agricultural engineering.eu 80(1), 2025, 1–15

DOI:10.15150/ae.2025.3327

agricultural engineering.eu

Adaptive dynamic programming for robust path tracking in an agricultural robot using critic neural networks

Alireza Azimi, Redmond R. Shamshiri, Aliakbar Ghasemzadeh

Trajectory tracking control for agricultural mobile robots poses unique challenges due to inherent non-holonomic constraints and external disturbances, which can cause deviations from the desired path, affecting the robot's performance and operational efficiency. This paper presents an advanced learning-based control framework for robust path tracking in agricultural robots with Ackermann-steering mechanisms. Using Adaptive Dynamic Programming (ADP) and a Critic Neural Network, the proposed method handles external disturbances, including wheel slippage, which is common in agricultural environments. The Critic Neural Network the Hamilton-Jacobi-Isaacs (HJI) equation, allowing the controller to learn the near-optimal control policy in real time and adapt to environmental disturbances. The critic network's weights are updated online through an adaptive law, ensuring continuous learning and adaptation throughout the operation. Furthermore, the paper presents comprehensive simulation studies to evaluate the effectiveness of the proposed framework. The results demonstrate significant improvements in trajectory tracking performance compared to existing control methods, particularly in scenarios with substantial uncertainties and disturbances.

Keywords

Ackermann-steering robots, adaptive dynamic programming, agricultural robots, path tracking control, H infinity control, wheel slippage disturbance

In recent years, the agriculture industry has been increasingly leaning towards the use of autonomous robots and self-driving vehicles to address labor shortages and improve operational efficiency (Sun et al. 2023, Ju et al. 2022), Although all of these endeavors are early-stage projects developed by startups and established companies and not broadly implemented worldwide. This interest has been particularly evident in the area of navigation and precise trajectory tracking for Autonomous agricultural vehicles (AAV), including electrical tractors (Shamshiri 2024) and mobile robots, that operate in challenging environments such as vineyards, orchards, and fields (Shamshiri et al. 2024, Zhou et al. 2023, Iberraken et al. 2022, Ravankar et al. 2021, Moysiadis et al. 2020). Accurate predetermined trajectory tracking is essential for tasks like crop monitoring, yield mapping, targeted spraying, autonomous harvesting, and overall farming productivity (Ju et al. 2022, Zhang et al. 2020). However, these robots often face significant challenges in agricultural farms due to off-road terrain, non-holonomic constraints imposed by their steering mechanisms, sensor accuracy, and external disturbances like wheel slippage on uneven or loose soil surfaces (Jing et al. 2021, Lenain et al. 2006).

Several studies have explored techniques for autonomous tracking control and navigation of agricultural robots, each with unique strengths and limitations. A classical controller, such as the pro-

received 14 June 2024 | accepted 18 December 2024 | published 6 January 2025

portional-integral-derivative (PID), has been applied in specific applications. However, it may lack robustness when dealing with external disturbances and adaptability when changing working conditions (Monsalve et al. 2022, Nagasaka et al. 2009). For instance, Wang et al. (2020) proposed a PID-based cascade controller for yaw rate and lateral position control. Still, they did not consider varying terrain or weather conditions in the lateral displacement control system. Fuzzy logic control (FLC) is another approach to AAV trajectory tracking that diverges from traditional approaches by not relying on precise mathematical models. A PID and a type 2 FLC were used to control an autonomous tractor's longitudinal velocity and yaw dynamics, along with a learning algorithm to enhance its trajectory tracking performance (Kayacan et al. 2015). The study demonstrated a 30% improvement compared to conventional controllers. Liu et al. (2024) proposed a kinematic model and a fuzzy sliding mode approach law for the G33 lawn mower. They used the IA* path planning algorithm to enhance path tracking efficiency and lessen fuel consumption. It is important to note that their findings were achieved under undisturbed conditions. While FLC is a suitable option due to its robustness and flexibility, it can be challenging to implement in practical scenarios due to its dependence on human expertise. Model Predictive Control (MPC) is a popular control strategy extensively studied for agricultural vehicle path tracking.

A modified MPC method for articulated steering tractors, accounting for vehicle dynamics and constraints, was presented by Zhou et al. (2023). Carpio et al. (2020) proposed a navigation architecture for Ackermann vehicles based on MPC considering kinematic and dynamic constraints. These MPC approaches can handle constraints but require accurate system models and involve heavy mathematical calculations. Researchers have also explored sliding mode control (SMC) techniques for trajectory tracking control of AAVs. Ge et al. (2023) developed a robust adaptive SMC approach that adapts to parametric uncertainties in modeling and external disturbances like road banks. In Nagasaka et al. (2009) and Zhang et al. (2022) a nonsingular fast terminal SMC was integrated with a disturbance observer, achieving finite-time trajectory tracking for agricultural tractors. Although sliding mode control methods provide robust performance, they may experience chattering problems that lead to significant mechanical fatigue, making them unsuitable for actual AAV applications. Fast supertwisting sliding mode control (FSTSMC) and barrier function adaptive sliding mode control (BFASMC) were proposed by Yang et al. (2024) and Ding et al. (2004) to curb the chattering issue that exists in SMC. A fixed-time generalized super-twisting SMC method was also utilized to account for wheel slipping, improving tracking performance on varying terrains (Sun et al. 2023).

To address the challenges mentioned earlier for each controlling method, this paper proposes a robust adaptive kinematic control framework tailored for an Ackermann-steering agricultural mobile robot in the presence of slipping of the wheels as an example of external disturbance. The proposed approach uses an adaptive dynamic programming (ADP) approach, employing a critic neural network to approximate the Hamilton-Jacobi-Isaacs (HJI) equation associated with the H-infinity optimal control problem. The critic network's weights are updated online through an adaptive law, enabling the controller to continuously learn and adapt to changing system dynamics and disturbances while providing robust trajectory tracking performance.

The main contributions of this work can be summarized as follow: (i) derivation of a comprehensive kinematic model for non-holonomic constraints and external disturbances that can be expected in agricultural fields, (ii) development of a critic neural network approximation technique for solving the HJI equation, (iii) formulation of an adaptive learning law for online weight adaptation that ena-

bles the controller to handle disturbances, and (iv) integration of the proposed method into a robust adaptive control framework customized for an Ackermann-steering agricultural robot. The remainder of the paper is organized as follows: Section II details the robot's kinematic model, accounting for external disturbances. Section III outlines the preliminaries and problem setup for the H-infinity robust adaptive control approach. Section IV presents the neural network approximation and adaptive learning law. Section V discusses the kinematic controller design. Section VI reports the simulation results and performance evaluation on a realistic orchard environment under wheel slippage disturbances conditions.

Kinematic Model of Ackermann-Steering Agricultural Mobile Robot

This section establishes a kinematic model for an agricultural mobile robot with a double Ackermann steering mechanism. The model captures the robot's motion by analyzing the instantaneous behavior of its center of mass (COM). The robot under consideration is a four-wheeled vehicle with a double Ackermann steering mechanism provided in Figure 1. It possesses a rectangular base with dimensions a (length) and b (width). The rear and front wheels are steerable and aligned according to the Ackermann condition.

