Journal of the Chinese Society of Mechanical Engineers, Vol.46, No.6, pp657~668 (2025)

Research on Uncertainty based Adaptive Model Predictive Control for Autonomous Vehicles under High-speed Operating Conditions

Jia-Cheng Mai*, Yi-Fan Zhao**,Fei Liu*,Xiaofeng Weng*,Sheng Zhou*and Shaoxiang Feng*

Keywords：Model predictive control, Uncertainty, Multicellular model, Trajectory tracking.

ABSTRACT

This study presents an uncertainty model predictive control (UM-MPC) algorithm that considers the uncertainties inherent in both environmental conditions and model parameters. Its core aim is to bolster the tracking accuracy and control system stability of autonomous vehicles. Establishing a multi-cellular model to mitigate the impact of uncertain parameters in the model. Based on the characteristic that drivers adjust their attention concentration according to road changes, a dynamic rule for the weight matrix has been designed through a large amount of comparative experimental data, achieving a shift in the focus of the algorithm during rolling optimization. Additionally, an adaptive predictive adjustment function for weights is proposed, and the optimal solution is derived through offline analytical optimization and the improved Particle Swarm Optimization (PSO) algorithm. Through a Hardware-in-the-Loop platform, a comparative analysis was conducted with the Adaptive Model Predictive Control (AMPC) based on fuzzy rules, affirming the effectiveness of the algorithm.

INTRODUCTION

As a revolutionary innovation, autonomous driving technology is rapidly changing the face of modern transportation systems. Autonomous vehicles play a vital role in improving ride comfort, reducing energy consumption, achieving higher road safety, efficiency.(Nie *et al.*, 2022). The rapid development of autonomous driving technology has garnered extensive attention from society, enterprises, and universities. However, one of the pivotal technologies essential for achieving safe and efficient autonomous driving is trajectory tracking.(Li *et al.*, 2017; Nan *et al.*, 2017).The accuracy, robustness and real-time performance of the trajectory tracking algorithm are very important for the accurate control and safe driving of the vehicle. The tracking effect directly affects the safety and comfort of drivers and passengers. (Xia *et al.*, 2016). The primary function of the trajectory tracking algorithm is to use the path planned by the upper-level modules as a reference. Sensors generate the target steering angle based on the reference path data, and then transmit this target angle to the lower-level controller, which controls the vehicle while ensuring tracking accuracy and stability. (Peza-Solis *et al.*, 2022).

In the past few decades, a variety of trajectory tracking algorithms have been proposed and studied. Trajectory tracking control algorithms commonly applied in autonomous vehicles include: Proportional-Integral-Derivative Control (PID), Model Predictive Control (MPC), Optimal Control, Fuzzy Logic control and Kalman Filter. (Geng and Liu, 2020; Koga *et al.*, 2016). These prevalent control strategies each have their strengths and specific applications, with the ultimate goal of achieving high-quality tracking performance for vehicles.

Zhou et al(Zhou *et al.*, 2019) designed the kinematic trajectory tracking controller and the electromechanical coupling dynamics trajectory tracking controller based on MPC, and a set of test devices is proposed to verify the tracking performance of the algorithm. However, the test scene is relatively specific and cannot fully cover the real scene; Xu et al(Xu and Peng, 2020) employs a control method that combines feedforward control for road curvature handling with feedback control for response error correction to enhance the tracking performance and computational efficiency of the vehicle. However, the applicability of its linear model at high speeds or in

Paper Received November, 2024. Revised January, 2025. Accepted January, 2025. Author for Correspondence: Yi-Fan Zhao.

* *School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, 333 Long Teng Road, Shanghai, 201620, P.R. China.*

** *School of Mechanical and Equipment Engineering, Hebei University of Engineering Handan,Hebei,China,Postal Code:056038.*

complex environments remains to be considered; Hang et al(Hang and Chen, 2021) established a linear parameter-varying system model, taking into account the road's coefficient of friction and longitudinal velocity, and has adopted a control method that combines feedforward control with secondary regulation control to eliminate errors caused by disturbances. However, real road environment is very complex, and considering only the coefficient of friction is insufficient.

In actual tracking control, due to the variability of road environments, traditional control systems are often subjected to a range of uncertain and complex factors such as external noise, disturbances, and environmental changes. These factors can cause uncertainties in the control system model and variations in parameters, leading to a reduction in accuracy and robustness.(Gao *et al.*, 2023; Li *et al.*, 2019; Wei *et al.*, 2023). Adaptive control can monitor and identify the dynamic characteristics, model and environmental changes of the system in real time, and automatically adapt to these changes by adjusting the parameters in real time. Compared with the traditional control strategy with fixed parameters in advance, adaptive control can better guarantee the tracking performance of the controller in the face of complex and changeable external disturbances. (Qu *et al.*, 2020).

In order to reduce the impact of model uncertainty. Chen et al(Chen and Chen, 2023), propose a nonlinear adaptive fuzzy control method. By eliminating the linearization error of the model, the accuracy and stability are improved, and a good tracking effect is obtained; Hu et al(Hu and Cheng, 2023) used the hybrid A-star algorithm and the minimum capture smoothing method for trajectory planning, and transformed the motion control formula into an adaptive model predictive control through a linear parameter-varying kinematic model, thereby reducing the tracking error of the lateral position and azimuth; In order to adapt to the complex changes of the environment, Yang et al. (Yang *et al.*, 2023) designed an event-triggered model predictive control with adaptive artificial potential field. The adaptive artificial potential field function is added to the original objective function to improve the obstacle avoidance effect of the vehicle and ensure the stability of the system to a certain extent; Luan et al. (Luan *et al.*, 2020) considered the steering angle oscillation caused by multiple step changes in the target angle actually received by the system node due to the random network delay of the sensor and actuator and the limitation of the CAN bus sampling mechanism of the steer-by-wire system. An adaptive model predictive control of linear uncertain model is proposed. The PSO algorithm is used to calculate the predictive time domain and the control time domain offline to find the time domain solution to achieve the optimal control effect.

To effectively address the interference or uncertainties present in the system's operating environment, it is necessary to further enhance the system's robustness. Mashadi et al. (Mashadi *et al.*, 2015) designed a μ-synthesis robust controller by combining four-wheel steering (4WS) with direct yaw moment control (DYC). Zhang et al. (Zhang *et al.*, 2021) proposed a novel robust event-triggered fault-tolerant control strategy for automatic steering systems. This strategy uses polyhedral set reduction and norm-bounded uncertainty attenuation to effectively handle uncertainties associated with time-varying velocities and tire cornering stiffness in the dynamic model, thereby reducing the impact of modeling errors.

Facing the impact of external disturbances and internal uncertainties, this paper draws inspiration from the aforementioned scholars' research. Starting from the underlying logic, an in-depth study of the rolling optimization in the key part of MPC is conducted, exploring the pattern of changes between the prediction horizon and the weight matrix at each step of the optimization process. At the same time, considering the variation of a series of uncertain factors such as the time-varying parameters of the model and the lateral stiffness, a polyhedral model is established. Based on an innovative adaptive control strategy, a method of Model Predictive Control that accounts for uncertainty (UM-MPC) is proposed. The weight matrix in the optimization process is improved, and an adaptive predictive level function for weight variation is proposed according to curvature, speed, lateral stiffness, slip rate, and temporal changes, and the optimal values are obtained through offline optimization and an improved Particle Swarm Optimization (PSO) algorithm to achieve good tracking performance. The overall framework of the controller is shown in Figure 1.

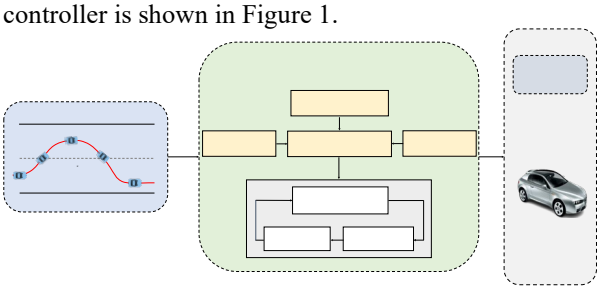

Figure 1. Architecture of control system

The main contributions of this paper are as follows:

(1) By comparing the tracking results, the law of the weight matrix changing with the rolling optimization step size is obtained, and the selection strategy of the adaptive weight matrix according to different working conditions is proposed.

(2) Considering the influence of dynamic uncertain parameters, speed, lateral stiffness, slip rate and other changes on the vehicle model, an adaptive predictive level function for weight variation is

proposed. This function adapts to the variations and utilizes an enhanced Particle Swarm Optimization (PSO) algorithm to determine the optimal coefficients that minimize tracking error under the current operating conditions. This approach differs from traditional weight adaptation based on fuzzy logic, as it improves tracking precision and stability comprehensively without increasing computational load or affecting the system's real-time capabilities.

