal (steering angle), robot linear velocity, and robot tracking in X and Y for row-end turning patterns of square, triangle, trapezoid, and half-circle shapes

efficacy in adhering to the reference inputs, that is with trapezoid or half-circle shapes, the robot's motion in both the X and Y directions closely corresponds to the respective reference paths, showing reduced deviations and minimal overshoot or undershoot.

The analysis of the robot's traveled path in the X and Y directions compared to the reference data as shown in Fig. 10 is used for error analysis and insights into the accuracy of online path following experiments. For each path experiment, the coefficient of determination (R^2) was calculated to represent the variance in the robot traveled path from the reference path in order to assess the effectiveness of the PID controller in accurately following the desired trajectory. The observed linear relationship suggests that the PID controller successfully maintains a consistent trajectory along the reference path, with high R^2 values indicating strong linear correlations between the robot's actual path and the reference path in both the X and Y directions due to the controller's ability to adjust robot's steering, minimizing tracking errors and deviations from the desired trajectory. It can also be implied that the shape and curvature of the path also play a role in influencing tracking accuracy. Paths with more gradual curvature or smoother transitions were found to be indeed easier for the robot to follow accurately, resulting in higher R^2 values for the X data compared to the Y data. Interestingly, the highest R^2 value observed for the half-circle path indicates that the controller achieves the most accurate path tracking performance for this particular path geometry, suggesting that the gradual curvature of the half-circle shape facilitates smoother trajectory tracking, allowing the PID controller to make precise steering adjustments that closely align the robot's position with the reference path.

The inconsistency observed in R^2 values between the X and Y data can be attributed to several factors related to the robot's kinematics, path geometry, and control strategies.

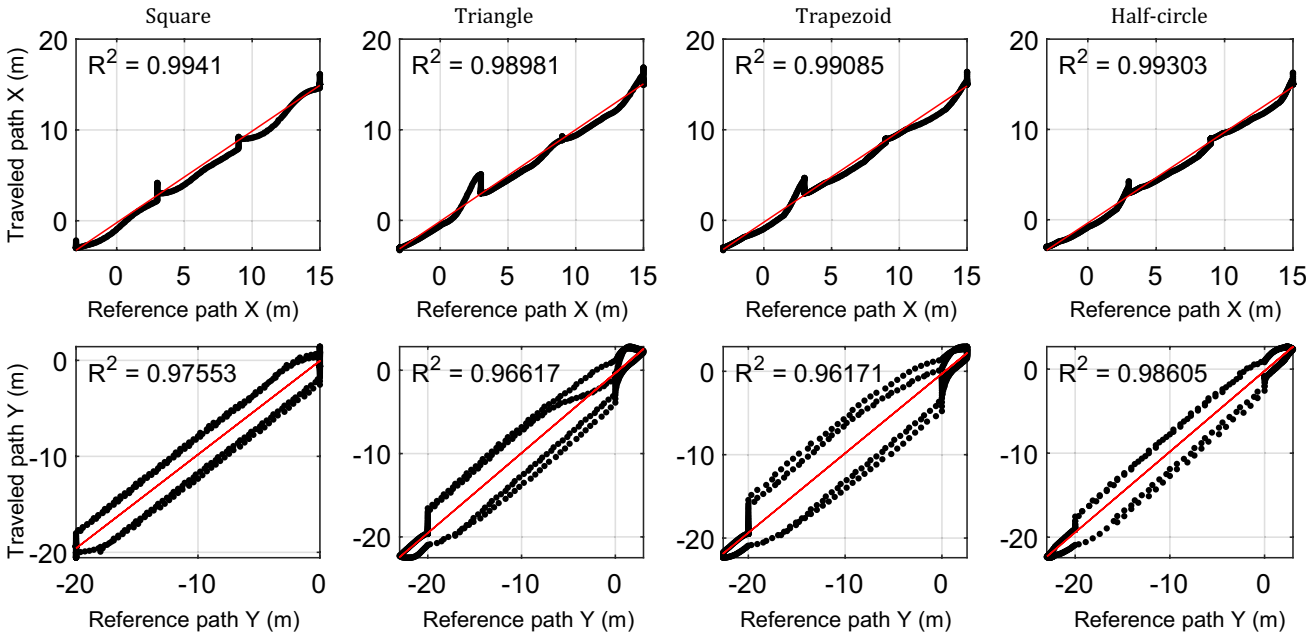

Fig. 10 Demonstration of path following accuracy through R-squared analysis for different row-end turning

The kinematics of the simulated robot, including its wheelbase, width, and steering joints influence its ability to maintain accuracy in different directions (here better tracking performance along X axis is observed). The unexpected observation where the R^2 value of the square path is slightly higher than that of the triangle path can be attributed to the predictability of the trajectory that differs between the square and triangle paths. Here the robot controller has encountered challenges in accurately following the trajectory of the triangle path due to its sharper corners and varying angles, leading to increased deviations from the reference path and a lower R^2 value. These findings highlight the importance of path geometry in online path following experiments, with certain shapes better suited to the constraints of the robot's kinematics. By selecting path shapes conducive to smoother trajectory tracking, such as trapezoid or half-circle configurations, users can enhance the performance of PID controllers in online path following tasks, particularly for row-end turnings with a radius of 3.0 m.