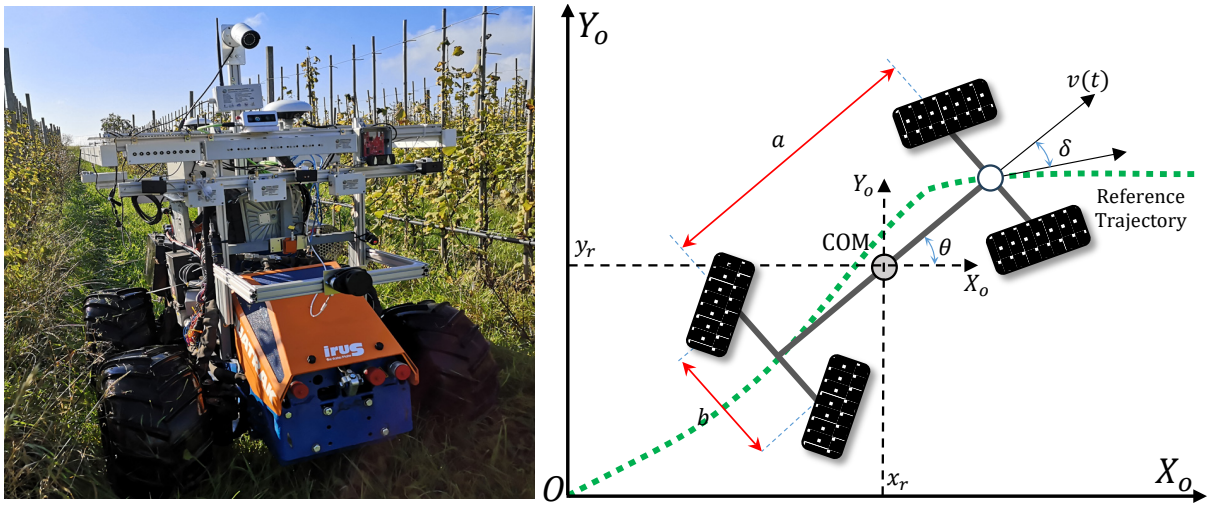

Figure 1: The double-Ackermann-Steering mobile robot used in the study

The model derivation revolves around analyzing the COM's instantaneous motion. Let (x_r, y_r) denote the COM's Cartesian coordinates, θ represents the robot's heading angle, and δ symbolizes the steering angle of the front wheels. These parameters collectively form the robot's posture vector, denoted as $q = [x, y, \theta, \delta]^T$. Additionally, v and ω represent the robot's linear and angular velocities about the COM, respectively. The ideal kinematic model of the robot is outlined in equation 1 (El Bou et al. 2022):

$$\dot{X} = J(X)U \quad , \quad \begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \end{bmatrix} = \begin{bmatrix} U_1(t)\ \cos(X_3(t)) \\ U_1(t)\ \sin(X_3(t)) \\ [U_1(t)\ \tan(X_4(t))]/L \\ U_2(t) \end{bmatrix} \qquad \text{(Eq. 1)}$$

The state vector X at time t is represented as $[X_1(t), X_2(t), X_3(t), X_4(t)]$, where $X_1(t)$ and $X_2(t)$ indicate the position coordinates in a 2D environment (x, y), $X_3(t)$ denotes the heading angle (θ) and $X_4(t)$ denotes the steering angle (δ). $U(t) = [U_1(t), U_2(t)]$ is the control vector, where $U_1(t) = v(t)$ and $U_2(t) = \omega(t)$, represents the linear and angular velocity, respectively. $J(X) = [\cos(X_3(t)), \sin(X_3(t)), \tan(X_4(t))/L, 0; 0,0,0,1]^T$ is the kinematic transformation matrix or the velocity transformation matrix. The superscript dot in equation 1 denotes the time derivative. These equations capture the core constraints imposed by the Ackermann steering geometry, where the robot cannot achieve pure lateral movement without altering its heading. The ideal kinematic model has limitations due to real-world agricultural environments that introduce external disturbances like slippage, requiring the incorporation of non-ideal kinematic effects. Therefore, the non-ideal kinematic of the robot is expressed in equation 2 and 3 (Keymasi and Jalalnezhad 2018). This equation includes time points T_1 and T_2 where disturbances occur, with $H(.)$ representing the unit step function (Keymasi and Jalalnezhad 2018). To provide a deeper analysis, the disturbance matrix $K(.)$ is the 4 × 4 identity matrix, indicating disturbances affecting all posture vector components x, y, θ, and δ. Following with the simplified disturbance vector of D as defined in equation 4 and D_{amp} is the amplitude of it. The derived kinematic model predicts efficient and safe robot motion in agriculture and helps design control algorithms for accurate maneuvers.

$$\dot{X} = J(X)U + K(X)D \tag{Eq. 2}$$

$$\begin{aligned} \dot{x} &= U_1 \cos\theta + 0.2H(t - T_1)\sin\theta \\ \dot{y} &= U_1 \sin\theta - 0.2H(t - T_1)\cos\theta \\ \dot{\theta} &= [U_1 \tan\theta + 0.2H(t - T_1)\tan\delta]/L \\ \dot{\delta} &= U_2 + 0.2H(t - T_1) - H(t - T_2) \end{aligned} \tag{Eq. 3}$$

$$D = \begin{bmatrix} 0.2H(t - T_1)\sin\theta \\ -0.2H(t - T_1)\cos\theta \\ (0.2H(t - T_1)\tan\delta)/L \\ H(t - T_1) - H(t - T_2) \end{bmatrix} \tag{Eq. 4}$$

Robust Adaptive Control: Preliminaries and Setup

The system under consideration is represented by equation 5. Here, $x \in \mathbb{R}^n$ denotes the state vector, $u \in \mathbb{R}^m$ represents the control input, and $z \in \mathbb{R}^p$ signifies the external disturbance. The functions $f(.)$, $g(.)$, and $h(.)$ capture the system dynamics, input dynamics, and disturbance dynamics, respectively. It is assumed that both u and z satisfy the L_2 norm on the interval $[0, \infty)$. Furthermore, these functions are assumed to be locally Lipschitz continuous, guaranteeing unique solutions, and $f(0) = 0$, implying the origin is an equilibrium point for the unforced system. Additionally, the system is presumed to be stabilizable and robustly controllable within a compact set $\Omega \in \mathbb{R}^n$ of the state space (Zhao et al. 2022). To determine the optimal control input (u), a cost function, as defined in equation 6, is formulated to be minimized. This function incorporates weighting matrices Q and R (positive definite diagonal) and a positive constant γ to ensure boundedness. The term γ_z represents the upper bound of uncertainties (Zhao et al. 2022). The H_∞ control problem is closely linked to the concept of

zero-sum games. Equation 7 expresses this connection by formulating the controller design task as finding a saddle point that satisfies the Nash condition. Here, *V*(x)* denotes the optimal value function. Differentiating equation 5 along system trajectories leads to the Bellman equation 8. This equation relates the optimal value function to the system dynamics and the cost function. The Bellman equation is further connected to the Hamilton equation 9. Through additional differentiation steps, the optimal values for the control input and disturbance, representing the two players in the zero-sum game, can be computed, as shown in equation 10 and 11. Substituting the optimal control u* (given by equation 10) and disturbance function z* (given by equation 11) into the Hamiltonian formulation (given by equation 9) yields the value function V(x) as given by equation 12. This equation represents the HJI partial differential equation, a cornerstone of the differential game formulation for H_∞ control.