ESTABLISHMENT OF VEHICLE DYNAMICS MODEL

In order to ensure that the motion behavior and driving mechanical characteristics of the vehicle can be described more accurately, the complexity of the controller design is reduced and the amount of calculation is reduced. In this paper, a widely used three-degree-of-freedom vehicle dynamics model (Yuan et al., 2019) is adopted and combined with the error model diagram. As shown in Figure2.

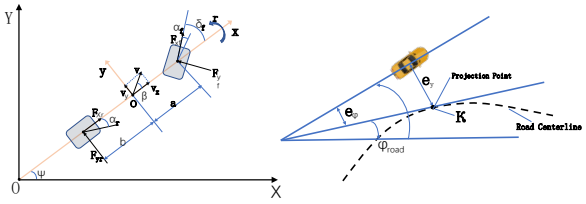

Fig. 2. Vehicle Three Degree of Freedom Dynamic Model and Tracking Error Model

The vehicle yaw dynamics equation is as follows given the presumptions:

$$\begin{cases} m(\ddot{x} - \dot{y}\dot{\varphi}) = F_{xf}\cos\delta_f - F_{yf}\sin\delta_f + F_{xr} \\ m(\ddot{y} + \dot{x}\dot{\varphi}) = F_{xf}\sin\delta_f + F_{yf}\cos\delta_f + F_{yr} \\ I_z\ddot{\varphi} = a(F_{yf}\cos\delta + F_{xf}\sin\delta) - bF_{yr} \end{cases} \quad (1)$$

where control quantity δ_f is the front wheel angle of the vehicle; κ is the curvature of the road; φ is the yaw angle of the vehicle body; e_y and e_φ are the lateral position and heading errors, respectively; β is the vehicle sideslip angle; $\dot{y}$ and $\dot{x}$ are the vehicle's longitudinal and lateral speeds; I_z represents the vehicle's moment of inertia around the z-axis. F_{ij} is tire force.

When calculating tire forces, it is possible to simplify the tire model to a linear one by assuming that the slip ratio of the wheel and the sideslip angle of the tire are both within a small range, while neglecting the effects of load transfer on the tire's sideslip stiffness and the wheel's longitudinal stiffness, thereby obtaining linear expressions for the tire's longitudinal force and lateral force.

$$F_l = \bar{C}\alpha \quad (2)$$

Among them, α represents the tire slip angle, and $\bar{C}$ represents the linear cornering stiffness of the tire. When the lateral acceleration ay is less than or equal to 0.4g and the tire sideslip angle is less than or equal to 6°, the expressions for the tire's lateral force and longitudinal force are as follows:

$$\begin{cases} F_x = C_x(\mu, F_z)s_{f,r} \\ F_y = C_\alpha(\mu, F_z)\alpha \end{cases} \quad (3)$$

The tire longitudinal force F_l of the linearized model is a linear function of the wheel longitudinal stiffness C_x and the longitudinal slip ratio s.

Based on the small angle assumption, the expressions for the tire slip angle and the vehicle sideslip angle are as follows:

$$\alpha_f = \arctan\left(\frac{\dot{y}+a\dot{\varphi}}{\dot{x}}\right) - \delta_f \approx \beta + \frac{a\dot{\varphi}}{\dot{x}} - \delta_f \quad (4)$$

$$\alpha_r = \arctan\left(\frac{\dot{y}-b\dot{\varphi}}{\dot{x}}\right) \approx \beta - \frac{b\dot{\varphi}}{\dot{x}} \quad (5)$$

$$\beta = \arctan\left(\frac{\dot{y}}{\dot{x}}\right) \approx \frac{\dot{y}}{\dot{x}} \quad (6)$$

where α_f and α_r are the front and rear tire sideslip angles respectively

When taking into account the road curvature, the lateral tracking error is significantly smaller than the error when road curvature is not considered. The vehicle yaw dynamics model incorporates the varying road curvature to develop a tracking error model.

$$\begin{cases} \dot{e}_y = \dot{x}\sin e_\varphi + \dot{y}\cos e_\varphi \approx \dot{x}e_\varphi + \dot{y} \\ \dot{e}_\varphi = \dot{\varphi} - \frac{\kappa\dot{x}\cos e_\varphi}{1-\kappa e_y} \approx \dot{\varphi} - \kappa\dot{x} \end{cases} \quad (7)$$

where e_y and e_φ are the lateral position error and heading angle error of the vehicle respectively. The 3-DOF yaw dynamics tracking error model is：

$$\begin{cases} \ddot{x} = \dot{y}\dot{\varphi} + \frac{s_f C_{lf} + s_r C_{lr}}{m} - \frac{C_{\alpha f}(\beta + \frac{l_f\dot{\varphi}}{\dot{x}} - \delta_f)\delta_f}{m} \\ \ddot{y} = -\dot{x}\dot{\varphi} + \frac{C_{\alpha f}(\frac{\dot{y}+a\dot{\varphi}}{\dot{x}} - \delta_f)}{m} + \frac{C_{\alpha r}(\frac{\dot{y}-b\dot{\varphi}}{\dot{x}})}{m} \\ \ddot{\varphi} = \frac{(l_f C_{\alpha f}(\frac{\dot{y}+a\dot{\varphi}}{\dot{x}} - \delta_f) - l_r C_{\alpha r}(\frac{\dot{y}-b\dot{\varphi}}{\dot{x}}))}{I_z} \\ \dot{e}_\varphi = \dot{\varphi} - \kappa\dot{x} \\ \dot{e}_y = \dot{x}e_\varphi + \dot{y} \end{cases} \quad (8)$$

Basic parameters of the vehicle: m、$C_{\alpha f}/C_{\alpha r}$、$C_{lf}/C_{lr}$ and l_f/l_r represent the mass of the vehicle, the cornering stiffness and longitudinal stiffness of the front and rear tires, and the distance from the center of mass to the front and rear wheels, respectively.

The nonlinear model often has high accuracy, but it will greatly increase the complexity of the controller setting, so it is necessary to linearize the vehicle dynamics model.

The state variable of the vehicle is $\xi = [\dot{y}, \dot{x}, \varphi, \dot{\varphi}, e_y, e_\varphi]^T$. The control input vector is $u = \delta_q$, additional input $\lambda = [\kappa]^T$, the state space form is $\dot{\xi} = f(\xi, u, \lambda)$. By linearizing the nonlinear dynamic model(Raffo *et al.*, 2009), and discretize the system state to obtain:

$$\begin{aligned} \xi(k+1) &= \xi(k) + T \cdot [f(\xi(k), u_1(k), \lambda(k)] \\ &= F\big(\xi(k), u_1(k), \lambda(k)\big) \end{aligned} \quad (9)$$

where k is the current sampling time point, k+1 is the next sampling time point, and T is the sampling

time.

Performing Taylor expansion at the operating point (ξ_o, u_o, λ_o), and the first-order term is retained:

$$\xi(k+1) = F(\dot{\xi}_o(k), u_o(k), \lambda_o(k)) + A_{k,o}(\dot{\xi}(k) - \hat{\dot{\xi}}_o(k)) + B_{1_{k,o}}(u(k) - u_o) + B_{2_{k,o}}(\lambda(k) - \lambda_o) \quad (10)$$

Subtract the state variable of the working point from the above formula to obtain the linear time-varying system equation of the vehicle after discrete linearization:

$$\xi(k+1) = A_{k,o}\dot{\xi}(k) + B_{1_{k,o}}u(k) + B_{2_{k,o}}\lambda(k) + D_{k,o} \quad (11)$$

where $D_{k,t} = \hat{\xi}_t(k+1) - \left(A_{k,t}\hat{\xi}_t(k) + B_{1_{k,t}}u_t + B_{2_{k,t}}\lambda_t\right)$, $A_{k,t}$, $B_{1_{k,t}}$ and $B_{2_{k,t}}$ are all linear time-varying Jacobian matrices:

$$A_{k,t} = \begin{bmatrix} \frac{(C_{\alpha f}+C_{\alpha r})}{m\dot{x}} & -\frac{C_{\alpha f}(\dot{y}+a\dot{\varphi})+C_{\alpha r}(\dot{y}-b\dot{\varphi})}{m\dot{x}^2} - \dot{\varphi} & 0 & \frac{(aC_{\alpha f}-bC_{\alpha r})}{m\dot{x}} - \dot{x} & 0 & 0 \\ -\frac{C_{\alpha f}\delta_f}{m\dot{x}} + \dot{\varphi} & -\frac{C_{\alpha f}(\dot{y}+a\dot{\varphi})\delta_f}{m\dot{x}^2} & 0 & -\frac{aC_{\alpha f}\delta_f}{m\dot{x}} + \dot{y} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{aC_{\alpha f}-bC_{\alpha r}}{I_z\dot{x}} & -\frac{aC_{\alpha f}(\dot{y}+l_f\dot{\varphi})-bC_{\alpha r}(\dot{y}-l_r\dot{\varphi})}{I_z\dot{x}^2} & 0 & \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z\dot{x}} & 0 & 0 \\ 1 & e_\varphi & 0 & 0 & 0 & \dot{x} \\ 0 & -\kappa & 0 & 1 & 0 & 0 \end{bmatrix} \quad (12)$$

$$B_{1_{k,t}} = \left[-\frac{C_{\alpha f}}{m} \quad \frac{C_{\alpha f}(2\delta_f - \frac{(\dot{y}+a\dot{\varphi})}{\dot{x}})}{m} \quad 0 \quad -\frac{aC_{\alpha f}}{I_z} \quad 0 \quad 0\right]^T \quad (13)$$

$$B_{2_{k,t}} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -\dot{x}]^T. \quad (14)$$