It should be noted that although an offline waypoint-based path following with the designed PID controller offers a practical solution for navigating through orchards with narrow row-end space, it does have inherent limitations. Despite its effectiveness in achieving satisfactory path tracking performance, this control method struggles with managing wheel slippage and other disturbances. The challenge of wheel slippage arises due to the dynamic nature of the robot's interaction with the terrain. As the robot maneuvers through orchard rows, variations in soil conditions or unexpected obstacles can cause wheels to slip, leading to deviations from the intended path. While the PID controller can adjust steering and velocity commands based on feedback signals (i.e., orientation error and distance to the next waypoint), it lacks the capability to directly address wheel slippage, making it challenging to maintain precise trajectory tracking in the presence of such disturbance. The achieved row-end turning radius of 3.0 m represents a practical limitation for the robot's maneuverability. With a wheelbase of 2.0 m and a width of 1.4 m, the robot's physical dimensions impose constraints on its ability to execute tight turns, particularly in narrow row-end spaces. Additionally, the maximum allowable steering angle of 25 degrees for the front wheels and – 25 degrees for the rear wheels further restricts the robot's maneuvering capabilities. Our simulation results with PID controller showed that below this threshold, the robot may become unstable, compromising its navigational accuracy and potentially leading to undesirable outcomes in real-world applications. These combined constraints highlight the challenges in achieving precise row-end turnings with the double-Ackermann steering robot, emphasizing the need for advanced control strategies to optimize path tracking within these limitations.

3.2 Evaluation of the H_∞ controller

The implementation of the H_∞ robust adaptive controller has significantly improved the robot's ability to follow a

predefined path by demonstrating better resistance against disturbances, allowing the robot to maintain stability and precision, especially during narrow row-end turnings. Plots of the simulation results with this controller, including trajectory tracking in the field, robot's position in an x–y direction, errors between the robot's position and the reference path are shown in Fig. 11. It can be observed that despite external wheel slippage disturbances, the robot maintains robust trajectory tracking while minimizing deviations from the reference path, which is required in orchard navigations with dynamic changes in the environment. Figure 11.a shows the trajectory tracking result of the robot for the half-circle shape row-end turning with a radius of 2.9 m. Here the orange and black curves respectively represent the reference and robot trajectories, and the green dots indicate the plants' lines. The trajectory segment where wheel slippage occurs is illustrated by the red line, with the start and end points denoted by circles. Comparing the robot's motion behavior along the x and y axes, as shown in Fig. 11.b and Fig. 11.c, respectively, reveals the remarkable superiority of this control method over the previous results from the PID controller. The absence of overshoots during row-end turns, even in the presence of slippage, indicates that the robot does not travel excessively along the x-axis. This characteristic is of significant practical value, particularly in applications like crop scouting or plant monitoring within orchard fields, ensuring stable and reliable data collection. Upon examining the plots displayed in Fig. 11b–e, it is evident that the proposed controller not only exhibits robustness against external disturbances but also displays optimal performance in both lateral and longitudinal directions which is attributed to benefiting from the power of dynamic programming and an intelligent controller.

Further analysis of the error plots (Fig. 11d, e) indicates that the robot has accurately tracked the reference trajectory with high precision, however some minor deviations are observed, especially around row-end turnings. These errors become obvious at the simulation time of t = 15 s, as the robot starts to slip, leading to deviation, and show their peaks between t = 80 and 110 s. Nevertheless, due to the superiority of the H_∞ controller over the PID, the robot can handle this disturbance and minimize its location to the reference path in about 25 s.

Several results are provided regarding the robot's linear velocity, steering angle, and heading angle, along with their respective errors compared to the corresponding reference values as shown in Fig. 12.a-e. As can be seen in Fig. 12.a, the robot's linear velocity remained relatively constant throughout the entire simulation, with some variations observed during the robot's state transitions, such as exiting straight-line traversal and initiating turns to enter the subsequent lane. This behavior enabled the robot to follow the predefined circular path at the end of the trajectory with appropriate performance. This phenomenon is supported by Fig. 12.b and Fig. 12.c, which demonstrate that the variations in the steering angle and heading angle, despite their continuity (in contrast to the observations with the PID controller shown in Figs. 8 and Figs. 9), assisted the robot's path tracking.

Another positive aspect derived from these two plots is the superiority of the path tracking approach over waypoint tracking, as evidenced by the smaller variations in the robot angle and steering angle over a limited time interval. A closer examination of their corresponding error profiles (Fig. 12d and Fig. 12e) further highlights the high accuracy of the proposed robust controller compared to the

