Hierarchical Control-based Trajectory Tracking Control of Robots

Ningxin Wang[1], Dongsong Li[2], Xiaoling Xu[3], Diantao Wang[4], Zhenhuai Ma[5], Zuoxun Wang[1, *]
[1]Shandong Technology and Business University, Yantai 264005, China
[2]Shandong Luneng Intelligence Technology Co., Ltd., Jinan 250101, China
[3]Shandong Haiguan Electric Co., Ltd., Jinan 250014, China
[4]Yantai Dongfang Wisdom Electric Co., Ltd., Yantai 264003, China
[5]Tajik National University, Dushanbe 734025, Republic of Tajikistan
*Corresponding email: 202214078@sdtbu.edu.cn

Abstract

Regular room inspections are crucial for the health and safety of elderly residents in nursing homes. To address the trajectory tracking problem of nursing robots, a hierarchical control framework is proposed. The upper layer combines an H∞ unscented Kalman filter (HUKF) with nonlinear model predictive control (NMPC) to enhance state estimation and optimize the reference trajectory, while the lower layer uses a sliding mode controller for high-response actuation. Simulation results demonstrate that the proposed control scheme achieves excellent tracking accuracy and robustness, significantly reducing deviations caused by slippage, modeling errors, and disturbances. These results offer a solid foundation for the intelligent control of skid-steering nursing robots in complex environments.

Keywords

H∞ filtering, Unscented Kalman filter, Nonlinear model predictive control, Sliding mode control

Introduction

Shortages of workers have been especially salient in elderly care. In many countries, nursing homes experience persistent staff shortages and high levels of turnover as the elderly population and demand for caregiving services grows [1]. With the development of robot technology and artificial intelligence, the emergence of nursing robots can address the above-mentioned problems. Nursing robots are commonly used for delivery, fetching supplies, cleaning, and aiding in direct-care activities [2]. However, when applied to the complex, dynamic, and highly safety-critical environment of nursing homes, the core challenge for nursing robots lies in their ability to not only accurately and stably follow the preset patrol trajectory but also to flexibly and safely handle dynamic obstacles in the corridors, such as walking elderly individuals, moving wheelchairs, and temporarily placed objects. Therefore, trajectory tracking control for nursing robots has become a hot and challenging research topic.

The performance of the trajectory tracking controller depends on the tracking accuracy and stability of the controller [3]. Traditional trajectory tracking control methods, such as proportional-integral-derivative (PID) control, fuzzy control, and neural networks, can, to some extent, satisfy control requirements under smooth road conditions [4-7]. However, in scenarios with severe slippage, parameter uncertainties, or significant external disturbances, these approaches often suffer from large tracking errors and poor system stability.

Hua et al. addressed the trajectory tracking control problem for multi-robot systems in complex environments by proposing a robust control framework that combines multi mobile robot trajectory model predictive control (MMRT-MPC) and the trajectory action dependence graph (TADG) [8]. Xu et al. tackled the issue that most existing trajectory tracking methods primarily focus on pose errors while neglecting the effects of path curvature [9]. Li and Zhou addressed the need for simultaneous trajectory tracking and obstacle avoidance for unmanned surface vehicles (USVs) in complex marine environments by proposing a real-time nonlinear model predictive control (NMPC) method [10]. Jesús A. et al. proposed a nonlinear H∞ control strategy based on a disturbance observer, which is used to

enhance the robustness of wheeled mobile robots in trajectory tracking tasks [11].

This controller can handle both matching and non-matching disturbances, by transforming the system into a second-order system affected by bounded disturbances and designing an H∞ controller to achieve trajectory tracking. Li et al. proposed an intelligent PID guidance control approach for autonomous underwater vehicle (AUV) path tracking [12]. The method integrates a traditional PID control algorithm with intelligent optimization techniques to enhance tracking performance, especially in dynamic and uncertain environments. Li et al. proposes an advanced nonlinear model predictive control (NMPC) method designed for autonomous trucks (ATs) to solve the trajectory tracking problem while managing the complexities of coupled longitudinal and lateral dynamics [13]. The key innovation in the study is the inclusion of terminal ingredients - such as terminal control gain, terminal constraints, and terminal cost functions - in the optimization control problem.

In recent years, control methods based on the hierarchical structure of Model Predictive Control and Sliding Mode Control have become a research focus [14-18]. The hierarchical control framework typically divides the trajectory tracking task into a trajectory tracking layer and a drive control layer [19]. The upper layer uses MPC for trajectory planning and reference trajectory generation, while the lower layer employs sliding mode control or robust control to achieve wheel speed closed-loop tracking, effectively enhancing the system's overall performance under complex operating conditions.

Considering that nursing robots require high stability and flexibility in their movement, a four-wheeled skid-steering robot model, is introduced [20]. This paper addresses the kinematic and dynamic models of four-wheel skid-steering mobile robots analyzes the effects of skid, and designs a hierarchical trajectory tracking control algorithm based on Heuristic Unscented Kalman Filter-nonlinear Model Predictive Control (HUKF-NMPC) and sliding mode control. The trajectory tracking task is divided into two layers: The HUKF-NMPC based trajectory tracking layer and the robust control-based drive control layer. The block diagram of the control algorithm is shown in Figure 1. The effectiveness and robustness of the proposed approach are validated through simulations and experiments, providing both theoretical foundation and technical support for achieving high-precision trajectory tracking control of skid-steering robots.