$$\dot{x} = f(x) + g(x)u + h(x)z \quad \text{(Eq. 5)}$$

$$V(x(t)) = \int_t^\infty e^{-\gamma(\tau - t)}(x^T(\tau)Qx(\tau) + u^T(\tau)Ru(\tau) - {\gamma_d}^2(x))z\tau \quad \text{(Eq. 6)}$$

$$V^*(x) = J(x, u^*, z^*) = \min_u \max_d J(x, u, z) = \max_d \min_u J(x, u, z) \quad \text{(Eq. 7)}$$

$$\frac{dv}{dx}[f(x) + g(x)u + h(x)z] + x^T Qx + u^T Ru - \gamma V(x) + {\gamma_z}^2(x) \quad \text{(Eq. 8)}$$

$$H(x, V, u, z) = \frac{dv}{dx}[f(x) + g(x)u + h(x)z] + x^T Qx + u^T Ru - \gamma V(x) + {\gamma_z}^2(x) \quad \text{(Eq. 9)}$$

$$u^* = \frac{\partial H}{\partial u} = 0 \rightarrow u^* = -\frac{1}{2}R^{-1}g(x)^T \frac{dV^*}{dx} \quad \text{(Eq. 10)}$$

$$z^* = \frac{\partial H}{\partial d} = 0 \rightarrow z^* = \frac{1}{2\gamma^2}h(x)^T \frac{dV^*}{dx} \quad \text{(Eq. 11)}$$

$$V(x) = x^T Qx + \nabla {V^*}^T f(x) - \frac{1}{4}\nabla {V^*}^T g(x)R^{-1}g^T(x)\nabla V^* + \frac{1}{4\gamma}\nabla {V^*}^T h(x)k^T(x)\nabla V^* = 0 \quad \text{(Eq. 12)}$$

Since obtaining an analytical solution to the HJI equation is often impractical, an Approximate Dynamic Programming (ADP) algorithm will approximate the solution. This approach, widely used in various studies (Liu et al. 2017, Penar and Hendzel 2022, Modares et al. 2015), leverages concepts from reinforcement learning and utilizes an actor-critic structure. The actor approximates the strategies of both the controller and the disturbance, while the critic approximates the value function, which captures the long-term cost associated with each state and the corresponding optimal strategies (Liu et al. 2017, Hendzel and Penar 2019).

Neural Critic Approximation and Adaptive Learning

To overcome the challenge of solving the HJI equation, a critic neural network is employed to approximate the value function V, denoted as $\hat{V}$ (equation 13). $\hat{W} \in \mathbb{R}^i$ represents the estimated neural network weights (since the ideal weights are unknown), and $\phi(x) \in \mathbb{R}^i$ signifies a fundamental vector function associated with the neurons in the network. By substituting $\hat{V}$ from equation 13 into equations 10 and 11, approximations for the optimal control and disturbance signals can be derived, respectively equation 14 and 15. The next step involves determining the neural network weights ($\hat{W}$) online through an adaptive learning law. Here, we utilized equation 16 and 17 based on the work by Zhao et al. (2022), these formulations lead to the updated HJI equation in equation 18, incorporating the critic NN approximation. Here ε_{HJI} is the approximation error. To estimate the parameter W, filtered regressor matrices, $E \in \mathbb{R}^{i \times i}$ and $O \in \mathbb{R}^i$ in equations 19 and 20 are defined (Luan et al. 2019). For updating the value of E and O equations 21 and 22 are used, where η is a positive constant and initial values of these matrices are set to zero. Using updated E and O , an auxiliary vector $N \in \mathbb{R}^i$ is shown in equation 23. This vector incorporates the weight estimation error ($\tilde{W} = W - \hat{W}$) and a bounded variable, $\rho = \int_0^t exp\,(-\eta(t-\tau))\varepsilon_{HJI}\,A(\tau)d\tau$. Finally, the adaptive law is obtained through equation 24 (Zhao et al. 2022). This law governs how the estimated weights ($\hat{W}$) are adjusted online based on the auxiliary vector N and a positive learning gain, ξ. The online learning algorithm can converge by including the estimation error. For further details about this assumption and proof of it, readers are directed to refer to Theorem 1 of (Zhao et al. 2022).

$$\hat{V}(x) = \hat{W}^T \phi(x) \quad \text{(Eq. 13)}$$

$$\hat{u} = -\frac{1}{2} R^{-1} g(x)^T (\nabla \phi(x))^T \hat{W} \quad \text{(Eq. 14)}$$

$$\hat{z} = \frac{1}{2\gamma^2} h(x)^T (\nabla \phi(x))^T \hat{W} \quad \text{(Eq. 15)}$$

$$\mathrm{A} = -\gamma\phi + \nabla\phi(x)[f(x) + g(x)u] \quad \text{(Eq. 16)}$$

$$\mathrm{B} = {\gamma_z}^2(x) + x^T Q x + u^T R u \quad \text{(Eq. 17)}$$

$$\mathrm{B} = -W^T \mathrm{A} - \varepsilon_{HJI} \quad \text{(Eq. 18)}$$

$$\dot{\mathrm{E}} = -\eta \mathrm{E} + \mathrm{A}\mathrm{A}^T \quad \text{(Eq. 19)}$$

$$\dot{\mathrm{O}} = -\eta \mathrm{O} + \mathrm{A}\mathrm{B} \quad \text{(Eq. 20)}$$

$$\mathrm{E} = \int_0^t e^{-\eta(t-\tau)} \mathrm{A}(\tau)\mathrm{A}^T(\tau) d\tau \quad \text{(Eq. 21)}$$

$$\mathrm{O} = \int_0^t e^{-\eta(t-\tau)} \mathrm{A}(\tau)\mathrm{B}(\tau) d\tau \quad \text{(Eq. 22)}$$

$$N = -\mathrm{E}\tilde{W} + \rho \quad \text{(Eq. 23)}$$

$$\dot{\hat{W}} = -\xi N \quad \text{(Eq. 24)}$$

Robust Adaptive Kinematic Path Tracking Control

Error-based control is a common approach in robotics. Here, the errors (e) between the robot's current posture (q) and the desired reference posture (q_d) are defined (equation 25). Additionally, the derivatives of these errors ($\dot{e}$) are incorporated to form error-rate signals, as given in equation 26. Considering the system dynamics (equation 5) and treating the error as the system state, an updated version of the derivative error (equation 27) can be derived. This equation reformulates the kinematics of AAV in terms of the error state (e):

$$e = q - q_d \tag{Eq. 25}$$

$$\dot{e} = \dot{q} - \dot{q}_d \tag{Eq. 26}$$

$$\dot{e} = J(X)U - \dot{q}_d \tag{Eq. 27}$$

By comparing equations 27 and 5, it is recognized that $J(X)U$ and $\dot{q}_d$ can be reformulated as $g(e)u$ and $f(e)$, respectively, where $g(e)$ and $f(e)$ represent the system dynamics in terms of the error state (e). The wheels sliding term will be modeled as the external disturbance in the form of $h(e)z$. This allows us to make the kinematic model into the nonlinear affine system structure. After reformulating the kinematic model, the control input signal ($\hat{U}$) for the Ackermann-steering robot is defined in equation 28. This control law (equivalently, linear and angular velocities) incorporates the estimated weights ($\hat{W}$) derived from the Critic Neural Network and leverages the gradient of the activation function ($\nabla\phi(e)$) in conjunction with the matrix $J(.)$ of the kinematic model. Moreover, it employs the design matrix R to facilitate accurate path tracking capabilities for the robotic system.