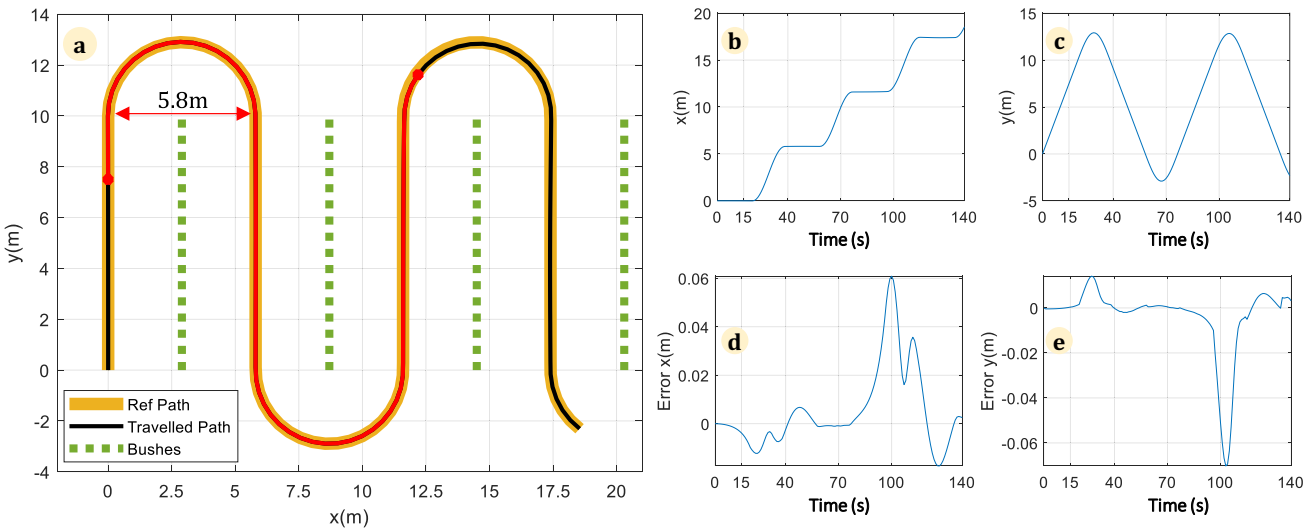

Fig. 11 Results of the H-inf controller, showing **a** reference trajectory tracking in the orchard with the half-circle shape row-end, **b**, **c** robot movement in X–Y axis, and **d**, **e** trajectory tracking error profiles

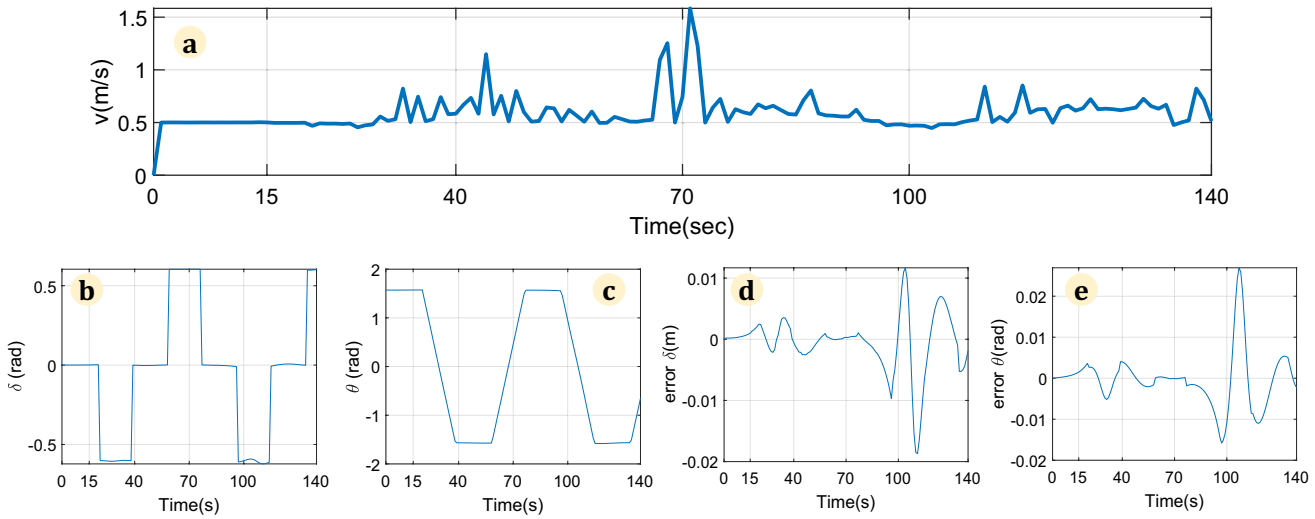

Fig. 12 Results of the H_∞ controller, showing **a** linear velocity of the robot, **b** steering angle, **c** heading angle, **d** steering angle error profile, and **e** heading angle error profile

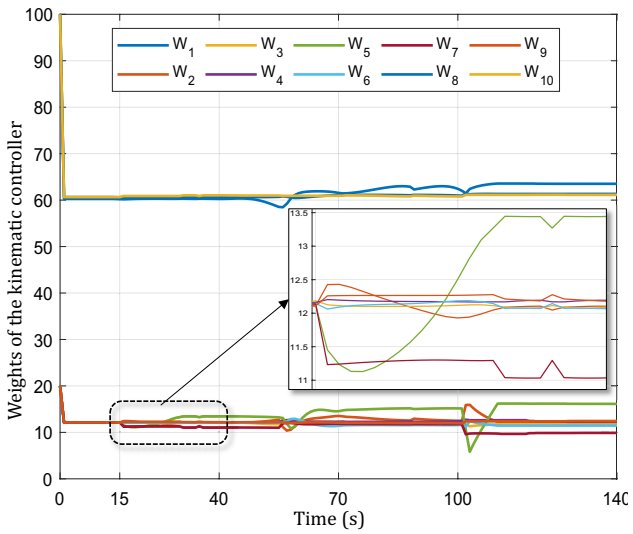

Fig. 13 Weights adaption and convergence during the simulation

PID method. As illustrated, the error profiles of these two parameters are in the order of hundredths, indicating the controller's precision. In the illustration provided by Fig. 13, the process of learning and adaptation of the controller's weights during the simulation is depicted. The weights tend to converge towards their optimal values, specifically (63.5, 12.3, 11.7, 12.3, 16.1, 11.4, 9.9, 61.3, 12.4, 61.1). This weight adaptation process serves to demonstrate the controller's capacity to adjust its parameters in response to the disturbance caused by wheel slippage, ultimately achieving convergence to the appropriate values for stable trajectory tracking performance.