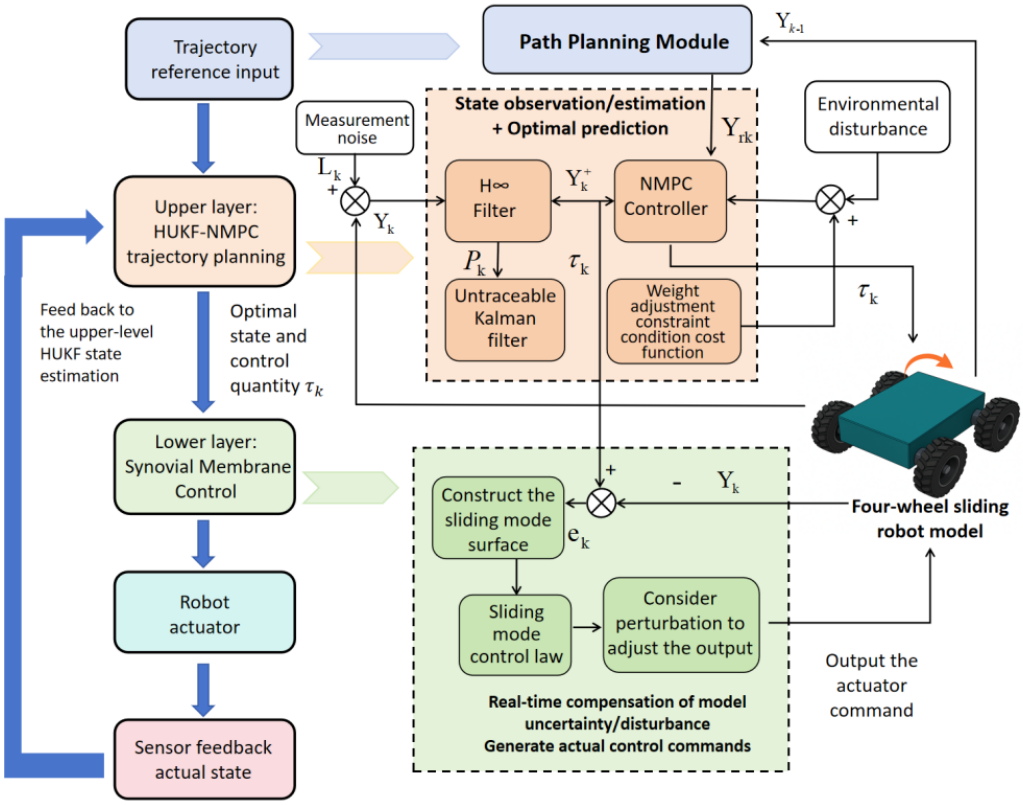


Figure 1. Hierarchical trajectory tracking control algorithm diagram.

Wheeled skid-steering robot model

Skid-steering robot coordinate system

The coordinate system of the skid-steering robot is shown in Figure 2. The global Cartesian coordinate system $O(X, Y, Z)$ is defined, with the Z-axis pointing vertically upwards. A local Cartesian coordinate system $O_r(X_r, Y_r, Z_r)$ is established at the robot's center of mass, moving with the robot, where the X-axis is aligned with the robot's longitudinal center line. The generalized position coordinate system of the robot is represented by the vector $S=[x, y, \theta]^T$, where x and y are the longitudinal and lateral coordinates of the robot's center of mass, and θ is the robot's yaw angle, the angle between the X_r-axis and the positive X-axis. The generalized velocity coordinate system of the robot is represented by the vector $V=[v_x, v_y, \omega]^T$, where v_x is the robot's X_r forward velocity, v_y is the Y_r lateral velocity, and ω is the yaw rate, the rotational speed around the Z_r-axis.

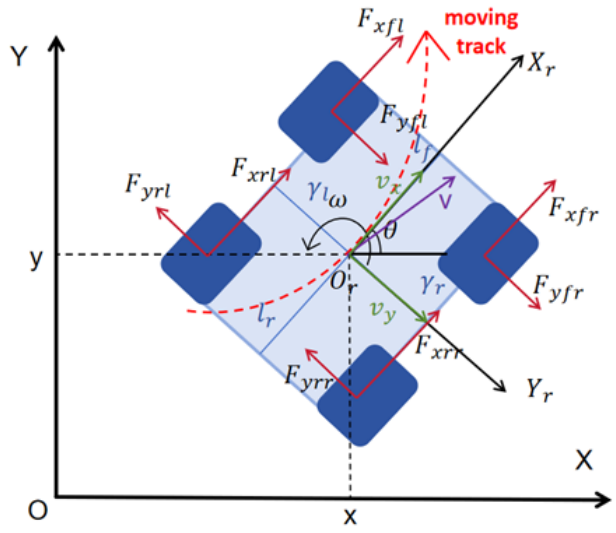


Figure 2. Force diagram of the four-wheeled sliding robot.

Kinematic model

$\dot{S}$ is the rate of change of the robot's position, and the change between the vector V and $\dot{S}$ is given by equation:

$$S=\dot{F}(\theta)V \tag{1}$$

Where the transformation matrix $F(\theta)$ between the position coordinate system and the velocity coordinate system is:

$$F(c)=\begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix} \tag{2}$$

Dynamic model

Assume the robot has motion in three directions: longitudinal, lateral, and yaw. Considering the forces acting on the robot during steady uniform motion around the instantaneous center of rotation, the dynamic model can be derived as follows:

$$\begin{cases} m\dot{v}_x = m\omega v_y + F_{xfl} + F_{xfr} + F_{xrl} + F_{xrr} \\ m\dot{v}_y = -m\omega v_x + F_{yfl} + F_{yfr} + F_{yrl} + F_{yrr} \\ J_\omega\dot{\omega} = \gamma_l(F_{xfl} + F_{xrl}) - \gamma_r(F_{xfr} + F_{xrr}) \\ \quad +l_f(F_{yfl} + F_{yrl}) - l_r(F_{yfr} + F_{yrr}) \end{cases} \tag{3}$$

In the equation: m is the total mass of the robot; J_ω is the moment of inertia of the robot around the Z_r-axis yaw; F_{xi} is the longitudinal force applied by the ground to the wheel, and F_{yi} is the lateral force applied by the ground to the wheel; l_f and l_r are the longitudinal distances from the front and rear wheel axles to the robot's center of mass, respectively; γ_l and γ_r are the lateral distances from the left and right wheels to the robot's center of mass, respectively. The subscript $i \in \{fl, fr, rl, rr\}$ refers to the left front, right front, left rear, and right rear wheels, respectively.

The dynamic model of each wheel can be represented as:

$$J_\gamma\dot{\omega}_i+RF_{xi}=\eta\kappa\tau_i \tag{4}$$

In the equation: J_γ is the moment of inertia of the wheel; $\dot{\omega}_i$ is the rotational speed of each wheel; R is the equivalent rotational radius of the wheel; η is the transmission efficiency of the drivetrain system; κ is the transmission ratio of the drivetrain system; τ_i is the torque output from the motor to each wheel.