$$\hat{U} = -\frac{1}{2}R^{-1}J(q)^T(\nabla\psi(e))^T\hat{W} \tag{Eq. 28}$$

Figure 2 depicts a schematic block diagram of the proposed kinematic control method with the key components breakdown as follow: (i) Reference Path Generation, (ii) neural estimator, and (iii) H_∞ Robust Controller. A reference desired trajectory for the robot is first generated using mathematical equations (specific equations are provided in the simulation section). The neural estimator block utilizes the robot's linear and angular velocity, the difference between the reference path and its position (equation 25), and their derivatives (equation 26) as inputs. Based on these inputs, the estimator approximates the HJI equation, providing an initial estimate for the controller. Based on the estimated weights from the neural estimator (equation 24) and the current location of the robot and relative error from the reference trajectory, this block (acting as the actor) generates the appropriate control signal using equation 28 (velocity commands) to guide the robot towards the desired trajectory. This control loop ultimately enables the Agriculture Mobile Robot (AAV) to track the reference trajectory with minimal deviation, even in the presence of slip modeled as an external disturbance.

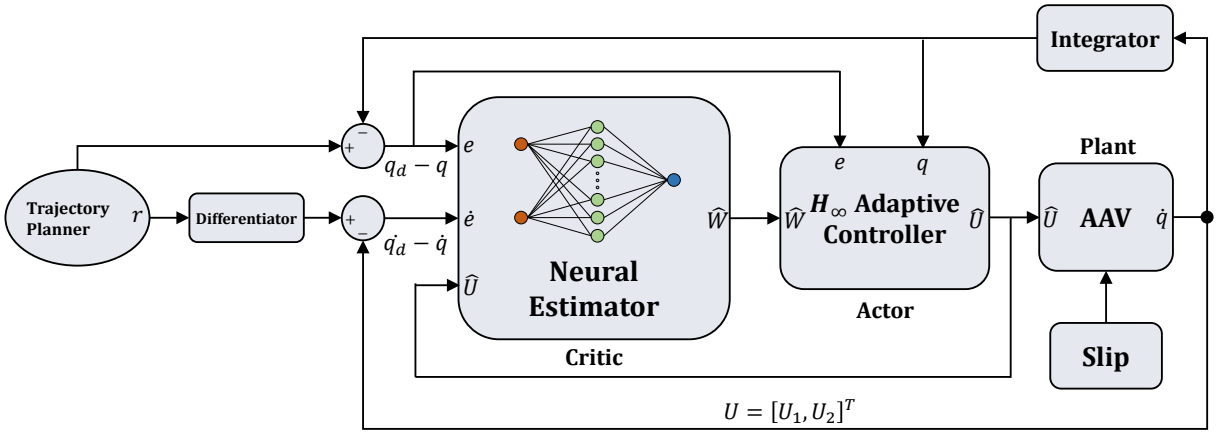

Figure 2: Schematic block diagram of the proposed kinematic controller

Simulation Results

A numerical simulation study is conducted using MATLAB-Simulink to test the effectiveness of the proposed control scheme under the assumption of wheel slippage on a double Ackermann-steering mobile robot with a wheelbase of 0.8 meters to follow a predefined trajectory in a vineyard environment consisting of 10-meter straight lines and 5-meter (diameter) half-circles at the end of each row. The track lines are placed at a distance of 5 meters from each other as shown in Figure 3.

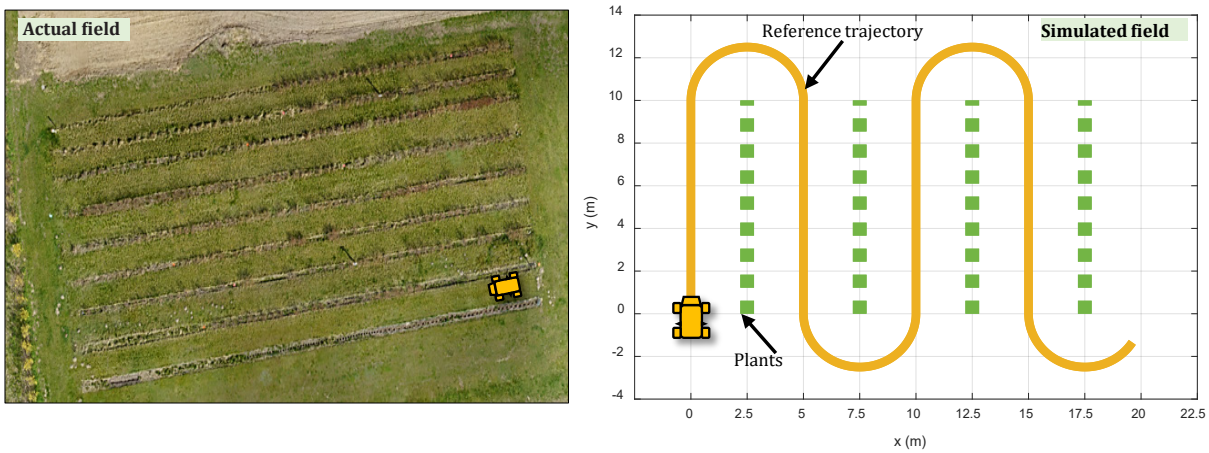

Figure 3: View of the actual and simulated field, showing reference trajectory with U-turns

The orange and black lines represent the reference and robot trajectories, respectively, and the green squares indicate plants in the track lines. The robot's initial condition for maneuvering is specified as $q = [-0.8, -1.2, 4\pi/9, 0]^T$, which is far from the beginning of the reference trajectory. The adaptive learning process relies on various parameters. These include the initial value of estimated critic neural network weights, filtered regressor matrices and their related constant parameters (which can be found in equations 16 to 24, and design matrices in the value function (as described in equation 6). All of these parameters are illustrated in equation 29. To illustrate better, the Critic Neural Network used in this framework is a feedforward neural network, comprising one hidden layer with 10

neurons. The activation function used is given by equation 30, which is the Kronecker product of the errors. The network's weights are initialized randomly and by excessive trial and error process and updated continuously during the operation, allowing the controller to learn in real-time. No dropout or batch normalization techniques were applied, as real-time learning was prioritized over static optimization.