It should be noted that while the robot platform offers the advantage of having a favorable wheelbase-to-width ratio for agricultural fields, which significantly enhances its stability and reduces the risk of tipping over particularly when navigating uneven terrain or executing sharp turns, it also exhibits certain limitations in maneuverability and constraints. Figure 14 summarized the results of further simulation experiments with half-circle row-end turnings, demonstrating instability in path following for both PID and H_∞ controllers when the row-end turning radius approaches the minimum turning radius of the robot, set at 2.85 m. Despite the nominal difference in achievable turning radii between the controllers, both exhibit instability at this critical threshold. This is primarily attributed to the kinematic limitations and mechanical constraints of the robot's steering system coupled with the controllers' inability to effectively compensate for these constraints, resulting in erratic behavior during path following. As explained earlier in Eq. 1, the minimum achievable turning radius for this platform is 2.84 m.

When considering the selection of mobile platforms for agriculture, it is important to consider the choice of wheelbase and width with the desired control methods for path following, especially when a swarm of robots is used for mechanized operation (Albiero et al. 2022). Ultimately, the optimal choice depends on balancing these factors to successfully meet the specific demands and constraints of agricultural tasks.

We compared the PID and H_∞ robust adaptive controllers based on several criteria including (i) tracking error to determine the deviation from the reference path, (ii) disturbance rejection to highlight the ability of the controller to maintain path following under disturbances like wheel slippage, (iii)

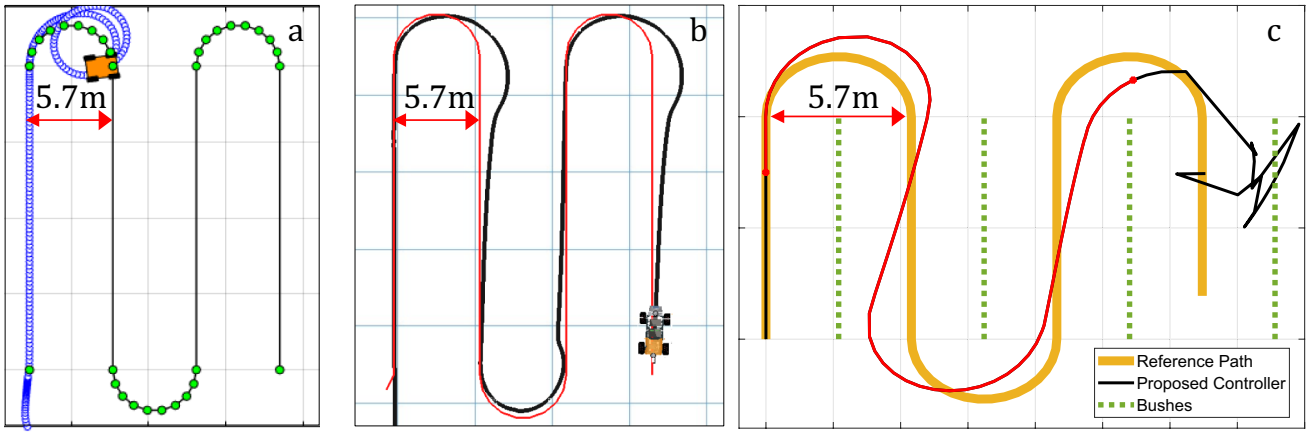

Fig. 14 The row-end turning radius R = 2.7 that all controllers were found to be unstable, **a** PID offline path following, **b** PID online path following, and **c** H-inf path following

minimum achievable turning radius to find the smallest turning radius where each controller maintained stability, and (iv) path-following accuracy to by measuring the coefficient of determination (R^2) between the traveled path and the reference path. For this purpose, we used two sets of MATLAB functions and Simulink blocks, including PID controller functions (like pid, sim, and ode45), as well as H∞ controller custom functions within MATLAB Function blocks for adaptive dynamic programming and neural network estimations. Our findings indicate that while the PID controller is simple and versatile, it struggles to adapt to environmental disturbances such as wheel slippage. On the other hand, the H∞ robust adaptive controller has demonstrated the ability to generate nearly optimal real-time control signals to effectively counteract these disturbances. This controller outperformed the PID controller in all aspects, particularly in maneuvering with a smaller row-end diameter. It maintained stability and precision during narrow row-end turns and online path tracking. The PID controller, known for its simplicity, has lower computational demands. However, the H∞ robust adaptive controller, despite being more computationally intensive due to the need for finding initial weights and design matrices, demonstrated superior real-time performance in terms of tracking, particularly under disturbances like wheel slippage. Overall, the combination of the steering mechanism and path tracking controller must be carefully chosen to optimize the robot's performance, balancing factors such as accuracy, stability, complexity, and cost. Our simulation-based results from the comparative study between PID and H_∞ controllers highlight the effectiveness of the proposed simulation framework built with MATLAB and CoppeliaSim for accelerating the evaluation and fine-tuning of controllers for the specific robot under study. This approach can be extended to serve as a tool in the iterative design process for other mobile platforms, offering the opportunity to experiment with diverse design choices, including robot platforms, steering systems, and controllers for their maneuverability and stability. This virtual investigation significantly reduced the time and cost constraints associated with physical prototyping and afforded insights into the complicated dynamics of robotic systems across varying operating conditions and scenarios. It can contribute to enhancing the autonomy and efficiency of orchard navigation and provides valuable insights into the optimal design and control parameters necessary for the successful deployment of agricultural robots in real-world scenarios.