Considering the design of the subsequent controller, the longitudinal motion and yaw motion of the robot are decoupled. Since the wheels at the bottom of the skid-steering robot can be independently controlled, it is assumed that when the robot moves in a straight line, the output torque of the wheels on both sides is the same:

$$\tau_l=\tau_r=\tau_0 \tag{5}$$

When the robot is turning, the torque of the wheel on the outer side of the turning arc is τ_+, while the torque of the wheel on the inner side is τ_-, and the wheels on the same side output the same magnitude of torque. The torques of the wheels on both sides can be expressed as:

$$\begin{cases}\tau_{fl}=\tau_{rl}=\tau_+ +\tau_- \\ \tau_{fr}=\tau_{rr}=\tau_+ -\tau_-\end{cases} \tag{6}$$

For most skid-steering robots, their chassis structure usually coincides with the robot's longitudinal axis, i.e., $\gamma_l = \gamma_r = \gamma$. Substituting this expression into the previous equation, the robot's dynamic model can be expressed as:

$$M\dot{V}+C(v)V+D(v)V=E\tau+\tau_d$$

$$\begin{cases} M=\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & J_\omega \end{bmatrix} \\ C(v)=\begin{bmatrix} 0 & -m\omega & 0 \\ m\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ D(v)=\begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix} \end{cases} \quad (7)$$

The control input vector is defined as the left and right wheel driving motor torques $\tau=(\tau_l \quad \tau_r)^T$. When the robot moves in a straight line:

$$E=\begin{bmatrix} \frac{2\eta\kappa}{R} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

when turning:

$$E=\begin{bmatrix} \frac{\eta\kappa}{R} & \frac{\eta\kappa}{R} \\ 0 & 0 \\ \frac{\eta\kappa}{R} & -\frac{\eta\kappa}{R} \end{bmatrix}.$$

$\tau_d=\left[\tau_{xd},\ \tau_{yd},\ \tau_{\omega d}\right]^T$ represents the force and torque components exerted on the robot by external environmental disturbances. M denotes the inertia matrix, m is the mass of the robot, $C(v)$ is the centripetal force matrix, and $D(v)$ is the damping matrix. d_{11}, d_{22}, and d_{33} represent the damping coefficients acting on the agent in the lateral, longitudinal, and yaw directions, respectively.

$$\begin{cases} d_{11}=\frac{C_{roll}+0.5\rho C_d A v_x}{m} \\ d_{22}=\frac{C_\alpha}{m} \\ d_{33}=\frac{R^2 C_\alpha}{J_\omega} \end{cases} \quad (8)$$

Where C_{roll} is the tire rolling resistance coefficient, $\rho C_d A$ is the aerodynamic drag term, ρ is the air density, C_d is the wind resistance coefficient, and A is the frontal area; C_α is the tire cornering stiffness.

Environmental disturbance and measurement noise model

The operating environment of the robot often introduces process disturbances due to factors such as uneven ground surfaces, wind forces, and irregular terrain. These disturbances can significantly affect the robot's tracking accuracy and stability. Since it is difficult to model these disturbances precisely, they are represented using a zero-mean time-varying random variable with weight α_d to simulate environmental disturbances, expressed as follows:

$$\tau_d=\begin{bmatrix} \tau_{xd} \\ \tau_{yd} \\ \tau_{\omega d} \end{bmatrix}=\alpha_d \begin{bmatrix} rand(t) \\ rand(t) \\ rand(t) \end{bmatrix} \quad (9)$$

In the equation, rand(t) denotes a random number with a mean of 0 and a range between -1 and 1.

Considering the precision errors of measurement sensors, it is assumed that the disturbance L follows a Gaussian distribution, i.e., $L\sim N(0,\sum L)$. Here, $\sum L$ is the covariance matrix of the process noise, which describes the noise variance and the correlation between noise terms; the mean $\mu_L=0$ indicates that the expected value of the noise is zero, and it is generally assumed that the system has no external bias.

Upper-level HUKF-NMPC controller design

The upper-level controller is based on the combination of the H∞ filter with the unscented Kalman filter (HUKF) and nonlinear model predictive control (NMPC) algorithm. The controller consists of three parts: First, the robot's state is estimated in real time by combining the unscented Kalman filter with the H∞ filter, which helps handle sensor noise and system model errors. Second, trajectory optimization is performed based on the predictive model. Finally, the optimal control is solved by the NMPC controller τ_k.

Nonlinear state-space model

Based on the kinematic and dynamic models of the robot, a nonlinear continuous-time state-space model of the agent is established. The state vector is $q=[S \quad G]^T$; the control input vector is $\tau=(\tau_l \quad \tau_r)^T$; the output vector is $Y=[x, y, \theta]^T$; W and L denote the environmental disturbance and measurement noise, respectively. The state-space representation is given as:

$$\begin{cases} \dot{q}=f(q,\tau,W)=\begin{bmatrix} F(\theta)V \\ -M^{-1}(C(V)+D(V))+M^{-1}(\tau+\tau_d) \end{bmatrix} \\ Y=Aq+L \end{cases} \quad (10)$$

In the equation, A is a block diagonal matrix with $A=[I_{33} \quad 0_{33}]$, where I_{33} and 0_{33} denote the 3×3 identity matrix and the zero matrix, respectively.

Model discretization

Since the controller can only handle discrete-time data, the model needs to be discretized. Using the forward Euler method, the discretized model is obtained as:

$$\dot{q}=\frac{q_{k+1}-q_k}{t}=f(q_k,\tau_k,W_k) \quad (11)$$

$$\begin{cases} q_{k+1}=q_k+f(q_k,\tau_k,W_k)T \\ Y_k=Aq_k+L_k \end{cases} \quad (12)$$

where t is the sampling time; q_k is the discrete state vector at time step k; q_{k+1} is the discrete state vector at time step k+1; τ_k is the control input vector at time step k; W_k is the environmental disturbance at time step k; Y_k is the output vector at time step k; and L_k is the measurement noise at time step k.

Constraints and optimization

The nonlinear model predictive controller (NMPC) enables the robot to achieve an optimized desired motion trajectory. Within each control cycle, NMPC computes the optimal control input τ_k based on the current system state and the reference trajectory, such that the value of the designed cost function is minimized. By employing NMPC, the robot chassis can continuously adjust the control input to ensure accurate tracking of the reference trajectory and stable control of the system states.