$$\begin{aligned}&\widehat{W}(0) = 19 \times [1,1,1,1,1,1,1,5,1,5]^T\\&\mathrm{E}(0) = \mathrm{diag}\{1,1,1,1,1,1,1,1,1,1\}\\&\mathrm{O}(0) = [1,1,1,1,1,1,1,1,1,1]^T\\&\gamma = 0.5, \eta = 100, \xi = 50\\&Q = \mathrm{diag}\{1,1,1,1\}, R = \mathrm{diag}\{0.01,0.01\}\end{aligned} \quad \text{(Eq. 29)}$$

$$\phi = [e_1^2, e_1e_2, e_1e_3, e_1e_4, e_2^2, e_2e_3, e_2e_4, e_3^2, e_3e_4, e_4^2]^T \quad \text{(Eq. 30)}$$

This nonlinear mapping of the error signals and their gradients is used in the approximation process and in producing control signals, respectively. e_1, e_2, e_3 and e_4 correspond to the first to fourth elements of posture error (i.e. e_{X_1}, e_{X_2}, e_{X_3}, e_{X_4}). The simulation results on the Ackermann mobile robot are shown in Figure 4 to Figure 8 under the assumption of wheel slippage. The slippage can be modeled as $D = 0.5H(t - 20) \times [\sin\theta, \cos\theta, \tan\delta/0.8, 0]^T + [0,0,0,H(t - 20) - H(t - 140)]^T$. In Figure 4a, the wheels of the robot began to slip from the 20th second when it was trying to track the half-circle row-end. This continued until the end of the experiment, which was the 140th second, as indicated by the red lines and circles. Even though the system was disrupted, the controller guided the robot to follow the predetermined path. As shown in Figure 4b–e, all posture-related states accurately matched their desired values.

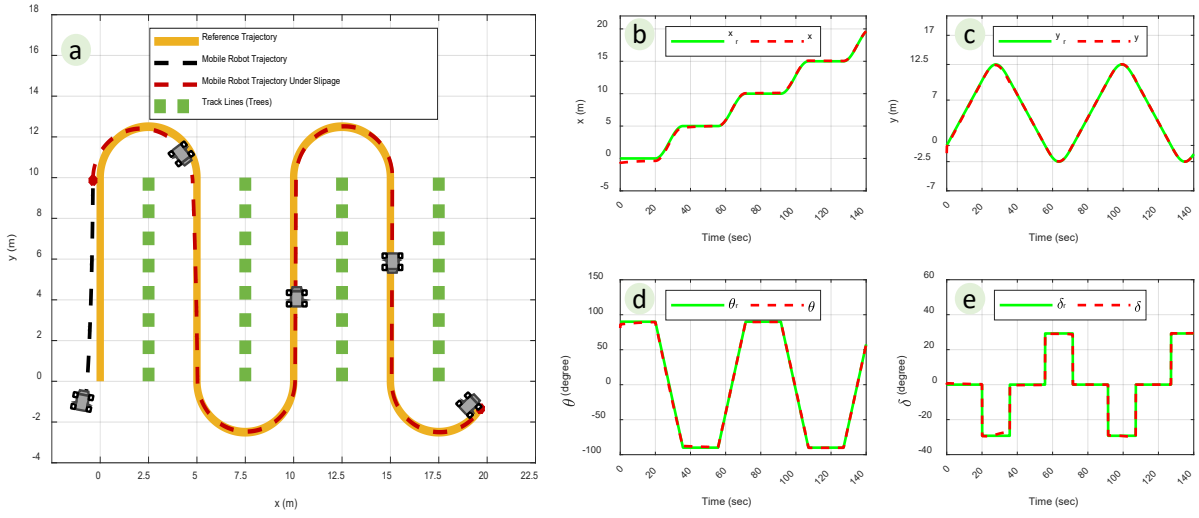

Figure 4: Demonstration of (a) robot trajectory with wheel slippage (b) position error in x-coordinate, (c) position error in y-coordinate, (d) heading angle error, and (e) steering angle error

Figure 5 displays the trajectory and angle tracking profiles. The trajectory tracking (Figure 5a) error shows that the robot can maintain a distance of 0.1 meters to its reference path even during slippage. At the 20th second, the robot is affected by the slippage and deviates from its desired path,

but it is able to recover and get back on track. This characteristic can also be observed in the angle tracking (Figure 5b), where the controller can accurately follow the desired profiles and effectively reduce the impact of disturbance caused by wheel slippage, with errors remaining below 0.02, which is acceptable for agricultural applications.

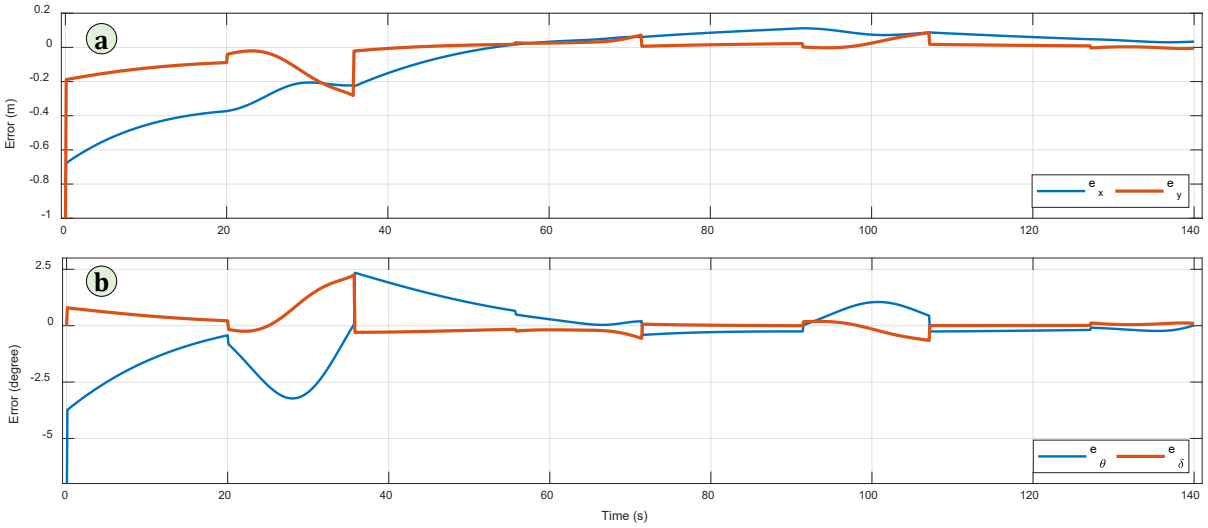

Figure 5: Trajectory and angle tracking error (a) trajectory tracking error in XY axis (b) robot orientation and steering angle tracking error