4 Conclusion

This study presented a comprehensive analysis of offline and online path following for a double-Ackermann steering robot in orchard navigation, using PID and H_∞ robust adaptive control methods. The proposed approach contributes to the improvement of navigation accuracy during row-end turnings with various turning patterns, particularly in scenarios where space at the row-ends is limited. The PID controller demonstrated its ability to regulate steering angle and speed, enhancing navigation between waypoints. Increasing waypoint density improved trajectory tracking and alignment with the reference path, minimizing tracking errors and enhancing navigation precision. Conversely, the H_∞ robust adaptive controller utilized adaptive dynamic programming to optimize trajectory tracking tasks, exhibiting promise in enhancing maneuverability and robustness, particularly in the presence of wheel slippage disturbances. Through comparative experiments, the PID controller demonstrated satisfactory performance, achieving a minimum

row-end turning radius of 3.0 m, although with limitations in managing wheel slippage and tight turns. In contrast, the H_∞ controller surpassed the PID, achieving a minimum radius of 2.9 m while maintaining stability and precision, particularly in online path tracking scenarios. Additionally, the study highlighted the significance of balancing factors such as accuracy, stability, complexity, and cost in control method selection. The research contributes valuable insights into agricultural mobile platform design and optimization, leveraging simulation frameworks for virtual evaluation of diverse design choices, accelerating the design and deployment of path tracking systems. Ultimately, these findings contribute to advancing the autonomy and efficiency of orchard navigation, laying a foundation for the successful integration of agricultural robots into real-world scenarios. Future research may aim to compare the performance of other advanced control methods, such as MPC and SMC, across different robot platforms and steering mechanisms, addressing various environmental disturbances and changes in the parameters that affected controller performance.

Acknowledgements We gratefully acknowledge the technical support provided by Keywan Majidi and Eduardo Navas Merlo at Adaptive AgroTech Consultancy.

Author contributions R.SH. wrote the main manuscript and established the conceptualization, storyline, problem definition, and most parts of the simulation and analysis and results. The 2nd author A.A contributed significantly to the H-inf controller section, and the response letter. The 3rd and the 4th author, M.B. and A.GH prepared some parts of the methodology and introduction and contributed to some of the figures. Also V.D, and C.W, and K.K and F.A.C contributed to the manuscript review, editing, revision, corrections, and results. All authors reviewed the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

References

AbdElmoniem, A., Osama, A., Abdelaziz, M., Maged, S.A.: A path-tracking algorithm using predictive Stanley lateral controller. Int. J. Adv. Rob. Syst. **17**(6), 1729881420974852 (2020)

Albiero, D., Pontin Garcia, A., Kiyoshi Umezu, C., Leme de Paulo, R.: Swarm robots in mechanized agricultural operations: a review about challenges for research. Comput. Electron. Agric. **193**, 106608 (2022)

Amertet, S., Gebresenbet, G., Alwan, H.M.: Optimizing the performance of a wheeled mobile robots for use in agriculture using a linear-quadratic regulator. Robot. Auton. Syst. **174**, 104642 (2024)

An, Y., Wang, L., Deng, X., Chen, H., Lu, Z., Wang, T.: Research on differential steering dynamics control of four-wheel independent drive electric tractor. Agriculture **13**(9), 1758 (2023)

Aqel, M.O.A., Marhaban, M.H., Saripan, M.I., Ismail, N.B.: Review of visual odometry: types, approaches, challenges, and applications. Springerplus **5**(1), 1897 (2016)

Asadi, K., et al.: An integrated UGV-UAV system for construction site data collection. Autom. Constr. **112**, 103068 (2020)

Bai, J., Du, J., Li, T., Chen, Y.: Trajectory tracking control for wheeled mobile robots with kinematic parameter uncertainty. Int. J. Control. Autom. Syst. **20**(5), 1632–1639 (2022)

Blok, P.M., van Boheemen, K., van Evert, F.K., IJsselmuiden, I., Kim, G.H.: Robot navigation in orchards with localization based on particle filter and kalman filter. Comput. Electron. Agric. **157**, 261–269 (2019)

M. Campbell, K. Ye, E. Scudiero, and K. Karydis, "A portable agricultural robot for continuous apparent soil electrical conductivity measurements to improve irrigation practices," In: 2021 IEEE 17th international conference on automation science and engineering (CASE), 2021, pp. 2228–2234