In practical trajectory tracking tasks, the control inputs of the robot are executed by the actuators of the robot chassis. However, these actuators are subject to mechanical limitations, and the robot itself has inherent performance constraints. Therefore, the following categories of constraints need to be considered: control input constraints, state constraints, and disturbance range constraints.

$$\begin{cases} \tau_{min} \leqslant \tau_k \leqslant \tau_{max} \\ q_{min} \leqslant q_k + W_k \leqslant q_{max} \\ W_{min} \leqslant W_k \leqslant W_{max} \end{cases} \tag{13}$$

In the equation: τ_{min} and τ_{max} represent the upper and lower bounds of the input vector, respectively; q_{min} and q_{max} represent the upper and lower bounds of the sum of the state vector and the external disturbance vector, respectively; W_{min} and W_{max} represent the upper and lower bounds of the external disturbance vector, respectively. The control objective is:

$$\lim_{k\to\infty} \left\| \widetilde{Y}_k \right\|_2 \to 0 \tag{14}$$

In the equation: $\widetilde{Y}_k = Y_k - Y_{rk}$, where Y_{rk} is the estimated value at time step k, provided by the planning module; $\| \|_2$ denotes the Euclidean norm of a vector.

During the optimization process, we consider the worst-case scenario of disturbances, which means that the maximum or expected value of the disturbance needs to be introduced and incorporated into the optimization objective. The optimization problem is defined as $\min_{\tau_k} \max_{W_k} J$, where max_{W_k} represents solving for the worst-case disturbance and min_{τ_k} represents designing the optimal control strategy. The optimization objective is typically to minimize the deviation of the system states from the reference trajectory while also taking into account the cost of the control input to avoid excessively large inputs. The quadratic cost function of the NMPC is defined as J:

$$J = \sum_{i=1}^{N_P} \left\| \widetilde{Y_{k+i}} \right\|_Q^2 + \sum_{i=0}^{N_c-1} \| \tau_{k+i} \|_G^2 + \sum_{i=0}^{N_c-1} \beta \| W_{k+i} \|_U^2) \tag{15}$$

Q, G, and U are the weighting matrices for the output, input, and disturbance, respectively; β is the weighting coefficient of the disturbance; N_p and N_c denote the prediction horizon and the control horizon, respectively; i is an iteration variable used to traverse each time step during the summation process.

H∞ unscented kalman filter

In practical robotic trajectory tracking control, uncertainties such as modeling errors, sensor noise, and external disturbances often affect the accuracy of state estimation. To achieve stronger robustness and higher estimation precision, this paper proposes an H∞ unscented Kalman filter (HUKF) method by incorporating the H∞ robust filtering concept into the unscented Kalman filter (UKF) framework. HUKF employs the unscented transformation to directly generate sigma points from the original system model, thereby avoiding the linearization process and significantly improving the accuracy of nonlinear state estimation. To compute the optimal state estimate Y_k^+ of the robot at time step k under environmental disturbances, an H∞ gain correction is introduced. Specifically, during the update phase of the UKF, the H∞ robustness criterion is applied to optimize and adjust the Kalman gain, thereby enhancing the resistance to model uncertainties, noise, and external disturbances.

(1) Prediction step

First, based on the given state estimate $\hat{q}_k$ and covariance matrix P_k at time step k,2n+1 sigma points are generated through the unscented transformation, where n is the dimension of the state vector. Each sigma point $\xi_k^{(\sigma)}$ is:

$$\xi_k^{(\sigma)} = \left[\hat{q}_k, \ \hat{q}_k \pm \sqrt{(n+\lambda) P_k} \right] \tag{16}$$

where $\lambda = \varepsilon^2 (n+\kappa) - n$, ε is a scaling factor, and κ is a constant.

Each sigma point is propagated through the system's

nonlinear state transition function $f(\xi_k^{(\sigma)}, \tau_k)$ to obtain the predicted sigma points:

$$\xi_{k+1|k}^{(\sigma)}=f(\xi_k^{(\sigma)}, \tau_k) \quad (17)$$

By weighted averaging, the predicted state estimate $\hat{q}_{k|k+1}$ and covariance $P_{k|k+1}$ are obtained:

$$\hat{q}_{k|k+1}=\sum_{i=0}^{2n} W_m^{(\sigma)} \xi_{k+1|k}^{(\sigma)} \quad (18)$$

$$P_{k|k+1}=\sum_{i=0}^{2n} W_c^{(\sigma)} \left[\xi_{k+1|k}^{(\sigma)}-\hat{q}_{k|k+1} \right] \left[\xi_{k+1|k}^{(\sigma)}-\hat{q}_{k|k+1} \right]^T +Z \quad (19)$$

where $\hat{x}_{k|k+1}$ is the predicted mean state at time k+1, $P_{k|k+1}$ is the predicted covariance at time k+1, $W_m^{(\sigma)}$ and $W_c^{(\sigma)}$ are the mean and covariance weights of the sigma points, and Z is the process noise covariance matrix.

(2) Update step

Similarly, the predicted sigma points are mapped to the observation space through the observation model:

$$\rho_{k+1|k}^{(\sigma)}=A\, \xi_{k+1|k}^{(\sigma)} \quad (20)$$

where $\rho_{k+1|k}^{(\sigma)}$ represents the predicted observation of the σ-th sigma point at time step k+1.

By weighted averaging, the predicted observation $\hat{Y}_{k|k+1}$, observation covariance P_{yy}, and cross covariance P_{xy} between the state and observation are obtained:

$$\hat{Y}_{k|k+1}=\sum_{i=0}^{2n} W_m^{(\sigma)} \rho_{k+1|k}^{(\sigma)} \quad (21)$$

$$P_{yy}=\sum_{i=0}^{2n} W_c^{(\sigma)} \left[\rho_{k+1|k}^{(\sigma)}-\hat{Y}_{k|k+1} \right] \left[\rho_{k+1|k}^{(\sigma)}-\hat{Y}_{k|k+1} \right]^T +T \quad (22)$$

$$P_{xy}=\sum_{i=0}^{2n} W_c^{(\sigma)} \left[\xi_{k+1|k}^{(i)}-\hat{q}_{k|k+1} \right] \left[\xi_{k+1|k}^{(i)}-\hat{q}_{k|k+1} \right]^T +T \quad (23)$$

where $\hat{Y}_{k|k+1}$ is the observation mean at time step k+1; P_{yy} is the predicted observation covariance at time step k+1; $W_m^{(\sigma)}$ is the mean weight of the σ-th sigma point; $W_c^{(\sigma)}$ is the covariance weight of the σ-th sigma point; T is the observation noise covariance matrix; and P_{xy} is the cross covariance between the state and the observation at time step k+1.