Define new state variable $\zeta(k) = \begin{bmatrix} \xi(k) \\ u_1(k-1) \\ u_2(k) \end{bmatrix}$, output state variable $\eta(k)$ and increment $\Delta u(k) = u(k) - u(k-1)$ of control input. The discrete state space controller model can be transformed into a new form:

$$\zeta(k+1|t) = \tilde{A}_{k,t}\zeta(k|t) + \tilde{B}_{k,t}\Delta u(k|t) + \tilde{D}_{k,t} \quad (15)$$

$$\eta(k|t) = H_p\zeta(k|t) \quad (16)$$

where $\tilde{A}_{k,t} = \begin{bmatrix} A_{k,t} & B_{1_{k,t}} & B_{2_{k,t}} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $\tilde{B}_{k,t} = \begin{bmatrix} B_{1_k} \\ 1 \\ 0 \end{bmatrix}$, $_{k,t} = \begin{bmatrix} D_{k,t} \\ 0 \\ 0 \end{bmatrix}$, $H_p = [1 \quad 0 \quad 0]$。

After obtaining the new state variable, iterate continuously in the following time series to obtain the prediction output expression in the prediction horizon:

$$Y(t) = \Psi_t\zeta(t) + \Theta_t\Delta U(t) + \Lambda_t\Upsilon(t) \quad (17)$$

where $Y(t) = [\eta(k+1|t) \quad \eta(k+2|t) \quad \cdots \quad \eta(k+N_c|t) \quad \cdots \quad \eta(k+N_p|t)]^T$,

$\Psi_t = \left[H_p\tilde{A}_{k,t} \quad H_p\tilde{A}_{k,t}^2 \quad \cdots \quad H_p\tilde{A}_{k,t}^{N_c} \quad \cdots \quad H_p\tilde{A}_{k,t}^{N_p}\right]^T$,

$\Delta U(t) = [\Delta u(k|t) \quad \Delta u(k+1|t) \quad \Delta u(k+2|t) \quad \cdots \quad \cdots \quad \Delta u(k+N_c-1|t)]^T$

$\Upsilon(t) = \left[\tilde{D}_{k,t}(k) \quad \tilde{D}_{k,t}(k+1) \quad \tilde{D}_{k,t}(k+2) \quad \cdots \quad \cdots \quad \tilde{D}_{k,t}(k+N_p-1)\right]^T$,

$$\Theta_t = \begin{bmatrix} H_p\tilde{B}_{k,t} & 0 & 0 & 0 \\ H_p\tilde{A}_{k,t}\tilde{B}_{k,t} & H_p\tilde{B}_{k,t} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ H_p\tilde{A}_{k,t}^{N_c-1}\tilde{B}_{k,t} & H_p\tilde{A}_{k,t}^{N_c-2}\tilde{B}_{k,t} & \cdots & H_p\tilde{B}_{k,t} \\ \vdots & \vdots & \ddots & \vdots \\ H_p\tilde{A}_{k,t}^{N_p-1}\tilde{B}_{k,t} & H_p\tilde{A}_{k,t}^{N_p-2}\tilde{B}_{k,t} & \cdots & H_p\tilde{A}_{k,t}^{N_p-N_c}\tilde{B}_{k,t} \end{bmatrix},$$

$$\Lambda_t = \begin{bmatrix} H_p & 0 & 0 & 0 \\ H_p\tilde{A}_{k,t} & H_p & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ H_p\tilde{A}_{k,t}^{N_c-1} & H_p\tilde{A}_{k,t}^{N_c-2} & \cdots & H_p \\ \vdots & \vdots & \ddots & \vdots \\ H_p\tilde{A}_{k,t}^{N_p-1} & H_p\tilde{A}_{k,t}^{N_p-2} & \cdots & H_p\tilde{A}_{k,t}^{N_P-N_p} \end{bmatrix}。$$

where N_p is the prediction horizon, and N_c is the control horizon.

Establishment of Polytope Model

The autonomous vehicle will not only encounter the uncertainty of the environment, but also be affected by the uncertain parameters of its own model. In the case of increasing speed, the impact of model uncertainty will also intensify. When the vehicle is traveling at high speed, the presence of time-variant $\frac{1}{\dot{x}}$ parameters in the model and the strong non-linear change of the tires can render the entire control system highly unstable and extremely susceptible to influences from system inputs and the external environment. As a result, the vehicle is unable to guarantee great tracking performance, and the constructed model fails to accurately express the dynamic characteristics of the vehicle.

To address this issue, this paper proposes to refine the three-degree-of-freedom vehicle model into a polyhedral model. Taking into account the impact of the vehicle's uncertain parameters: $\left[\frac{1}{\dot{x}} \quad C_{\alpha f} \quad C_{\alpha r}\right]$, the number of vertices of the convex polytope model is determined to be 2^3, where the uncertain parameter ranges are respectively:

$$\begin{aligned} \frac{1}{\dot{x}} &\in \left[\frac{1}{\dot{x}_{max}}, \frac{1}{\dot{x}_{min}}\right] \\ C_{\alpha f} &\in \left[C_{\alpha f,min} \quad , C_{\alpha f,max} \quad\right] \\ C_{\alpha r} &\in \left[C_{\alpha r,min} \quad , C_{\alpha r,max} \quad\right] \end{aligned} \quad (18)$$

According to the range of uncertain parameters, the vertex parameters of the polytope can be determined as:

$$\begin{aligned} 1/v_x &= \sum_{2}^{i=1} \bar{\gamma}_i(t)\bar{\eta}_i \\ C_{\alpha f} &= \sum_{2}^{i=1} \tilde{\gamma}_i(t)\tilde{\eta}_i \\ C_{\alpha r} &= \sum_{2}^{i=1} \hat{\gamma}_i(t)\hat{\eta}_i \end{aligned} \quad (19)$$

Make the following definition:

$$\gamma_i = \bar{\gamma}_j\tilde{\gamma}_k\hat{\gamma}_l \quad (j = 1,2, k = 1,2, l = 1,2) \quad (20)$$

$$\sum_{i=1}^{8}\gamma_i = 1 \quad (21)$$

$$\begin{aligned} \bar{\gamma}_1(t) &= \frac{1/v_{x,min} - 1/v_x}{1/v_{x,min}}, \bar{\gamma}_2(t) = \frac{1/v_x - 1/v_{x,max}}{1/v_{x,max}} \\ \tilde{\gamma}_1(t) &= \frac{C_{\alpha f,max} - C_{\alpha f}}{C_{\alpha f,max}}, \ \tilde{\gamma}_2(t) = \frac{C_{\alpha f} - C_{\alpha f,min}}{C_{\alpha f,min}} \\ \hat{\gamma}_1(t) &= \frac{C_{\alpha r,max} - C_{\alpha f}}{C_{\alpha r,max}}, \ \hat{\gamma}_2(t) = \frac{C_{\alpha r} - C_{\alpha r,min}}{C_{\alpha r,min}} \end{aligned} \quad (22)$$

where $\bar{\gamma}, \tilde{\gamma}, \hat{\gamma}$ is the correction coefficient. Replace a in the original state space equation with $\frac{1}{\dot{x}}, C_{\alpha f}, C_{\alpha r}$ time-varying variable $\bar{\eta}, \tilde{\eta}, \hat{\eta}$. The state space parameter matrix at the ith vertex of the convex polytope is A_i, B_i, D_i ($i = 2^3$). The forward Euler method is used for discretization.

$$\tilde{X}(k+1) = \tilde{A}\tilde{X}(k) + \tilde{B}\tilde{u}(k) + \tilde{D}\lambda(k) \quad (23)$$

$$\tilde{A} = \sum_{8}^{i=1}\gamma_i A_{d,i} \quad \tilde{B} = \sum_{8}^{i=1}\gamma_i B_{d,i} \quad \tilde{C} = \sum_{8}^{i=1}\gamma_i C_{d,i} \quad (24)$$

$$[\tilde{A} \quad \tilde{B} \quad \tilde{D}] = \sum_{8}^{i=1}\gamma_i[A_{d,i} \quad B_{d,i} \quad D_{d,i}]$$

$$[\tilde{A} \quad \tilde{B} \quad \tilde{D}] \in \Omega = O_0\{(A_{d,1}, B_{d,1}, D_{d,1}), (A_{d,2}, B_{d,2}, D_{d,2}), \cdots, (A_{d,i}, B_{d,i}, D_{d,i})\} \quad (25)$$

where γ_i is a nonnegative constant.