Chang, L., Shan, L., Jiang, C., Dai, Y.: Reinforcement based mobile robot path planning with improved dynamic window approach in unknown environment. Auton. Robot. **45**(1), 51–76 (2021)

Chen, C., Zhu, H., Li, M., You, S.: A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives. Robotics **7**(3), 45 (2018)

Chen, W., Xu, T., Liu, J., Wang, M., Zhao, D.: Picking robot visual servo control based on modified fuzzy neural network sliding mode algorithms. Electronics **8**(6), 605 (2019)

H. Chen, X. Wang, L. Zhao, R. Jiang, and B. Zhumadil, "Research on path tracking control of mobile storage robot based on model predictive control and linear quadratic regulator," In: Proc. SPIE, 2023, vol. 12748, p. 127482O

Corke, P., Jachimczyk, W., Pillat, R.: Mobile robot vehicles bt - robotics, vision and control: fundamental algorithms in MATLAB®, pp. 127–160. Springer International Publishing, Cham (2023)

"Differential games," In: Optimal control, 438–460 (2012)

Deniz, N., Jorquera, F., Torres-Torriti, M., Cheein, F.A.: Model predictive path-following controller for Generalised N-Trailer vehicles with noisy sensors and disturbances. Control. Eng. Pract. **142**, 105747 (2024)

Farooq, M.U., Eizad, A., Bae, H.-K.: Power solutions for autonomous mobile robots: a survey. Robot. Auton. Syst. **159**, 104285 (2023)

M. Fnadi, F. Plumet, and F. Benamar, "Model Predictive control based dynamic path tracking of a four-wheel steering mobile robot," In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS), 2019, pp. 4518–4523

Fragapane, G., de Koster, R., Sgarbossa, F., Strandhagen, J.O.: Planning and control of autonomous mobile robots for intralogistics: literature review and research agenda. Eur. J. Oper. Res. **294**(2), 405–426 (2021)

Greenberg, J.N., Tan, X.: Dynamic optical localization of a mobile robot using kalman filtering-based position prediction. IEEE/ASME Trans. Mechatron. **25**(5), 2483–2492 (2020)

Guevara, L., Jorquera, F., Walas, K., Auat-Cheein, F.: Robust control strategy for generalized N-trailer vehicles based on a dual-stage disturbance observer. Control. Eng. Pract. **131**, 105382 (2023)

Gyagenda, N., Hatilima, J.V., Roth, H., Zhmud, V.: A review of GNSS-independent UAV navigation techniques. Robot. Auton. Syst. **152**, 104069 (2022)

He, J., et al.: Path tracking control method and performance test based on agricultural machinery pose correction. Comput. Electron. Agric. **200**, 107185 (2022)

Huang, P., Zhang, Z., Luo, X., Zhang, J., Huang, P.: Path tracking control of a differential-drive tracked robot based on look-ahead distance. IFAC-PapersOnLine **51**(17), 112–117 (2018)

Kan, X., Teng, H., Karydis, K.: Online exploration and coverage planning in unknown obstacle-cluttered environments. IEEE Robot. Autom. Lett. **5**(4), 5969–5976 (2020)

Kan, X., Thayer, T.C., Carpin, S., Karydis, K.: Task planning on stochastic aisle graphs for precision agriculture. IEEE Robot. Autom. Lett. **6**(2), 3287–3294 (2021)

Khalaji, A.K., Moosavian, S.A.A.: Robust adaptive controller for a tractor-trailer mobile robot. IEEE/ASME Trans. Mechatron. **19**(3), 943–953 (2014)

Kivrak, H., Cakmak, F., Kose, H., Yavuz, S.: Waypoint based path planner for socially aware robot navigation. Clust. Comput. **25**(3), 1665–1675 (2022)

Li, X., et al.: Review of PPP–RTK: achievements, challenges, and opportunities. Satell Navig **3**(1), 28 (2022)

Lian, B., Xue, W., Lewis, F.L., Modares, H., Kiumarsi, B.: Integral reinforcement learning for zero-sum games BT - integral and inverse reinforcement learning for optimal control systems and games, pp. 109–147. Springer Nature Switzerland, Cham (2024)

Luan, F., Na, J., Huang, Y., Gao, G.: Adaptive neural network control for robotic manipulators with guaranteed finite-time convergence. Neurocomputing **337**, 153–164 (2019)

Lv, Y., Ren, X.: Approximate nash solutions for multiplayer mixed-zero-sum game with reinforcement learning. IEEE Trans. Syst., Man, Cybernetics: Syst. **49**(12), 2739–2750 (2019)

Majdoubi, R., Masmoudi, L., Elharif, A.: Coupled nonlinear controller for vehicle trajectory tracking in a deformable soil: application to a four-wheeled mobile agricultural robot. J. Terrramech. **110**, 47–68 (2023)

Meshram, A.T., Vanalkar, A.V., Kalambe, K.B., Badar, A.M.: Pesticide spraying robot for precision agriculture: a categorical literature review and future trends. J. Field Robot. **39**(2), 153–171 (2022)

A. J. Prado, D. Chávez, O. Camacho, M. Torres-Torriti, and F. A. Cheein, "Adaptive Nonlinear MPC for Efficient Trajectory Tracking Applied to Autonomous Mining Skid-Steer Mobile Robots," In: 2020 IEEE ANDESCON, 2020, pp. 1–6