Unlike the standard UKF, the H∞ filter does not aim to minimize the mean squared error but instead adopts the criterion of minimizing the worst-case effect of disturbances. This approach can effectively handle uncertainties caused by system noise, external disturbances, and model inaccuracies. By tolerating and compensating for model errors and disturbances, the accuracy of state estimation becomes less sensitive to such uncertainties. In the design of the Kalman gain K_k, an H∞ norm constraint is introduced, and the optimization problem is formulated as follows:

$$\sup_{q_k \neq 0} \frac{\|g_k\|_2^2}{\|W_k\|_2^2} < \phi \quad (24)$$

where $g_k=q_k-\hat{q}_k$ is the estimation error, W_k is the disturbance, and ϕ is the prescribed robust performance bound parameter (the smaller, the more conservative). The filter gain designed in this way not only adapts to system parameter variations and uncertainties, but also provides stronger suppression against external disturbances and measurement noise.

Based on the H∞ robustness criterion, the unscented Kalman gain can be expressed as:

$$K_k=P_{xy}\left(P_{xy}+\frac{1}{\epsilon^2} I\right)^{-1} \quad (25)$$

where ϵ is the H∞ filtering gain factor, which controls the suppression strength against noise.

By applying this gain, the predicted state is corrected as follows:

$$\hat{q}_{k+1}=\hat{q}_{k|k+1}+K_k(Y_{k+1}-\hat{Y}_{k+1|k}) \quad (26)$$

Where Y_{k+1} is the actual observation at time step k+1.

Meanwhile, the covariance matrix is updated as:

$$P_{k+1}=P_{k+1|k}-K_k P_{yy} K_k^T \quad (27)$$

The entire filtering process is executed recursively at each time step, continuously fusing observation data with system predictions to produce the optimal robust state estimate.

By introducing the H∞ constraint, the unscented Kalman filter is able to effectively enhance the reliability of state estimation and the overall robustness of the system when facing modeling errors, noise, and unknown disturbances. This method provides a solid foundation and guarantee for the precise implementation of the subsequent nonlinear model predictive control.

Compared with the traditional UKF, the HUKF demonstrates better robustness against abnormal observations and disturbances, and achieves smaller estimation errors in the presence of significant noise or model uncertainties.

As shown in the left panel of Figure 3, both filtering algorithms greatly suppress the effect of measurement noise, resulting in smoother trajectories that are closer to the ground truth. However, the HUKF estimates exhibit slightly better fitting with the true trajectory in certain curved and abrupt regions, with smaller fluctuations.

As illustrated in the right panel of Figure 3, the overall error for both algorithms remain within 0.2 m, with small deviations at most time steps. Nevertheless, the HUKF shows slightly smaller errors than the UKF for the majority of the time, and exhibits stronger noise-

suppression capability during periods of high disturbance. At local peak points, the error fluctuation of the HUKF also demonstrates a more pronounced convergence advantage. These results confirm the H∞ constraint's theoretical advantages and support the subsequent NMPC controller.

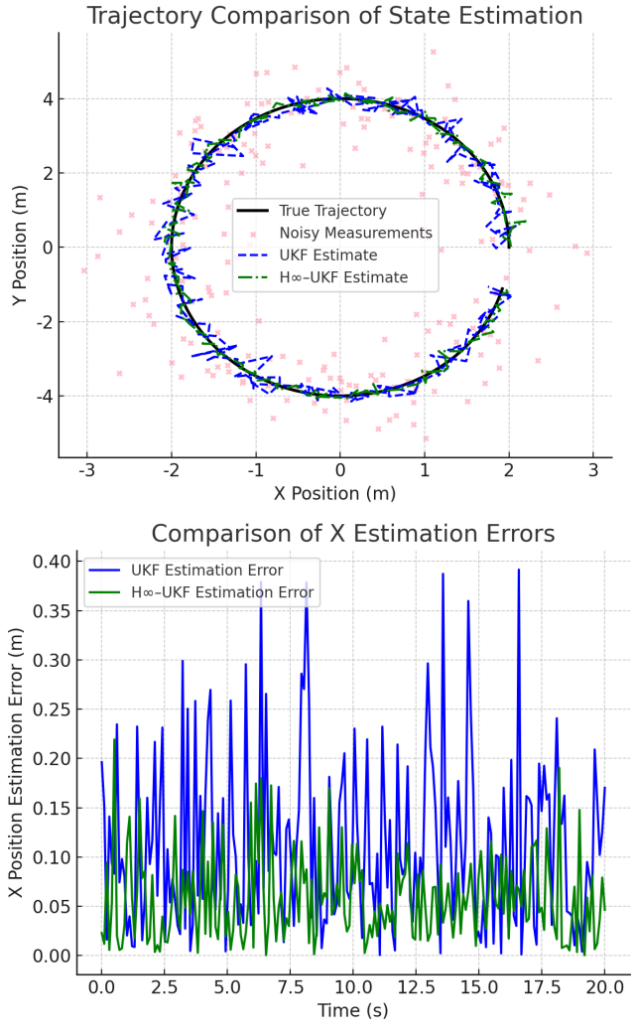


Figure 3. Comparison chart of UKF and HUKF state estimation with true values.

Design of the bottom-level synovial controller

The tracking velocity error vector of the robot is defined as $e=[e_{vx}, e_{vy}, e_{\omega}]^T$, and its error dynamics can be expressed as:

$$\begin{bmatrix} e_{vx} \\ e_{vy} \\ e_{\omega} \end{bmatrix} = \begin{bmatrix} v_{xk|ref}-v_{xk} \\ v_{yk|ref}-v_{yk} \\ \omega_{k|ref}-\omega_{k} \end{bmatrix} \tag{28}$$

where e_{vx}, e_{vy}, e_{ω} denote the errors of the robot's longitudinal velocity, lateral velocity, and yaw rate along the reference trajectory, respectively. Here, v_{xk} is the longitudinal velocity of the system at time step k, and $v_{xk|ref}$ is the reference longitudinal velocity at the same time step. The other variables are defined in the same manner.