Using the method of controlling increment, a new state space equation is established as follows:

$$\xi(k) = \begin{bmatrix} \tilde{X}(k) \\ \tilde{u}(k-1) \end{bmatrix} \tag{26}$$
$$\xi(k+1) = \tilde{A}_k \xi(k) + \tilde{B}_k \Delta\tilde{u}(k) + \tilde{D}_k \lambda(k)$$

where: $\tilde{A}_k = \begin{bmatrix} \tilde{A} & \tilde{B} \\ 0_{j\times k} & I_j \end{bmatrix}$, $\tilde{B}_k = \begin{bmatrix} \tilde{B} & I_j \end{bmatrix}^T$, $\tilde{C}_k = \begin{bmatrix} \tilde{C} & 0_j \end{bmatrix}^T$, j=1, k=2。

To ensure the stability of the system, the convergence analysis of the polyhedral model can be conducted through the following methods.

The polyhedral model can be expressed as:

$$\begin{aligned} \xi(k+1) &= \tilde{A}_k \xi(k) + \tilde{B}_k \Delta\tilde{u}(k) + \tilde{D}_k \lambda(k) \\ y(k) &= C\xi(k) \\ [\tilde{A}_k \quad \tilde{B}_k] &\in \Omega \end{aligned} \tag{27}$$

where $\Delta\tilde{u}(k) \in R^m$ is the input variable; $\xi(k) \in R^n$ is the state variable; $y(k) \in R^p$ is the output variable; Ω is the set that describes the model uncertainty.

Considering the uncertainties within Ω, the optimization problem can be formulated as a Min-Max optimization:

$$\min_{\Delta\tilde{u}(k+i|i),i=0,1,\cdots,m} \max_{\tilde{A}_k(k+i)\tilde{B}_k(k+i)]\in\Omega} J_\infty(k) \tag{28}$$

where $\Delta\tilde{u}(k+i|k)$ $\xi(k+i|k)$ represent the predicted values of the state variable and control variable at time t=k+i starting from time t=k. The optimization objective $J_\infty(k)$ is:

$$J_\infty(k) \sum_{i=0}^{\infty} [\xi(k+i|k)^T Q_1 \xi(k+i|k) + \Delta\tilde{u}(k+i|k)^T R_1 \Delta\tilde{u}(k+i|k)] \tag{29}$$

Traditional methods set the control rate to a fixed value:

$$u(k+i|k) = Fx(k+i|k), i \geq 0 \tag{30}$$

This approach is used to obtain the upper bound of the original optimization problem.

Consider the quadratic function:

$$V(x) = x^T P x, P > 0, V(0) = 0 \tag{31}$$

Assuming at time k, for all $\xi(k+i|k)$,$\Delta\tilde{u}(k+i|k), i \geq 0$ that satisfy equation $\xi(k+1) = f(\xi(k),\Delta\tilde{u}(k))$, we have:

$$\begin{aligned} &V(\xi(k+i+1|k)) - V(\xi(k+i|k) \\ &\leq -[\xi(k+i|k)^T Q_1 \xi(k+i|k) + \Delta\tilde{u}(k+i|k)^T R_1 \Delta\tilde{u}(k+i|k)] \end{aligned} \tag{32}$$

It is evident that equation (32) stipulates the non-increasing monotonicity of $V(\xi(k|k))$, hence the polyhedron is robustly stable.

ANALYSIS OF WEIGHT VARIATION WITH TIME DOMAIN

Rolling optimization is the biggest difference between MPC and traditional control, which is carried out simultaneously by local optimization and online rolling. The process of rolling optimization predicts the state quantity of the next moment according to the control quantity and state quantity of the current moment. At each sampling time, the optimal control rate in the time domain is solved according to the optimal performance index at that time. A set of optimal control sequence and optimal predictive output are obtained at time k, but only the first control is actually executed. When the k+1 moment comes, the previous operation is resumed, and then it is rolled one by one until the k+N_p moment stops.

At present, the path tracking control algorithm based on MPC needs to set up the objective function in advance, and the tracking performance mainly depends on the optimization of the objective function. The objective function of the Traditional MPC controller is as follows:

$$J(\xi(k), u(k-1), \Delta U(k)) = \sum_{i=0}^{Np} \| \tilde{\eta}(k+i,k) - \eta_{ref}(k+i,k) \|_Q^2 + \sum_{i=0}^{Nc-1} \| \Delta U(k+i,k) \|_R^2 + \rho\varepsilon^2 \tag{33}$$

where Np is Prediction time domain in MPC algorithm; Nc is the control time domain。Q is the output weight matrix; R is the control increment weight matrix; ρ is the weight of relaxation factor; ε is the relaxation factor.

In the objective function, the first term represents the error of the output, which signifies the tracking performance of the autonomous vehicle. The weight matrix Q can adjust the emphasis on each parameter in the output. The second term represents the smoothness of the control input, which is derived from solving a constrained optimization problem to obtain the control sequence. The weight matrix R can adjust the emphasis on each parameter in the control input. Larger weight coefficients typically enable the system to track the reference trajectory more quickly and accurately, but may lead to larger changes in the control actions, thereby affecting the system's stability. Conversely, smaller weight coefficients can make the output response smoother, but may reduce the tracking speed and tracking performance.

Traditional MPC controllers maintain constant weight values at each step when facing changing road conditions and driving states. However, the optimization problem solved within the Np horizon at each sampling instance is susceptible to variations in the external driving environment. In response to the dynamically changing road conditions, the requirements for vehicle tracking performance are also changing in real-time, implying that the objective function should adapt accordingly. When driving, a driver must consider both distant and immediate road conditions but tends to focus more on short-term changes in the immediate vicinity. Based on this driving habit, the weight matrix should be optimized. The weight matrix is defined as follows:

$$\mathrm{Q} = \begin{bmatrix} q_1 & 0 & \cdots & 0 \\ 0 & q_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & q_{N_p} \end{bmatrix} = \omega_Q \begin{bmatrix} q & 0 & \cdots & 0 \\ 0 & q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & q \end{bmatrix} \tag{34}$$

$$R = \begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{N_{c-1}} \end{bmatrix} = \omega_R \begin{bmatrix} r & 0 & \cdots & 0 \\ 0 & r & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & r \end{bmatrix} \quad (35)$$

In the formula : q_1 denotes the error weight of the output term of the optimization objective function in each step, i=1,2…N_p. The current MPC takes the weight matrix equivalent to $q_1 = q_2 \ldots = q_{Np}$. In this paper, the weight coefficient ω_Q is added, In order to achieve the purpose of the algorithm to exert different attention in different regions.

Analysis of Variable Step Weight on Tracking Effect

Through the aforementioned design, the first error term of the objective function is calculated over Np steps within each sampling period, with variations occurring at each step. The paper conducts extensive comparative experiments to summarize the relationship between tracking performance and the pattern of weight variation. Then selects several representative function changes to verify the rationality of the constructed change objective function.