Raj, T., Hashim, F.H., Huddin, A.B., Ibrahim, M.F., Hussain, A.: A survey on LiDAR scanning mechanisms. Electronics **9**(5), 741 (2020)

Ryu, J.H., Irfan, M., Reyaz, A.: A review on sensor network issues and robotics. J. Sensors **2015**, 140217 (2015)

Samuel, M., Hussein, M., Mohamad, M.B.: A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle. Int. J. Comput. Appl. **135**(1), 35–38 (2016)

C. Shamshiri, R.; Navas, E.; Dworak, V.; Fernández, R.; Schütte, T.; Weltzien, "Teleoperation of an agricultural mobile robot inside berry orchard using lora connectivity," In: 80th international conference on agricultural engineering. LAND.TECHNIK AgEng, 191–197(2023)

Tiozzo Fasiolo, D., Scalera, L., Maset, E., Gasparetto, A.: Towards autonomous mapping in agriculture: a review of supportive technologies for ground robotics. Robotics Auton. Syst. **169**, 104514 (2023)

Tu, X., Gai, J., Tang, L.: Robust navigation control of a 4WD/4WS agricultural robotic vehicle. Comput. Electron. Agric. **164**, 104892 (2019)

Virlet, N., Sabermanesh, K., Sadeghi-Tehran, P., Hawkesford, M.J.: Field Scanalyzer: an automated robotic field phenotyping platform for detailed crop monitoring. Funct. Plant Biol. **44**(1), 143–153 (2017)

Vulpi, F., Milella, A., Marani, R., Reina, G.: Recurrent and convolutional neural networks for deep terrain classification by autonomous robots. J. Terrramech. **96**, 119–131 (2021)

Wang, Y., Li, X., Zhang, J., Li, S., Xu, Z., Zhou, X.: Review of wheeled mobile robot collision avoidance under unknown environment. Sci. Prog. **104**(3), 00368504211037771 (2021)

Weber, A.M., Gambao, E., Brunete, A.: A survey on autonomous offline path generation for robot-assisted spraying applications. Actuators **12**(11), 403 (2023)

Wen Zhu, C., Hill, E., Biglarbegian, M., Andrew Gadsden, S., Cline, J.A.: Smart agriculture: development of a skid-steer autonomous robot with advanced model predictive controllers. Robot. Auton. Syst **162**, 104364 (2023). https://doi.org/10.1016/j.robot.2023.104364

Wijayathunga, L., Rassau, A., Chai, D.: Challenges and solutions for autonomous ground robot scene understanding and navigation in unstructured outdoor environments: a review. Appl. Sci. **13**(17), 9877 (2023)

L. Xu, Y. Yang, Q. Chen, F. Fu, B. Yang, and L. Yao, "Path Tracking of a 4WIS–4WID Agricultural Machinery Based on Variable Look-Ahead Distance," *Applied Sciences*, vol. 12, no. 17. 2022.

Xue, S., Luo, B., Liu, D., Gao, Y.: Neural network-based event-triggered integral reinforcement learning for constrained H∞ tracking control with experience replay. Neurocomputing **513**, 25–35 (2022)

Yan, J., Zhang, W., Liu, Y., Pan, W., Hou, X., Liu, Z.: Autonomous trajectory tracking control method for an agricultural robotic vehicle. Int. J. Agric. Biol. Eng. **17**(1), 215–224 (2024)

Zhao, J., Na, J., Gao, G.: Adaptive dynamic programming based robust control of nonlinear systems with unmatched uncertainties. Neurocomputing **395**, 56–65 (2020)

Zhao, J., Na, J., Gao, G.: Robust tracking control of uncertain nonlinear systems with adaptive dynamic programming. Neurocomputing **471**, 21–30 (2022)

Zhou, B., Su, X., Yu, H., Guo, W., Zhang, Q.: Research on path tracking of articulated steering tractor based on modified model predictive control. Agriculture **13**(4), 871 (2023)

Redmond R. Shamshiri is a senior scientist at the Leibniz-Institut für Agrartechnik und Bioökonomie, working on digitizing agriculture to enhance food security. He holds a Ph.D. in Automation, specializing in control systems and dynamics, as well as a Doctor of Engineering in Biosystems Engineering. His research areas include autonomous navigation of agricultural mobile robots, sensor fusion for collision avoidance, teleoperation, wireless systems, and the optimization of controlled environments. With over 150 publications to his name, he has been recognized among the top 2% of scientists worldwide.

Alireza Azimi received his M.S. degree in Control Engineering from Kharazmi University in 2024. He is currently working as an R&D Engineer at Adaptive AgroTech, focusing on autonomous navigation and dynamic simulation of agricultural vehicles. His research interests include the design of robust and adaptive nonlinear controllers for mobile robots.

Maryam Behjati earned both her B.Sc. and M.Sc. degrees in Mechanical Engineering, specializing in Agricultural Machinery, from the University of Tehran. She currently works as a Robotics Engineer at Adaptive AgroTech, where she focuses on 3D modeling and simulation to explore various steering mechanisms for agricultural machinery on uneven terrains, with an emphasis on stability, control performance, and maneuverability.