To ensure that the system state error e_k converges to zero, a sliding surface is designed, such that the control system performs sliding motion along it. A second-order sliding surface is selected as:

$$s(t)=e-\eta\int_0^t e(\tau)\,d\tau \tag{29}$$

Where η is the sliding surface coefficient. From (29), it can be observed that when $\eta>0$ and $s(t)\to 0$, the tracking error $e\to 0$.Based on the concept of equivalent control, the control law is designed in a switching function form, such that the tracking error on the sliding surface converges to zero:

$$\dot{s(t)}=-\chi\cdot sign(s(t)) \tag{30}$$

where $\chi>0$ is the switching gain, which determines the control effort along the sliding surface to ensure that the system state performs sliding motion. The function sign($s(t)$) takes values in $\{-1,1\}$. When $s(t)>0$, the control signal is positive; otherwise, it is negative.By combining the tracking error at the robot's chassis level, the control torque outputs can be designed accordingly.

$$E\tau+\tau_d=M\begin{bmatrix} \dot{v_{xk|ref}}+\eta e_{vx}+-\chi\cdot sign(s(t)) \\ \dot{v_{yk|ref}}+\eta e_{vy}+-\chi\cdot sign(s(t)) \\ \dot{\omega_{k|ref}}+\eta e_{\omega}+-\chi\cdot sign(s(t)) \end{bmatrix}+C(v)\begin{bmatrix} v_{xk|ref}-e_{vx} \\ v_{yk|ref}-e_{vy} \\ \omega_{k|ref}-e_{\omega} \end{bmatrix}$$
$$+D(v)\begin{bmatrix} v_{xk|ref}-e_{vx} \\ v_{yk|ref}-e_{vy} \\ \omega_{k|ref}-e_{\omega} \end{bmatrix} \tag{31}$$

From the above expression, it can be observed that except for τ_d, all other terms are known. Substituting into the derivative form of the sliding surface yields:

$$\dot{s(t)}=-\chi\cdot sign(s(t))-\tau_d \tag{32}$$

To ensure that the system can reach the designed sliding surface, it is necessary to select an appropriate value of τ_d. As assumed earlier, τ_d is bounded and known, with $-1\leqslant\tau_d\leqslant 1$. Accordingly, the selection of τ_d must satisfy the following conditions:

When $s(t)>0$, to ensure $\dot{s(t)}<0$, set $\tau_d=-1$;

When $s(t)<0$, to ensure $\dot{s(t)}>0$, set $\tau_d=-1$.

Based on this logical relationship, τ_d can be expressed as: $\tau_d=1-sign(s(t))$.

Data simulation and analysis

To verify the effectiveness, robustness, and reliability of the proposed hierarchical controller, numerical simulations were conducted. The controller parameters were set as follows:

$T_s=h=0.05\ s, N_P=10, N_C=2, Q=diag(1,1,0.5)$, $G=diag(1,1,1), U=diag(1,1,1), \eta=1.2, \chi=0.82$,

α_d=0.5, q_0=$\left[-1 \quad 0 \quad \frac{\pi}{2} \quad 0 \quad 0 \quad \frac{\pi}{2}\right]^T$.

The input and state constraints were set as:

q_{min}=[-10 -10 -10 -2 -2 -2]^T, τ_{min}[-1 -1],

q_{max}=[10 10 10 2 2 2]^T, τ_{max}[1 1].

The simulation analysis was conducted to evaluate the tracking performance of the controller along a circular reference trajectory. The reference trajectory is defined as:

$$Y_r(t)=\begin{bmatrix} x_r(t) \\ y_r(t) \\ \phi_r(t) \end{bmatrix}=\begin{bmatrix} 4-4\cos(0.2t) \\ 4\sin(0.2t) \\ \frac{\pi}{2}-0.2t \end{bmatrix} \tag{33}$$

The corresponding simulation results are shown in Figure 4.

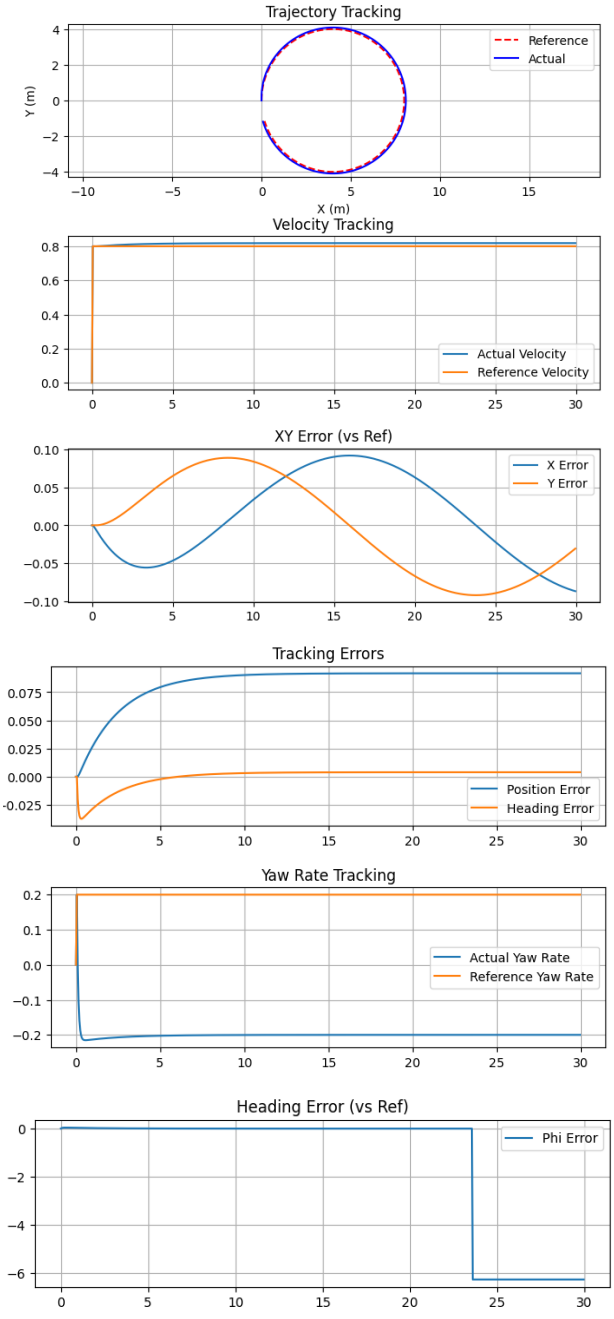


Figure 4. Simulation results of circular trajectory tracking.