Taking the double lane change (DLC) condition as the test, three different simulation speeds are set, which are 30km/h, 60km/h and 90km/h respectively. The fixed weight MPC with better tracking effect is selected as the control group, and the weights representing four different forms of change are designed as the experimental group. Their modes of variation differ in that they either gradually increase or decrease, have a fixed change magnitude, or have a variable change magnitude. The change of the weight value is transformed into the function change of the weight ratio as shown in the figure 3:

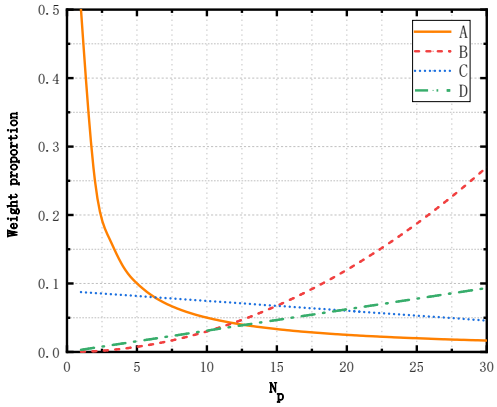

Fig. 3. Different forms of weight proportion variation

The DLC working condition simulation is carried out at the speed of 30km/h, 60km/h and 90km/h, and the four weight change forms are compared with the tracking effect of the traditional MPC controller. This paper focuses on high-speed, focusing on 90km/h simulation results, as shown in Figure 4:

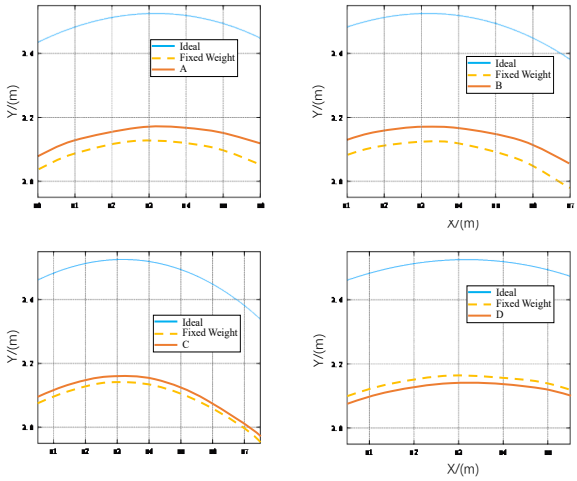

Fig. 4. Lateral position tracking effect

At the most obvious curvature change of DLC working condition, that is, the maximum lateral error of vehicle tracking, the tracking effects of the two controllers are shown. It can be seen that the changing weight method can indeed improve the tracking effect. In order to compare more intuitively, under the conditions of 30km/h, 60km/h and 90km/h of the vehicle, the maximum lateral error and maximum lateral acceleration output by the controller with different weight changes are selected to compare with the tracking effect of the fixed weight MPC controller. The maximum variation and percentage are listed, as shown in Table 1. Among them, the negative sign indicates a decrease, and the positive sign indicates an increase.

Table 1. Maximum Lateral deviation and Maximum Lateral Acceleration Variation

Change mode	ΔE_{max} /m	ΔE_{max} /%	ΔAy_{max} /g	ΔAy_{max} /%
A	-0.1420	-48.78	+0.0143	+3.20
B	+0.0210	+3.65	-0.0225	-12.83
C	-0.0138	-4.69	+0.0048	+1.10
D	+0.0200	+6.72	-0.0076	-1.72

It can be seen from Table 1 that the controller with the weight matrix changing in the form of A has the most obvious improvement on the tracking accuracy, which can reach 48 %, while the controller with the weight changing in the form of B has a greater impact on the stability without affecting the tracking accuracy. Combined with Figure5, it can be seen that when the weight decreases gradually, the greater the change of the weight near the current moment, the more obvious the tracking effect of the algorithm on trajectory tracking. When the weight gradually increases, the greater the weight change away from the current moment, the more obvious the algorithm improves the stability of trajectory tracking. This paper only shows the four most representative changes,

but through extensive experimentation and summarization, the following rules governing the variation of weights over time can be obtained, which are then applied to the design of the controller in the subsequent text.

(1)The greater the proportion of weight coefficients close to the current moment, the smaller the lateral deviation of the system's tracking, and the better the tracking accuracy. Conversely, the greater the proportion of weight coefficients further from the current moment, the better the system's stability, with a reduction in lateral acceleration and an enhanced ability for the vehicle to stabilize.

(2)As weights gradually decrease, the greater the change in weights close to the current moment, the more pronounced the improvement in tracking accuracy by the algorithm; as weights gradually increase, the greater the change in weights further from the current moment, the more significant the improvement in tracking stability by the algorithm.

MODEL PREDICTIVE CONTROLLER DESIGN CONSIDERING UNCERTAIN FACTORS

From the analysis of the previous section, it can be seen that with the progress of rolling optimization, different weight changes will bring different tracking effects, but at the same time, the change of speed will also have a great impact on the tracking effect of the path tracking controller. The design purpose of the tracking controller is to ensure accurate and stable tracking of the target path at any speed. This brings great challenges to the adaptive model predictive controller. According to the law of weight changing with time domain in the above research, the proposed controller can consider the influence factors of curvature and slip rate at the same time according to the change of speed, and adjust the change mode of weight adaptively to compensate for the inaccurate focus of the same weight change mode when the environment changes.

When the vehicle is running at low speed, the controller should focus on the tracking accuracy of the lifting path. Because the lateral acceleration corresponding to low speed driving is small, the driving stability of the vehicle can be generally guaranteed. When the vehicle is running at high speed, due to the increase of lateral acceleration, the requirements for stability are gradually increasing. At this time, the tracking accuracy and lateral stability of the vehicle should be taken into account. Similarly, in different driving sections and large curvature sharp steering sections, due to the constant predicted distance, the controller 's control of the front wheel angle cannot meet the road change under large curvature. The stability and tracking accuracy of the vehicle will be significantly affected, resulting in a rapid decline. But the change of the time domain will increase the computational burden. In the straight road section, the vehicle can generally accurately track the target path.

In this section, the road curvature and operating speed are considered comprehensively, and the consideration of algorithm parameters and their impact on trajectory tracking performance, system stability, and computational efficiency is also given due attention. Within the constraints of the unchanged prediction time domain, an adaptive prediction level parameter is posited. And the undetermined coefficient is adjusted by offline optimization multiple iterations, so as to obtain the result of minimizing the calculation time and tracking error. The adaptive prediction level function is as follows:

$$\omega_Q = F_{\omega_Q}(V_k, \kappa, s, f, N_p) = \upsilon_1 f + \upsilon_2 V_k + \upsilon_3 \kappa + \upsilon_4 s + \upsilon_5 N_p + b \tag{36}$$

where f is the dominant item of the weight change mode; V_k is the longitudinal velocity of the vehicle at time K; κ, s is the road curvature and slip rate; $\upsilon_1, \upsilon_2, \upsilon_3, \upsilon_4, \upsilon_5$ and b is the undetermined coefficient of off-line optimization.

The offline parameter optimization model is shown in Formula 12 and Figure 5:

$$\begin{cases} min \quad f_f = \frac{k_1}{s_1} f_T(\omega_Q, \omega_R) + \frac{k_2}{s_2} f_E(\omega_Q, \omega_R) \\ s.t \quad max\{f_T(\omega_Q, \omega_R)\} \le T_s \\ 0 \le \omega_Q \le 1 \\ 0 \le \omega_R \le 1 \end{cases} \tag{37}$$

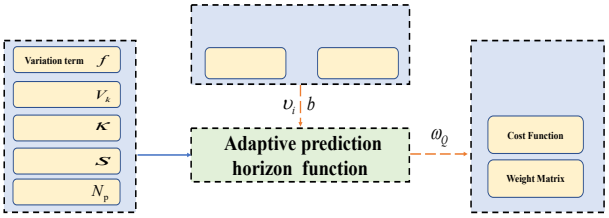

Fig. 5. Offline optimization model diagram

where: Because the two objective functions are not of the same order of magnitude, the scale factor k_1, k_2 and s_1, s_2 are introduced. The objective function f_T is the time required for a single rolling optimization. f_E is the tracking error of the whole process. The constraint dictates that the computation time for a single scroll must not surpass the sampling time. Since q in the weight matrix generally takes a large value, changes in a small range will have a great impact.

The unspecified parameters within the adaptive prediction layer function necessitate offline optimiz - ation. The traditional particle swarm optimization algorithm has the disadvantage of easy to fall into local optimum.(Shami *et al.*, 2022), In this paper, the improved particle swarm optimization algorithm is adopted, so that the inertia weight can be adaptively changed with the iteration of the algorithm.

$$\begin{cases} V_i^{(t+1)} = \omega_f^{(t)} V_i^{(t)} + c_1 m_1 \left(P_{\text{besti}}^{(t)} - X_i^{(t)}\right) + c_2 m_2 \left(G_{\text{best}}^{(t)} - X_i^{(t)}\right) \\ X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)} \\ \omega_f^{(t)} = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \frac{f_i^{(t)} - f_{min}}{f_{avg} - f_{\min}} \end{cases} \quad (38)$$

where $\omega_f^{(t)}$ is the adaptive inertia weight; $\omega_f^{(t)} \in (\omega_{min}, \omega_{max})$; f_{avg} is the average value of fitness; $f_{avg} \in (f_{max}, f_{min})$.