In order to thoroughly assess the effectiveness of the proposed control method, Table 1 and Figure 6 have been included to demonstrate the tracking errors. Various approaches have been utilized to calculate the errors such as Root Mean Squared Error (RMSE), Integral Error (IE), Integral Absolute Error (IAE), Integral Time Absolute Error (ITAE), Integral Squared Error (ISE), and Integral Time Squared Error (ITSE). The RMSE metric in Figure 6a reveals that the longitude tracking (e_x) has the highest error of 0.21, indicating a substantial deviation from the desired trajectory along the x-axis. Conversely, the steering angle tracking (e_δ) demonstrates the lowest RMSE of 0.01, suggesting precise control over the robot's steering mechanism. It is worth mentioning that different agricultural tasks have varying precision requirements, which should be accounted for when designing control systems. For instance, autonomous harvesting might require higher precision (within a few centimeters) to ensure accurate manipulation of crops, while tasks like yield mapping or crop monitoring may tolerate slightly higher positional errors. Based on existing studies, a precision threshold of approximately 2–3 cm for lateral position errors is generally considered sufficient for most agricultural operations. In our proposed framework, the Root Mean Squared Error (RMSE) for trajectory tracking is below 0.21 meters, which aligns with the acceptable range for applications like row navigation and targeted spraying. As shown in Figure 6b for IE, Negative values for e_x, e_y, and e_θ imply that the robot lagged behind the reference trajectory, undershooting the desired position and orientation. However, the positive value for e_δ suggests an overshooting behavior in the steering angle control. IAE and ITAE (Figure 6c, 6d) metrics offer additional perspectives on the cumulative error magnitudes. Consistent with the RMSE and IE results, the longitude tracking exhibits the highest IAE (21.69) and ITAE (924.10) values, further emphasizing the challenges in accurately following the x-coordinate trajectory. The steering angle tracking, on the other hand, exhibits the lowest IAE (0.67) and ITAE (29.75),

indicating superior control performance in this aspect. ISE and ITSE metrics, which assign higher weights to larger errors in Figure 6e and 6f, also support the findings from the other error calculation approaches. The longitude tracking (e_x) demonstrates the highest ISE (6.40) and ITSE (130.67) values, suggesting the presence of significant deviations from the desired trajectory along the x-axis. Conversely, the steering angle tracking (e_δ) exhibits the lowest ISE (0.01) and ITSE (0.32), indicating minimal large errors in this aspect of the control system's performance.

Table 1: Numerical values of different methods of error calculation

Errors	RSME	IE	IAE	ITAE	ISE	ITSE
e_x	0.21	−6.94	21.69	924.10	6.40	130.67
e_y	0.07	−2.15	6.35	256.34	0.69	19.00
e_θ	0.02	−0.63	2.21	91.99	0.07	1.85
e_δ	0.01	0.2	0.67	29.75	0.01	0.32

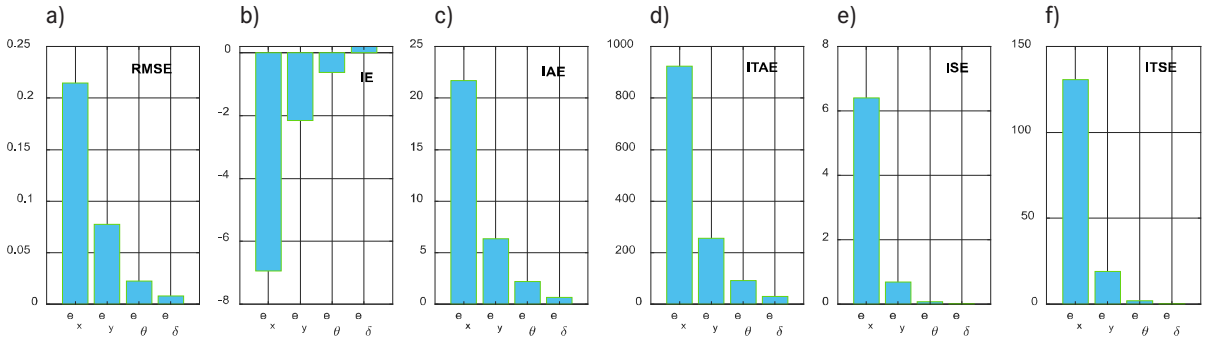

Figure 6: Assessing tracking accuracy with different error calculation metrics, including RMSE, IE, IAE, ITAE, ISE, and ITSE

Figure 7 illustrates the evolution of the controller's weights, which converge to the following values after undergoing changes (18.5, −1.3, −5, −5.9, 9.8, −16.3, 29.7, 51.3, 9.1, 93). During the wheel slippage condition, the w_1, w_2, w_4, w_5 and w_9 see tangible changes. This weight adaptation process demonstrates the controller's capability to adjust its parameters in response to the disturbance caused by wheel slippage, ultimately converging to the appropriate values for stable trajectory tracking performance. Overall, the simulation results showcase the proposed controller's effectiveness in achieving accurate trajectory tracking, handling disturbances, and adapting its parameters through learning to maintain stable and precise control performance within the vineyard environment.

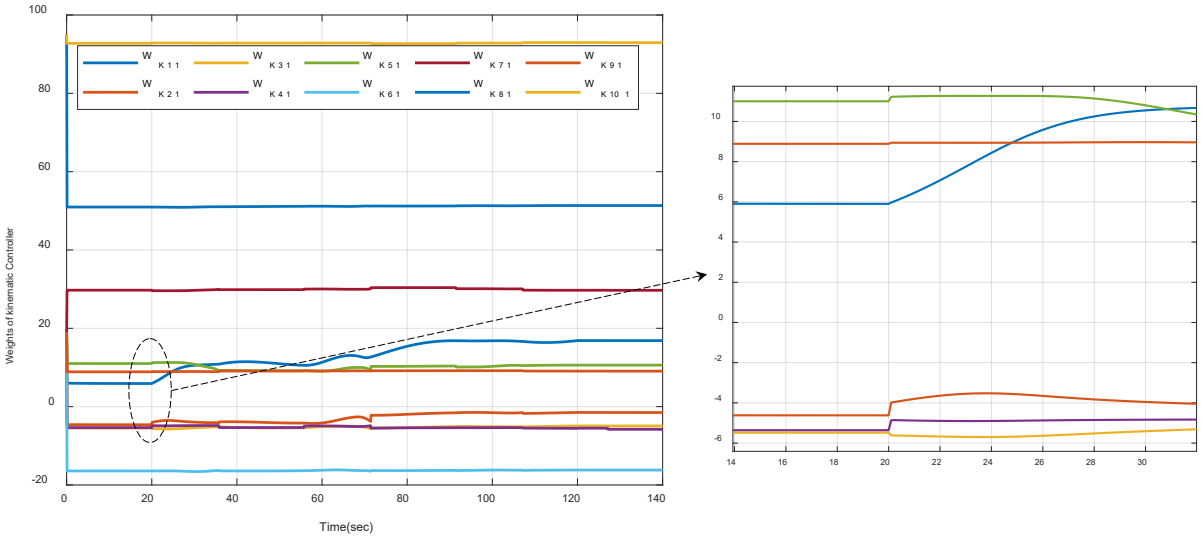

Figure 7: Adaptation of critic neural network weights

Figure 8 displays the different levels of disturbance boundaries studied during the simulation. The numerical values (0.5, 0.7, 0.8) represent the tested disturbance magnitudes. As shown in the figure, when the disturbance magnitude is equal to or greater than 0.8, it causes significant deviation from the desired trajectory, leading to instability. On the other hand, when the magnitudes are equal to or smaller than 0.7, the tracking performance is much better. With an amplitude of 0.8, the robot can maneuver for nearly 40 seconds after slippage and then rotate itself and move back on the line; however, it cannot follow the second row and exit from the field.