From the simulation results, it can be observed that the actual trajectory of the robot closely follows the reference trajectory, indicating that the coordinated design of NMPC and the sliding mode hierarchical controller can achieve effective motion control of wheeled mobile robots. The detailed analysis is as follows:

In the trajectory tracking plot, the reference trajectory and the actual trajectory almost completely overlap, with only a slight deviation at the initial stage. As the simulation progresses, the system rapidly corrects the error and achieves stable tracking, demonstrating the strong trajectory tracking capability of the proposed controller.

The error curves show that both position errors and heading errors converge quickly at the beginning and remain within a very small range throughout the tracking process. The final steady-state errors are nearly negligible, confirming the high control accuracy of the system.

The comparison between the actual velocity and the reference velocity shows that the actual velocity converges rapidly to the reference value and then remains almost identical, reflecting the fast response and high-precision tracking performance of the velocity loop.

The angular velocity comparison indicates that the actual angular velocity is almost identical to the reference, verifying the high accuracy and robustness of the system in steering control.

The single-axis errors further illustrate that the position errors in both the X-axis and Y-axis remain very small, with the maximum absolute error less than 0.1 m and bounded oscillations. Similarly, the heading angle error stabilizes quickly around zero after the initial adjustment, without noticeable accumulated deviation.

Overall, these simulation results consistently demonstrate the controller's excellent tracking performance across position, velocity, and heading control. The system exhibits rapid convergence, minimal steady-state error, and strong robustness against disturbances. These characteristics are essential for practical deployment in complex environments. They

also indicate that the proposed method can maintain stable performance under varying operating conditions.

The simulation analysis verifies the effectiveness of the controller for circular trajectory tracking. The reference trajectory is defined as:

$$Y_r(t)=\begin{bmatrix} x_r(t) \\ y_r(t) \\ \phi_r(t) \end{bmatrix}=\begin{bmatrix} 0.75t \\ \cos(0.75t)+0.75t+1 \\ \arctan2(-0.75\sin(0.75t)+0.75,0.75) \end{bmatrix} \quad (34)$$

And the corresponding simulation results are illustrated in Figure 5.

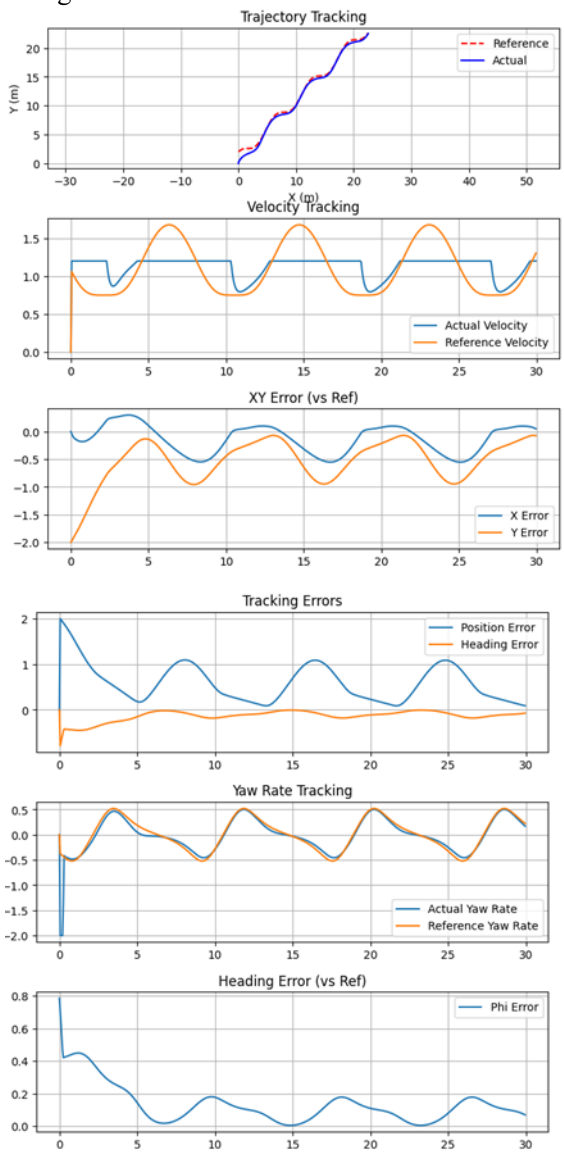


Figure 5. Simulation results of curved trajectory tracking for the bending curve.

From the simulation results, it can be concluded that the hierarchical controller achieves satisfactory tracking of curved and bent trajectories, with all system performance indices demonstrating high robustness and accuracy. This confirms the controller's capability to meet the high-precision tracking requirements of mobile robots operating in complex road environments.

In the trajectory tracking plot, the actual trajectory (blue line) follows the reference trajectory (red dashed line) very closely, realizing accurate tracking of the complex curved path. During the entire motion process, the robot is able to promptly adjust its posture at each turning point, thereby remaining tightly aligned with the predefined path without exhibiting significant deviations.

The tracking error plot illustrates the evolution of position and heading errors over time. It can be observed that the position error experiences slight fluctuations at the beginning of the trajectory and at points with high curvature variation, but overall remains at a low level, with the maximum position error not exceeding 2 m, and quickly converges thereafter. The heading error remains consistently small, indicating accurate attitude control without oscillations or accumulated deviations.

The middle two plot show the tracking performance of velocity and angular velocity. The actual velocity almost coincides with the reference velocity, with minimal oscillations, demonstrating that the controller responds promptly to velocity commands. Similarly, the angular velocity tracking is highly accurate, as the actual angular velocity nearly overlaps with the reference value, providing strong dynamic support during turning and curvature transition phases.