The fitness of the particle is calculated by the integral value of the tracking error between the body position and the sideslip angle of the center of mass. As shown in Formula 14.

$$f_i^{(\iota)} = \omega_P P_{\text{SUM}}^{(\iota)} + \omega_\beta \beta_{\text{SUM}}^{(\iota)} \quad (39)$$

In the formula: $f_i^{(t)}$ is the fitness of the particle $P_{\text{SUM}}^{(\iota)}$ and $\beta_{\text{SUM}}^{(\iota)}$ are used as evaluation indexes to judge the driving stability of vehicles. They represent the integral value of the lateral position error and the integral value of the sideslip angle error of the vehicle tracking the whole process respectively; $\omega_p \; and \; \omega_\beta$ are the weights of the two evaluation indexes of vehicle tracking accuracy and stability, respectively, which vary according to different working conditions. When the vehicle speed is low, the adaptive function is more focused on the tracking accuracy, and the stability of the vehicle is more focused when the vehicle speed is high.

In a prediction time domain, the model is predicted by continuous iteration. predicts the state space equation a for the next b moments, and the resulting output equation is shown in Formula 15.

$$Y(k) = \Psi\zeta(k \mid t) + \Theta\Delta U(k) + \Gamma\omega(k) \quad (40)$$

$$\Theta = \begin{bmatrix} \widetilde{\mathrm{D}}_K \tilde{B}_K & 0 & 0 & 0 \\ \widetilde{\mathrm{D}}_K \widetilde{\mathrm{A}}_K \tilde{B}_K & \widetilde{\mathrm{D}}_K \tilde{B}_K & 0 & 0 \\ \cdots & \cdots & \ddots & \cdots \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-1} \tilde{B}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-2} \tilde{B}_K & \cdots & \widetilde{\mathrm{D}}_K \tilde{B}_K \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c} \tilde{B}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-1} \tilde{B}_K & \cdots & \widetilde{\mathrm{D}}_K \widetilde{\mathrm{A}}_K \tilde{B}_K \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-1} \tilde{B}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-2} \tilde{B}_K & \cdots & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-N_c-1} \tilde{B}_K \end{bmatrix} \quad (41)$$

$$\Gamma = \begin{bmatrix} \widetilde{\mathrm{D}}_K \tilde{C}_K & 0 & 0 & 0 \\ \widetilde{\mathrm{D}}_K \widetilde{\mathrm{A}}_K \tilde{C}_K & \widetilde{\mathrm{D}}_K \tilde{C}_K & 0 & 0 \\ \cdots & \cdots & \ddots & \cdots \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-1} \tilde{C}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-2} \tilde{C}_K & \cdots & \widetilde{\mathrm{D}}_K \tilde{C}_K \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c} \tilde{C}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_c-1} \tilde{C}_K & \cdots & \widetilde{\mathrm{D}}_K \widetilde{\mathrm{A}}_K \tilde{C}_K \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-1} \tilde{C}_K & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-2} \tilde{C}_K & \cdots & \widetilde{\mathrm{D}}_K \tilde{A}_K^{N_p-N_c-1} \tilde{C}_K \end{bmatrix} \quad (42)$$

Combining the polytope model established by considering time-varying parameters and tire cornering stiffness and the proposed adaptive predictive weight function strategy, an adaptive model predictive control algorithm has been developed for uncertain systems. In order to make the high-speed self-driving car have higher tracking accuracy and can quickly realize the path tracking function. It is necessary to select the appropriate optimization objective, establish the objective function, and obtain the future optimal control increment sequence of the controlled system. Therefore, as shown in formula 16. Utilizing the multi-cell dynamics model, an optimized index for trajectory tracking dynamics is formulated and established.

$$\begin{aligned} J\left(\tilde{X}(k), \tilde{u}(t-1), \Delta\tilde{u}(k), \lambda(k)\right) = & \sum_{N_p}^{i=1} \| e_y(k+i) \|_{Q_{ey}}^2 + \sum_{N_p}^{i=1} \| e_\varphi(k+i) \|_{Q_{ep}}^2 \\ & + \sum_{N_p}^{i=1} \| Y(k+i \mid t) - Y_{ref}(k+i \mid t) \|_Q^2 \\ & + \sum_{N_c}^{i=1} \| \Delta\tilde{u}(k+i) \|_R^2 + \rho\varepsilon^2 \end{aligned} \quad (43)$$

where Y and Y_{ref} are the output predictive value and reference value of the system. In order to meet the vehicle can achieve good tracking performance at high speed, $e_y(k+i)$ and $e_\varphi(k+i)$ are added to the original objective function. They are lateral deviation and heading deviation at time k, respectively. The purpose of introducing two evaluation indexes is to reduce the deviation between the actual tracking trajectory and the reference trajectory of the automatic vehicle and improve the tracking accuracy of the vehicle. $Q_{ey} \; and \; Q_{e\varphi}$ are the weight coefficients of position error; $Q \; and \; R$ are improved weight matrices; ε is the relaxation factor.

Through the reformulation of the objective function into a standard quadratic form, it is necessary to consider not only the constraints of vehicle adhesion conditions, but also the hard constraints of centroid sideslip angle constraints and tire sideslip angle constraints to ensure vehicle stability. (Kanghyun Nam *et al.*, 2012)。In the adaptive adjustment of the weight coefficient, the model should also be adaptively changed at each sampling time, so a dynamic model needs to be established, as shown in Equation 17 below. To achieve a balance between the efficiency and stability of the quadratic programming problem in each iteration period.

$$\begin{aligned} & min \quad [\Delta u(t)^T, \varepsilon]^T H_t [\Delta u(t)^T, \varepsilon] \\ & \qquad + G_t [\Delta u(t)^T, \varepsilon] \\ & \text{s.t.} \quad \begin{bmatrix} \Delta U_{min} \\ y_{hc\,min} \\ y_{sc\,min} - \varepsilon \end{bmatrix} \leq \begin{bmatrix} \Delta U \\ y_{hc} \\ y_{sc} \end{bmatrix} \leq \begin{bmatrix} \Delta U_{max} \\ y_{hc\,max} \\ y_{hc\,max} + \varepsilon \end{bmatrix} \\ & H_t = \begin{bmatrix} \Theta^T Q \Theta + R & 0 \\ 0 & \rho \end{bmatrix}, \quad G_t \\ & \qquad = [2e^T Q \Theta \quad 0] \end{aligned} \quad (44)$$

where ΔU serves as the control increment constraint, while y_{hc} represents the hard output constraint. y_{sc} serves as the soft constraint for lateral acceleration, which can be adjusted by ε, and e is the tracking error in the predicted time domain.

When a vehicle is driving and suddenly experiences a large steering angle or encounters a slippery road surface, it can easily enter the nonlinear operating region of the tires, where the tire forces tend to saturate. If the saturated tire forces cannot meet the demands of the vehicle body, it can lead to skidding and other instability phenomena, which seriously affect driving safety. Therefore, to avoid the occurrence of such phenomena, it is necessary to impose constraints on the state parameters from a dynamics perspective, ensuring that the tires always operate within the linear region.

The yaw dynamics model of high-speed autonomous vehicles takes into account the matching between the road adhesion coefficient and the tire forces, as well as the limitations of the vehicle's lateral

acceleration. The expression and constraint conditions for the vehicle's lateral acceleration are as follows:

$$\begin{cases} a_y = \dot{x}\dot{\varphi} + \dot{y} \\ a_y \le \mu g \end{cases} \tag{45}$$

In the expression for the vehicle's lateral acceleration，a_y accounts for approximately 85% of the maximum allowable lateral acceleration. Therefore, the constraints on the yaw rate $\dot{\varphi}$ and the vehicle's lateral slip angle β can be expressed in the following form:

$$|\dot{\varphi}| \le 0.85\frac{\mu g}{\dot{x}} \tag{46}$$

$$|\beta| \approx \left|\frac{\dot{y}}{\dot{x}}\right| \le \arctan(0.02\mu g) \tag{47}$$

When a vehicle experiences a sideslip, its lateral stability is affected, which typically leads to rear-wheel skidding. Therefore, the maximum slip angle constraint applies to the rear wheels. The constraint equation for the maximum yaw angle of the rear wheels can be expressed as an inequality involving the vehicle's lateral velocity and yaw rate:

$$\left|\frac{\dot{y}-\mathrm{b}\dot{\varphi}}{\dot{x}}\right| \le \alpha_{sat_{rl,rr}} \tag{48}$$

where $\alpha_{sat_{rl,rr}}$ represents the tire slip angle when the lateral force of the rear wheels reaches saturation.