Aliakbar Ghasemzadeh received his B.E. degree from Shahed University, Iran, in 2021, and his M.Sc. degree from Kharazmi University, Iran, in 2024, both in electrical and control enegineering. His research interests include learning-based control, nonlinear MIMO control, and robotics.

Volker Dworak is senior scientist in the ATB Department "Agromechatronics" and head of the working group "Automation and field robotics" since 2023. After finishing the professional training as an "electro technical assistance", he studied electrical engineering with the focus on microelectronics at the "Bergischen Universität Wuppertal". During the advanced study period he was granted by the "Prof. Dr. Koeppchen Stiftung". He finished his diploma thesis with three months of work experience at TopoMetrix, Santa Clara, CA, USA, and received a prize from the VDE and won the third prize for the best work at the university from the "Gesellschaft der Freunde der Bergischen Universität". After working as a researcher in the working Grupp of Prof. Dr. W. Mathis, he moved the experimental physics department of the "Universität des Saarlandes" and developed a scanning SQUID microscope as a new kind of scanning probe microscope. Then he became a project manager at the company "AMS Angewandte Magnetosensorik mbH" in Saarbrücken, and developed different kinds of magnetic and electromagnetic nondestructive measurement systems for industry products. After this, he changed to the Triple-O Microscopy GmbH as the head of the development and production department. Going back to research, he worked at the "Bundesanstalt für Materialforschung und -prüfung (BAM), Berlin" and developed a new kind of scanning force microscope for temperature-controlled measurements on polymer probes. He reached his doctoral degree with summa cum laude, and started as a researcher the ATB in December 2008. Since then, he worked on LiDAR scanner, camera design, sensor fusion, THz-scanner, and robotics for agriculture.

Cornelia Weltzien is head of ATB Department "Agromechatronics" and chair of Agro Mechatronics at the TU Berlin since 2015. After studying agricultural engineering at the University of Applied Sciences Cologne she completed her postgraduate studies in mechanical engineering at Technical University Braunschweig in 2006. In 2008 Prof. Dr.-Ing. Weltzien obtains her doctorate at the Institute for Agricultural Machinery and Fluid Technology with the thesis "Assistance System for the Overloading Process with a selfpropelled Forage Harvester". Before joining ATB she was part of interdisciplinary advanced engineering teams in the industry, developing electro hydraulic control systems for mobile working machinery. She aims to develop technical innovations to enable the diversity of crop production. With a main research interest in the field of agromechatronics and digital transformation of agriculture. This includes sensor application for knowledge based crop production and automation of agricultural machines and processes. Prof. Dr.-Ing. Weltzien is a nationally and internationally recognized elite researcher in the field of digital agriculture..

Konstantinos Karydis is currently an Associate Professor in the Department of Electrical and Computer Engineering at the University of California, Riverside (UCR), with cooperating faculty appointments in the Departments of Mechanical Engineering, and Computer Science and Engineering at UCR. Dr. Karydis is a recipient of a prestigious NSF CAREER Award in 2021, focusing on resilient autonomy of legged robots operating in agricultural fields. His research program addresses foundational robotics research problems underlying applications in precision agriculture, environmental monitoring, and human-robot collaboration. His research seeks to enable diverse existing and new robot embodiments (and teams thereof) to operate in efficient and resilient manners autonomously and/or in cooperation with humans despite the presence of uncertainty associated with action, perception, and the operating environment.

Fernando A. Auat Cheein is a Professor of Engineering at Harper Adams University, England, UK. He is associate editor of Computers and Electronics in Agriculture, Biosystems Engineering, IEEE OJ on Intelligent Transportation Systems, Journal of Field Robotics, Robotica-Cambridge, technical editor of IEEE Transactions on Mechatronics and IEEE senior member. He has published more than 100 journal articles and served as part of scientific board of numourous conferences. His research interests cover robotics and perction in agriculture, sustainability, motion control, and electrically powered vehicles.

Authors and Affiliations

Redmond R. Shamshiri[1,2] · Alireza Azimi[3] · Maryam Behjati[1] · Aliakbar Ghasemzadeh[3] · Volker Dworak[1] · Cornelia Weltzien[1,2] · Konstantinos Karydis[4] · Fernando A. Auat Cheein[5]

✉ Redmond R. Shamshiri
rshamshiri@atb-potsdam.de

Alireza Azimi
Alireza@AdaptiveAgroTech.com

Maryam Behjati
Maryam@AdaptiveAgroTech.com

Aliakbar Ghasemzadeh
aliakbar7@khu.ac.ir

Volker Dworak
vdworak@atb-potsdam.de

Cornelia Weltzien
CWeltzien@atb-potsdam.de

Konstantinos Karydis
Karydis@ucr.edu

Fernando A. Auat Cheein
F.Auat@hw.ac.uk

[1] Leibniz Institute for Agricultural Engineering and Bioeconomy (ATB), Max-Eyth-Allee 100, 14469 Potsdam, Germany

[2] Chair of Agromechatronics, Technische Universität Berlin, Straße Des 17. Juni 144, 10623 Berlin, Germany

[3] Department of Electrical and Computer Engineering, Engineering Faculty, Kharazmi University, Tehran, Iran

[4] Department of Electrical and Computer Engineering, University of California, 900 University Avenue, RiversideRiverside, CA 92521, USA

[5] Department of Engineering, Harper Adams University, Newport, UK