The XY error plot shows that the errors in both the X and Y directions remain within ±2 m and gradually decrease over time, indicating effective control in both lateral and longitudinal directions. The heading error comparison further demonstrates that the overall heading error remains small, with a maximum deviation of about 0.8 rad, and mostly within 0.4 rad, confirming the system's good convergence capability in heading control.

Conclusion

This paper addresses the trajectory tracking problem of four-wheeled skid-steering mobile robots by constructing a refined model that incorporates both dynamics and environmental disturbances, and proposes a hierarchical control strategy based on HUKF-NMPC and sliding mode control. The method fully considers multiple constraints, including control inputs, state variables, and external disturbances, and integrates state estimation, trajectory optimization, and robust actuation in a unified framework.

Simulation results demonstrate that the proposed controller achieves high-precision and robust trajectory tracking under various reference trajectories (e.g., circular and curved paths), while effectively resisting the influence of environmental disturbances and modeling uncertainties. The tracking errors converge rapidly and remain within a very small range, while both velocity and angular velocity exhibit excellent tracking performance. The findings provide new insights for the intelligent control of nursing robots in practical engineering applications and highlight promising application prospects. Future work will focus on optimizing the real-time performance of the algorithm and extending the framework to cooperative robust trajectory tracking in multi-robot systems.

Funding

This work was supported in part by the Key research and development (R&D) Program of Shandong Province, China under Grant Nos: 2024CXGC010809, 2024XDRH109, 2025TSGCCZZB0597, 2025TSGCCZZB0365, 2025TSGCCZZB0390, 2025TSGCCZZB0369, and 2025TSGCCZZB0756.

Acknowledgements

The authors would like to show sincere thanks to those technicians who have contributed to this research.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] Lee, Y. S., Iizuka, T., Eggleston, K. (2025) Robots and labor in nursing homes. *Labour Economics*, 92, 102666.

[2] Shaw, R. J., Chen, B. (2025) Physical artificial intelligence in nursing: robotics. *Nursing Outlook*, 73, 102495.

[3] Gong, H., Yan, T., Xie, H., Gao, R., Zhan, Y., Xia, Y. (2025) Learning stabilizable symplectic ODE-net-based MPC for autonomous vehicle trajectory tracking. *Neurocomputing*, 131222.

[4] Wang, L., Chen, Z., Zhu, W. (2022) An improved pure pursuit path tracking control method based on heading error rate. *Industrial Robot: The International Journal of Robotics Research and Application*, 49(5), 973-980.

[5] Li, Y., Jiang, Y. Q., Wang, L. F., Cao, J., Zhang, G. C. (2015) Intelligent PID guidance control for AUV path tracking. *Journal of Central South University*, 22(9), 3440-3449.

[6] Luo, X., Zhu, M., Zhang, Y., Zheng, Z., Chen, T. (2024) Self-triggered fuzzy trajectory tracking control for the stratospheric airship. *Advances in Space Research*, 74(11), 5874-5889.

[7] Deng, Z., Cui, J., Zhu, Z., Dan, Y., Xie, H., Hu, Y. (2025) Adaptive trajectory tracking control of underactuated ships using parameter prediction based neural network. *Ocean Engineering*, 329, 121188.

[8] Hua, Y., Ji, Z., Wang, Y. (2025) Robust trajectory tracking control for multiple mobile robots. *Journal of Intelligent & Robotic Systems*, 111, 79.

[9] Xu, W., Liu, N., Du, S., Zhou, H. (2025) Adaptive trajectory tracking control based on path evaluation model: W. Xu et al. *International Journal of Intelligent Robotics and Applications*, 9(4), 1692-1715.

[10] Li, W., Zhou, H. (2025) Constrained nonlinear MPC with rudder-roll stabilization for integrated path following and collision avoidance in underactuated surface vessels. *Journal of Marine Science and Engineering*, 13(3), 468.

[11] Rodriguez-Arellano, J. A., Miranda-Colorado, R., Aguilar, L. T., Negrete-Villanueva, M. A. (2023) Trajectory tracking nonlinear H∞ controller for wheeled mobile robots with disturbances observer. *ISA Transactions*, 142, 372-385.

[12] Zhao, G., Chen, Z., Liao, W. (2024) Reinforcement-tracking: an end-to-end trajectory tracking method based on self-attention mechanism. *International Journal of Automotive Technology*, 25(3), 541-551.

[13] Li, W., Yu, S., Lin, B., Chen, H. (2025) Nonlinear model predictive control of trajectory tracking for autonomous trucks with terminal ingredients. *Nonlinear Dynamics*, 113, 21383-21402.

[14] Lai, G., Zhuang, B., Ouyang, Y., Tang, J. (2026) EKF-based MPC scheme for image-based visual servoing of UAVs subject to multiple uncertain disturbances. *Nonlinear Dynamics*, 114(9), 691.

[15] Cheng, Z., Xie, W., Li, Z., Ye, Z., Tang, G. (2025) Trajectory tracking and collision avoidance for unmanned surface vehicle swarms via model

predictive control. *IFAC-Papers OnLine*, 59(22), 555-560.

[16] Vu, T. M., Moezzi, R., Cyrus, J., Hlava, J. (2021) Model predictive control for autonomous driving vehicles. *Electronics*, 10(21), 2593.

[17] Feng, H., Jing, H., Zhang, X., Kuang, B., Song, Y., Wei, C., Qian, T. (2025) Research on the stability and trajectory tracking control of a compound steering platform based on hierarchical theory. *Electronics*, 14(14), 2836.

[18] García-Chávez, R. E., Silva-Ortigoza, R., Hernández-Guzmán, V. M., Marciano-Melchor, M., Orta-Quintana, Á. A., García-Sánchez, J. R., Taud, H. (2023) A robust sliding mode and PI-based tracking control for the MIMO "DC/DC buck converter-inverter-DC motor" system. *IEEE Access*, 11, 119396-119408.

[19] Yu, L. S., Su, X. J., Sun, S. X., Jiao, C. T. (2023) Lateral stability control of six-wheeled skid-steering robot based on hierarchical control strategy. *Acta Automatica Sinica*, 49(7), 1421-1432.

[20] Juman, M. A., Wong, Y. W., Rajkumar, R. K., Kow, K. W., Yap, Z. W. (2019) An incremental unsupervised learning based trajectory controller for a 4 wheeled skid steer mobile robot. *Engineering Applications of Artificial Intelligence*, 85, 385-392.