The vehicle's yaw rate can be constrained as follows:

$$\dot{\varphi}_{max} = \frac{C_{\alpha r}\alpha_{sat_{rl,rr}}\left(1+\frac{b}{a}\right)}{m\dot{x}} \tag{49}$$

To ensure stability, we introduce terminal constraints:

$$\hat{\xi}(k+N|k) = 0 \tag{50}$$

The optimal value of the objective function in each period is used as the Lyapunov function:

$$V^0(K) = \min_u \sum_{i=1}^{N} \ell(\xi(k+i), u(k+i)) \tag{51}$$

where: $\ell(\xi, u) \ge 0$，$\ell(\xi, u) = 0$ if and only if $\xi = 0$，$u = 0$。

Assuming the model is unbiased and noise interference is not considered, the following is derived through calculation and derivation:

$$V^0(K+1) \le -\ell(\xi(k+1), u^0(k)) + V^0(k) + \min_u\{\ell(\xi(k+1+N), u(k+N))\} \tag{52}$$

Due to the presence of terminal constraints, it can be ensured that:

$$\min_u\{\ell(x(k+1+N), u(k+N))\} = 0 \tag{53}$$

Since $\ell(\xi(k+1), u(k)) \ge 0$,it follows that $V^0(k+1) \le V^0(k)$， ensuring the stability of the closed-loop system.

RESULTS AND ANALYSIS

To validate the proposed algorithm, the hardware-in-the-loop (HIL) test platform illustrated in Figure 6 is employed. A HIL setting includes a real controller and a virtual controlled object, which ensures the real-time performance of the simulation process. The specific implementation is to build the road model and algorithm model through CarSim and MARLAB / Simulink, and then import them into the PXI platform to run.

Fig. 6. Hardware-in-the-loop

A test scenario with DLC centerline tracking problem is selected from CarSim, and the trajectory tracking performance and system stability of the proposed adaptive algorithm car at high speed are analyzed. The test speed is 90 km / h. The test vehicle model used by is selected from carsim, and the main parameters of the vehicle are shown in the following Table 2.

Tab. 2. The main parameters of the experiment

Parameter（Unit）	Value	Parameter（Unit）	Value
$m(kg)$	1723	$I_x(kg\cdot m^2)$	606.1
$l_f(m)$	1.232	$C_{\alpha f}/(N\cdot rad^{-1})$	-66900
$l_r(m)$	1.468	$C_{\alpha r}/(N\cdot rad^{-1})$	-62700
Iz	4175	$g/(m\cdot s^{-2})$	9.8
T	0.05	ε	1000
N_p	12	N_c	5

The framework design and parameter setting of PSO have an important influence on the solution process of particle swarm optimization. In this paper, the parameter settings of PSO algorithm are shown in Table 3.

Tab. 3. PSO Algorithm Parameter

parameter	Description	Value
T	Maximum number of Iterations	300
N	Number of initial population	100
C1	Individual learning factors	0.9
C2	Group learning factor	0.9
W	Inertia weight	0.7

The optimal solution obtained after iteration by the PSO algorithm is used for the design of the control and the resulting values of the coefficients to be determined are shown in Table 4

Tab. 4 Optimization coefficient results

Parameter	Value
υ_1	0.53

υ_2	0.42
υ_3	0.005
υ_4	0.005
υ_5	0.23
b	0.39

HIL simulation was conducted to compare AMPC with UM-AMPC. The AMPC algorithm utilizes a three-degree-of-freedom model and employs fuzzy rules for adaptive control. In contrast to the AMPC algorithm, the UM-AMPC algorithm incorporates the control strategy proposed in this paper, which not only adaptively adjusts parameters based on driving conditions but also accounts for disturbances caused by some dynamic uncertain parameters.

The reference trajectory is as follows:

$$Y_{\text{ref}}(p) = \frac{dy_1}{2}\left(1+\tanh\left(\frac{\text{shape}}{dx_1}\left(X_{\text{predict}}(p)-X_{s1}\right)-\frac{\text{shape}}{2}\right)\right) \frac{dy_2}{2}\left(1+\tanh\left(\frac{\text{shape}}{dx_2}\left(X_{\text{predict}}(p)-X_{s2}\right)-\frac{\text{shape}}{2}\right)\right) \quad (54)$$

$$shape = 2.4,\ dx_1 = 25,\ dx_2 = 21.95,\ dy_1 = 4.05,\ dy_2 = 5.7,\ X_{s1} = 27.19,\ X_{s2} = 56.46$$

DLC tests were performed under the same conditions for both methods, and the trajectory tracking performance and driving stability were analyzed. The simulation results are as follows:

Trajectory Tracking Performance Analysis

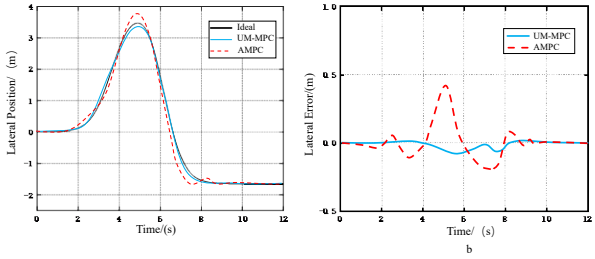

Fig. 7. Lateral analysis of the vehicle

It is particularly important to verify whether the improvement of a tracking algorithm has a good effect and whether the tracking accuracy is improved. This chapter first analyzes the trajectory tracking performance of autonomous vehicles through the lateral position and lateral tracking error of the vehicle simulated by the HIL platform, as shown in Figure 7. In contrast to the AMPC algorithm, the adaptive UM-MPC algorithm yields enhanced tracking efficacy in 2 ~ 8 seconds, especially in the peak error of about 6 seconds, the accuracy is improved most obviously. This is because the UM-MPC can focus the attention of the algorithm by changing its own parameters when the road curvature changes sharply and the vehicle travels faster. Through calculation and analysis, the average tracking error is reduced by 46.11 %. Compared with the graph in the third chapter, it is obvious that the weight matrix with good effect can be obtained by off-line optimization regardless of the distance from the moment, which also verifies the correctness of the change rule of the weight matrix with the prediction time domain.

Vehicle Stability Analysis

Based on the analysis of tracking accuracy, the tracking stability of the vehicle during driving is analyzed from four aspects: target angle, yaw angle error, yaw rate and sideslip angle error. Four indicators are simulated by the HIL as shown in Figure 8. It can be seen from the target angle of the controller output that the target angle fluctuation of the UM-MPC algorithm is relatively gentle, and it converges to a stable state before the AMPC algorithm. In terms of yaw angle, yaw rate and sideslip angle, the UM-MPC algorithm can greatly reduce the average error and keep the vehicle in a stable state. In addition, on the sideslip angle index, the adaptive UM-MPC algorithm shows the best performance, and the stability improvement is particularly obvious.

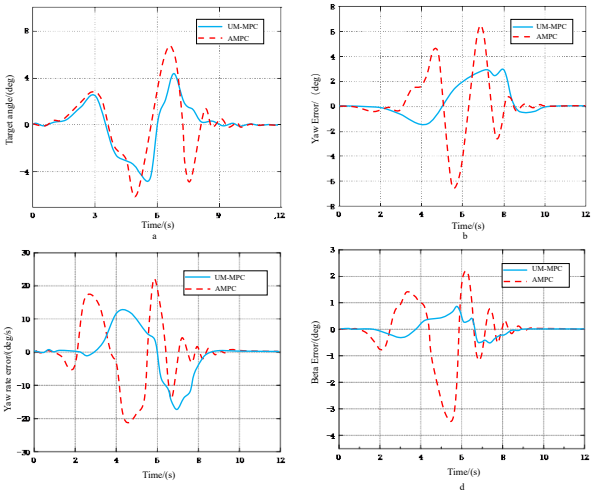

Fig. 8 Analysis of automotive stability indicators

CONCLUSIONS

This paper investigates the model predictive method for adaptive trajectory tracking considering time-varying parameter uncertainty models by analyzing the patterns of temporal domain and weight variations. To enhance the tracking accuracy, robust stability, and computational efficiency during the rolling optimization process, an UM-AMPC algorithm addressing system uncertainties is proposed. Based on the variation patterns of the weight matrix, experiments were conducted to explore the relationship between the changes in the weight matrix and the tracking performance of autonomous vehicles. Furthermore, to reduce interference from external environments, a polytopic model was established. Subsequently, an adaptive weight level prediction function was

designed based on the vehicle's longitudinal speed, road curvature, the variation pattern of weights, and the slip ratio. An improved particle swarm optimization algorithm was then employed to calculate the target parameter coefficients. To accommodate the adaptive changes of the model at each sampling moment, a dynamic quadratic programming solver was designed. Finally, through semi-physical simulation, it was verified that the UM-MPC algorithm reduced the average yaw rate error by 59.68% and the average tracking error by 46.11% compared to the MPC algorithm. The test results indicate that this method can accurately suppress the adverse effects brought by uncertain parameters and predict the target angle signal based on the current working state. This control algorithm provides a safety guarantee for autono- mous driving systems, offering improvements in stability, accuracy, and computational efficiency over traditional model predictive control.