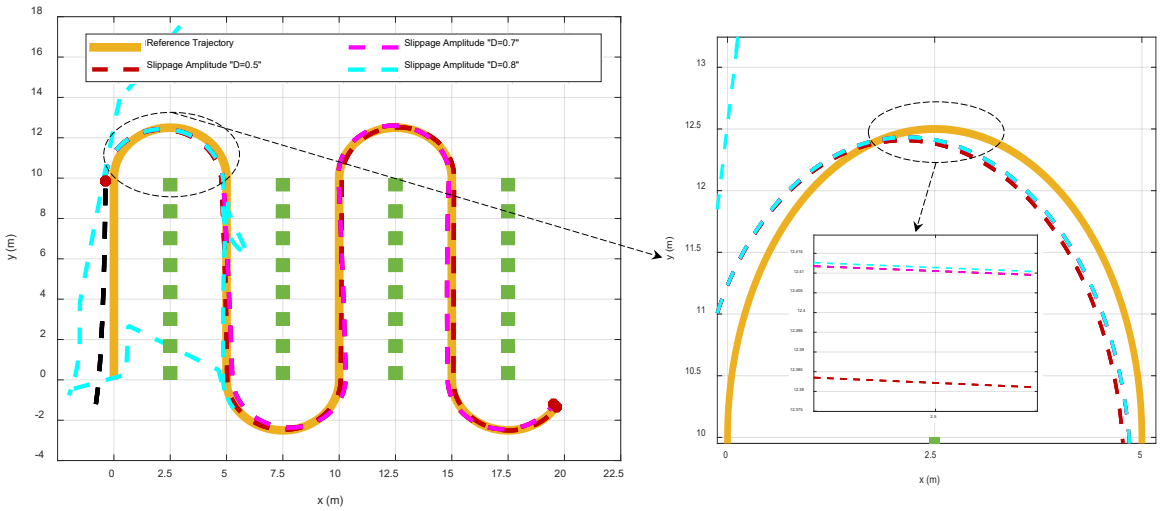

Figure 8: Tracking performance for different wheel slippage disturbance magnitudes

Discussion

The proposed framework‘s reliance on continuous online learning and adaptation through the Critic Neural Network does introduce computational complexity, which may pose challenges in real-time applications. The neural network updates must be processed rapidly to ensure responsiveness to changing environmental conditions. Hardware considerations, such as GPU acceleration or optimized embedded systems, may be necessary to meet the real-time demands of agricultural robots. Additionally, energy consumption may increase due to the continuous learning process, making energy-efficient computation strategies an important consideration for future implementations. While the current study validates the control framework through comprehensive simulation, real-world testing will provide a more complete evaluation of its performance under true agricultural conditions. Future work will involve testing the system in real-world scenarios such as vineyards and orchards, where wheel slippage and uneven terrain are common. Additionally, we plan to compare the proposed method with other established control methods like Model Predictive Control (MPC) to quantify its advantages in trajectory tracking precision, energy consumption, and computational efficiency.

It should also be noted that that there is no set rule for determining the best values at each stage of implementing a neural network. The results included in the presented study are the product of extensive work to determine the best configuration for this setup. We took into account not only the control effort but also the effectiveness of the controller in addressing this problem. It‘s worth noting that existing methods of Adaptive Dynamic Programming (ADP) also lack additional information in the published literatures, even in the most cited ones. Overall, for the current scope, we considered the value of weights and gains from the minimum values to a reasonable range to achieve these results and we explained that the chosen configuration demonstrates the proposed framework’s ability to adapt and track the desired path with the minimum possible errors for this platform. Another important point to mention is that due to the nature of approximation in finding solutions to the HJI equation, we should not expect the results to be optimal. To the best of out knowledge, in majority of the published literatures, the results are described as being somewhat near-optimal.

Conclusions

The research proposes an adaptive kinematic control framework for Ackermann steering agricultural robots, addressing issues like wheel slippage with focus on controlling the kinematics of the robot. The framework includes a comprehensive kinematic model and an adaptive dynamic programming approach using neural networks to ensure robust path tracking performance. The adaptive controller utilizes the estimated weights from the critic network with the kinematic model to compute velocity commands for the robot, allowing it to accurately track desired trajectories while compensating for external disturbances like wheel slippage. The results validate the controller‘s ability to achieve stable path tracking even when wheel slip occurs. While the robust control framework proposed in this study can handle general disturbances, a more detailed model that considers gravitational effects and multi-wheel slippage would provide greater accuracy. However, addressing these effects required expanding the scope of the study and considering the dynamic model of the mobile platform. In this context, future studies can incorporate the forces affecting the systems into the equation of the dynamic model. Future work could also explore alternative neural network architectures or advanced learning algorithms to improve performance and adaptability further. Experimental testing in real-world farm environments would also be valuable to assess practical implementation aspects.

References

Carpio, R. F. et al. (2020): A Navigation Architecture for Ackermann Vehicles in Precision Farming. IEEE Robotics and Automation Letters 5(2), pp. 1103–1110, https://doi.org/10.1109/LRA.2020.2967306

Ding, C.; Ding, S.; Wei, X.; Ji, X.; Sun, J.; Mei, K. (2024): Disturbance-Observer-Based Barrier Function Adaptive Sliding Mode Control for Path Tracking of Autonomous Agricultural Vehicles with Matched-Mismatched Disturbances. IEEE Transactions on Transportation Electrification, pp. 1–1, https://doi.org/10.1109/TTE.2023.3333001

El Bou, C. M.; Von Ellenrieder, K. D.; Gupta, S. K. (2022): A homogeneity-based path following shared control system for UGVs. 30th Mediterranean Conference on Control and Automation (MED), Vouliagmeni, Greece, 2022, pp. 725–730, https://doi.org/10.1109/MED54222.2022.9837268

Ge, Z. et al. (2023): Robust adaptive sliding mode control for path tracking of unmanned agricultural vehicles. Computers and Electrical Engineering 108, p. 108693, https://doi.org/10.1016/j.compeleceng.2023.108693

Hendzel, Z.; Penar, P. (2019): Zero-Sum Differential Game in Wheeled Mobile Robot Control. pp. 151–161, https://doi.org/10.1007/978-3-030-15857-6_16

Iberraken, D.; Gaurier, F.; Roux, J. C.; Chaballier, C.; Lenain, R. (2022): Autonomous Vineyard Tracking Using a Four-Wheel-Steering Mobile Robot and a 2D LiDAR. AgriEngineering,4(4), pp. 826–846, https://doi.org/10.3390/agriengineering404005