Future Outlook: To further verify the feasibility of the algorithm, road vehicle tests will also be conducted. In addition, as this study did not consider the impact of road gradient changes, future research will also need to strengthen the study of the roll state of autonomous vehicles. The algorithm will be designed and optimized at a deeper level to further improve vehicle stability and tracking accuracy.

ACKNOWLEDGMENT

Funding: The authors acknowledge the support provided by the Natural Science Foundation of Hebei Province of China (No. E2016402066) and the High-Level Talent Project of Hebei Province of China: Integrated Research and Simulation Realization of Vehicle Ride Comfort and Handling Stability (Grant No. B2017003026). Scientific research projects of universities in Hebei Province in 2024. (QN2024133)

REFERENCES

Chen, Y. and Chen, Y. (2023), “Nonlinear Adaptive Fuzzy Control Design for Wheeled Mobile Robots with Using the Skew Symmetrical Property”, *Symmetry*, Vol. 15 No. 1, p. 221, doi: 10.3390/sym15010221.

Gao, F., Han, Y., Eben Li, S., Xu, S. and Dang, D. (2023), “Accurate Pseudospectral Optimization of Nonlinear Model Predictive Control for High-Performance Motion Planning”, *IEEE Transactions on Intelligent Vehicles*, Vol. 8 No. 2, pp. 1034–1045, doi: 10.1109 /TIV.2022.3153633.

Geng, K. and Liu, S. (2020), “Robust Path Tracking Control for Autonomous Vehicle Based on a Novel Fault Tolerant Adaptive Model Predictive Control Algorithm”, *Applied Sciences*, Vol. 10 No. 18, p. 6249, doi: 10.3390/app10186249.

Hang, P. and Chen, X. (2021), “Path tracking control of 4-wheel-steering autonomous ground vehicles based on linear parameter-varying system with experimental verification”, *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, Vol. 235 No. 3, pp. 411–423, doi: 10.1177/0959651820934572.

Hu, K. and Cheng, K. (2023), “Trajectory Planning for an Articulated Tracked Vehicle and Tracking the Trajectory via an Adaptive Model Predictive Control”, *Electronics*, Vol. 12 No. 9, p. 1988, doi: 10.3390/electronics 12091988.

Kanghyun Nam, Fujimoto, H. and Hori, Y. (2012), “Lateral Stability Control of In-Wheel-Motor-Driven Electric Vehicles Based on Sideslip Angle Estimation Using Lateral Tire Force Sensors”, *IEEE Transactions on Vehicular Technology*, Vol. 61 No. 5, pp. 1972–1985, doi: 10.1109/TVT.2012.2191627.

Koga, A., Okuda, H., Tazaki, Y., Suzuki, T., Haraguchi, K. and Kang, Z. (2016), “Realiz -ation of different driving characteristics for autonomous vehicle by using model predictive control”, *2016 IEEE Intelligent Vehicles Symposium (IV)*, presented at the 2016 IEEE Intelligent Vehicles Symposium (IV), IEEE, Gotenburg, Sweden, pp. 722–728, doi: 10.1109/IVS.2016.7535467.

Li, D.-P., Liu, Y.-J., Tong, S., Chen, C.L.P. and Li, D.-J. (2019), “Neural Networks-Based Adaptive Control for Nonlinear State Constrained Systems With Input Delay”, *IEEE Transactions on Cybernetics*, Vol. 49 No. 4, pp. 1249–1258, doi:10.1109/ TCYB. 2018.2799683.

Li, X., Sun, Z., Cao, D., Liu, D. and He, H. (2017), “Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles”, *Mechanical Systems and Signal Processing*, Vol. 87, pp. 118–137, doi: 10.1016/j.ymssp. 2015.10.021.

Luan, Z., Zhang, J., Zhao, W. and Wang, C. (2020), “Trajectory Tracking Control of Autonomous Vehicle With Random Network Delay”, *IEEE Transactions on Vehicular Technology*, Vol. 69 No. 8, pp. 8140–8150, doi: 10.1109/TVT.2020.2995408.

Mashadi, B., Ahmadizadeh, P., Majidi, M. and Mahmoodi-Kaleybar, M. (2015), “Integrated robust controller for vehicle path following”, *MULTIBODY SYSTEM DYNAMICS*, Springer, Dordrecht, Vol. 33 No. 2, pp. 207–228, doi: 10.1007/s11044-014-9409-8.

Nan, S., Danfeng, S. and Dejun, Y. (2017), "An integrated stability control system for decentralized drive electric vehicles", *2017 36th Chinese Control Conference (CCC)*, presented at the 2017 36th Chinese Control Conference (CCC), IEEE, Dalian, China, pp. 9519–9524, doi: 10.23919/ChiCC.2017.8028876.

Nie, X., Min, C., Pan, Y., Li, K. and Li, Z. (2022), "Deep-Neural-Network-Based Modelling of Longitudinal-Lateral Dynamics to Predict the Vehicle States for Autonomous Driving", *Sensors*, Vol. 22 No. 5, p. 2013, doi: 10.3390/s22052013.

Peza-Solis, J.F., Silva-Navarro, G., Garcia-Perez, O.A. and Trujillo-Franco, L.G. (2022), "Trajectory tracking of a single flexible-link robot using a modal cascaded-type control", *Applied Mathematical Modelling*, Vol. 104, pp. 531–547, doi: 10.1016/j.apm.2021.12. 002.

Qu, X., Yu, Y., Zhou, M., Lin, C.-T. and Wang, X. (2020), "Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: A reinforcement learning based approach", *Applied Energy*, Vol. 257, p. 114030, doi: 10.1016/j.apenergy. 2019. 114030.

Raffo, G.V., Gomes, G.K., Normey-Rico, J.E., Kelber, C.R. and Becker, L.B. (2009), "A Predictive Controller for Autonomous Vehicle Path Tracking", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 10 No. 1, pp. 92–102, doi: 10.1109/TITS.2008.2011697.

Shami, T.M., El-Saleh, A.A., Alswaitti, M., Al-Tashi, Q., Summakieh, M.A. and Mirjalili, S. (2022), "Particle Swarm Optimization: A Comprehensive Survey", *IEEE Access*, Vol. 10, pp. 10031–10061, doi: 10.1109/ ACC ESS.2022.3142859.

Wei, C., Paschalidis, E., Merat, N., Crusat, A.S., Hajiseyedjavadi, F. and Romano, R. (2023), "Human-Like Decision Making and Motion Control for Smooth and Natural Car Following", *IEEE Transactions on Intelligent Vehicles*, Vol. 8 No. 1, pp. 263– 274, doi: 10.1109/TIV.2021.3098184.

Xia, Y., Pu, F., Li, S. and Gao, Y. (2016), "Lateral Path Tracking Control of Autonomous Land Vehicle Based on ADRC and Differential Flatness", *IEEE Transactions on Industrial Electronics*, Vol. 63 No. 5, pp. 3091–3099, doi: 10.1109/TIE.2016.2531021.

Xu, S. and Peng, H. (2020), "Design, Analysis, and Experiments of Preview Path Tracking Control for Autonomous Vehicles", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21 No. 1, pp. 48–58, doi: 10.1109/TITS.2019.2892926.

Yang, H., Wang, Z., Xia, Y. and Zuo, Z. (2023), "EMPC with adaptive APF of obstacle avoidance and trajectory tracking for autonomous electric vehicles", *ISA Transactions*, Vol. 135, pp. 438–448, doi: 10.1016/j.isatra.2022.09.018.

Yuan, S., Zhao, P., Zhang, Q. and Hu, X. (2019), "Research on Model Predictive Control-based Trajectory Tracking for Unmanned Vehicles", *2019 4th International Conference on Control and Robotics Engineering (ICCRE)*, presented at the 2019 4th International Conference on Control and Robotics Engineering (ICCRE), IEEE, Nanjing, China, pp. 79–86, doi: 10.1109/ICCRE.2019.8724158.

Zhang, J., Zhang, B., Zhang, N., Wang, C. and Chen, Y. (2021), "A novel robust event-triggered fault tolerant automatic steering control approach of autonomous land vehicles under in-vehicle network delay", International Journal of Robust and Nonlinear Control, Vol. 31 No. 7, pp. 2436–2464, doi: 10.1002/rnc.5393.

Zhou, L., Wang, G., Sun, K. and Li, X. (2019), "Trajectory Tracking Study of Track Vehicles Based on Model Predictive Control", *Strojniški Vestnik - Journal of Mechanical Engineering*, Vol. 65 No. 6, pp. 329–342, doi: 10.5545/sv-jme.2019.5980.