Jing, Y.; Liu, G.; Luo, C. (2021): Path tracking control with slip compensation of a global navigation satellite system based tractor-scraper land levelling system. Biosystems Engineering 212, pp. 360–377, https://doi.org/10.1016/j.biosystemseng.2021.11.010

Ju, C.; Kim, J.; Seol, J.; Il Son, H. (2022): A review on multirobot systems in agriculture. Computers and Electronics in Agriculture 202, p. 107336, https://doi.org/10.1016/j.compag.2022.107336

Kayacan, E.; Kayacan, E.; Ramon, H.; Kaynak, O.; Saeys, W. (2015): Towards Agrobots: Trajectory Control of an Autonomous Tractor Using Type-2 Fuzzy Logic Controllers. IEEE/ASME Transactions on Mechatronics 20(1), pp. 287–298, https://doi.org/10.1109/TMECH.2013.2291874

Keymasi Khalaji, A.; Jalalnezhad, M. (2018): Control of a wheeled robot in presence of sliding of wheels using adaptive backstepping method, vol. 18, no. 4, pp. 144–152, https://mme.modares.ac.ir/article-15-16838-en.html, accessed on 1 Dec 2023

Lenain, R.; Thuilot, B.; Cariou, C.; Martinet, P. (2006): Mobile robot control in presence of sliding: Application to agricultural vehicle path tracking. Proceedings of the 45th IEEE Conference on Decision and Control, pp. 6004–6009, https://doi.org/10.1109/CDC.2006.377520

Liu, D.; Wei, Q.; Wang, D.; Yang, X.; Li, H. (2017): Adaptive Dynamic Programming with Applications in Optimal Control. Springer Cham, https://doi.org/10.1007/978-3-319-50815-3

Liu, L. et al. (2024): Path Planning and Tracking Control of Tracked Agricultural Machinery Based on Improved A* and Fuzzy Control. Electronics 13(1), p. 188, https://doi.org/10.3390/electronics13010188

Luan, F.; Na, J.; Huang, Y.; Gao, G. (2019): Adaptive neural network control for robotic manipulators with guaranteed finite-time convergence. Neurocomputing 337, pp. 153–164, https://doi.org/10.1016/J.NEUCOM.2019.01.063

Modares, H.; Lewis, F. L.; Jiang, Z. P. (2015): H_∞ Tracking Control of Completely Unknown Continuous-Time Systems via Off-Policy Reinforcement Learning. IEEE Transactions on Neural Networks and Learning Systems 26(10), pp. 2550–2562, https://doi.org/10.1109/TNNLS.2015.2441749

Monsalve, G.; Ltaief, N. B.; Amoriya, V.; Cardenas, A. (2022): Kinematic Navigation Control of Differential Drive Agricultural Robot. 2022 IEEE International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM), pp. 1–6, https://doi.org/10.1109/CISTEM55808.2022.10043876

Moysiadis, V.; Tsolakis, N.; Katikaridis, D.; Sørensen, C. G.; Pearson, S.; Bochtis, D. (2020): Mobile Robotics in Agricultural Operations: A Narrative Review on Planning Aspects. Applied Sciences 10(10), p. 3453, https://doi.org/10.3390/app10103453

Nagasaka, Y.; Saito, H.; Tamaki, K.; Seki, M.; Kobayashi, K.; Taniwaki, K. (2009): An autonomous rice transplanter guided by global positioning system and inertial measurement unit. Journal of Field Robotics 26(6–7), pp. 537–548, https://doi.org/10.1002/rob.20294

Penar, P.; Hendzel, Z. (2022): Experimental Verification of the Differential Games and H_∞ Theory in Tracking Control of a Wheeled Mobile Robot. Journal of Intelligent and Robotic Systems 104(4), p. 61, https://doi.org/10.1007/s10846-022-01584-6

Ravankar, A.; Ravankar, A. A.; Rawankar, A.; Hoshino, Y. (2021): Autonomous and Safe Navigation of Mobile Robots in Vineyard with Smooth Collision Avoidance. Agriculture 11(10), p. 954, https://doi.org/10.3390/agriculture11100954

Shamshiri, R. (2024): Electrical Tractors for Autonomous Farming. In: Shamshiri, R.; Hamed, I.(eds.): Mobile Robots for Digital Farming. CRC Press, Boca Raton, p. 89-106, https://doi.org/10.1201/9781003306283

Shamshiri, R.R., Navas, E., Dworak, V.; Schütte, T.; Weltzien, C., Auat Cheein, F. A. (2024): Internet of robotic things with a local LoRa network for teleoperation of an agricultural mobile robot using a digital shadow. Discov Appl Sci 6, 414, https://doi.org/10.1007/s42452-024-06106-7

Sun, J.; Li, Q.; Ding, S.; Xing, G.; Chen, L. (2023): Fixed-time generalized super-twisting control for path tracking of autonomous agricultural vehicles considering wheel slipping. Computers and Electronics in Agriculture 213, p. 108231, https://doi.org/10.1016/j.compag.2023.108231

Wang, S.; Yin, C.; Gao, J.; Sun, Q. (2020): Lateral Displacement Control for Agricultural Tractor Based on Cascade Control Structure. International Journal of Control Automation and Systems 18(9), pp. 2375–2385, https://doi.org/10.1007/s12555-019-0428-3

Yang, W.; Ding, S.; Ding, C. (2024): Fast Supertwisting Sliding Mode Control With Antipeaking Extended State Observer for Path-Tracking of Unmanned Agricultural Vehicles. IEEE Transactions on Industrial Electronics 71(10), pp. 1–10, https://doi.org/10.1109/TIE.2024.3355507

Zhang, T.; Jiao, X.; Lin, Z. (2022): Finite time trajectory tracking control of autonomous agricultural tractor integrated nonsingular fast terminal sliding mode and disturbance observer. Biosystems Engineering 219, pp. 153–164, https://doi.org/10.1016/j.biosystemseng.2022.04.020

Zhang, Z.; Kayacan, E.; Thompson, B.; Chowdhary, G. (2020): High precision control and deep learning-based corn stand counting algorithms for agricultural robot. Autonomous Robots 44(7), pp. 1289–1302, https://doi.org/10.1007/s10514-020-09915-y

Zhao, J.; Na, J.; Gao, G. (2022): Robust tracking control of uncertain nonlinear systems with adaptive dynamic programming. Neurocomputing 471, pp. 21–30, https://doi.org/10.1016/J.NEUCOM.2021.10.081

Zhou, B.; Su, X.; Yu, H.; Guo, W.; Zhang, Q. (2023): Research on Path Tracking of Articulated Steering Tractor Based on Modified Model Predictive Control. Agriculture 13(4), pp. 871, https://doi.org/10.3390/agriculture13040871

Authors

Dr Redmond R. Shamshiri is a senior researcher at Leibniz Institute for Agricultural Engineering and Bioeconomy (ATB), Max-Eyth-Allee 100, 14469 Potsdam, Germany. E-Mail: rshamshiri@atb-potsdam.de

Alireza Azimi and **Aliakbar Ghasemzadeh** are research assistants at Control Lab at Kharazmi University, Shahid Mofatteh Ave., Kharazmi University, Tehran, Iran