Proceedings of the 4th International Conference on Mathematical Physics and Computational Simulation
DOI: 10.54254/2753-8818/2026.34703

Research on Vehicle Trajectory Accuracy Improvement Based on Adaptive Weight MPC

Huazhi Tong

College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling, China
15165792336@163.com

Abstract. Aiming at the problem that traditional Model Predictive Control (MPC) adopts fixed weights and cannot adapt to fluctuations in vehicle speed and dynamic changes of tracking error in vehicle trajectory tracking, this paper proposes an adaptive weight MPC control algorithm based on joint perception of tracking error and longitudinal velocity. A linear 2-Degree-of-Freedom vehicle dynamic model is established, and a trajectory tracking MPC framework is constructed. The hyperbolic tangent function and velocity correction factor are introduced to realize real-time dynamic adjustment of the state weight matrix and control weight. MATLAB simulation results show that compared with fixed-weight MPC, the proposed adaptive weight MPC reduces the lateral error convergence time by 48.5%, the maximum lateral absolute error by 39.1%, and the maximum sideslip angle by 31.3%. It significantly improves the trajectory tracking accuracy, optimizes the yaw rate and vehicle sideslip angle, effectively enhances vehicle driving stability, and can provide a reference for lateral trajectory tracking control of autonomous vehicles.

Keywords: adaptive weight, model predictive control, trajectory tracking, yaw rate, sideslip angle

1. Introduction

With the rapid development of autonomous driving technology, vehicle trajectory tracking control, as one of the core technologies, directly determines the driving safety and ride comfort of autonomous vehicles. Benefiting from the advantages of multi-constraint processing, future trajectory prediction and rolling optimization, Model Predictive Control (MPC) has been widely applied in the field of vehicle lateral control [1].

Model Predictive Control (MPC) algorithm is widely applied to the path tracking control of intelligent driving vehicles. However, the traditional MPC algorithm adopts fixed time-domain parameters, which deteriorates the stability and reliability of intelligent vehicles under complex working conditions. Accordingly, the time-domain parameter optimized MPC combined with other algorithms has become a current research hotspot [2, 3]. Traditional MPC generally adopts a fixed weight matrix, which cannot dynamically adjust weights according to real-time driving conditions. Under working conditions such as longitudinal velocity fluctuation and sudden changes in tracking error, it is prone to problems including decreased tracking accuracy and deteriorated lateral stability.

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2026.34703

To address the above limitations, this paper designs an adaptive weight MPC algorithm integrating real-time tracking error and longitudinal vehicle speed. By dynamically correcting state weights and control weights, the algorithm balances trajectory tracking accuracy and driving stability under various working conditions and improves the controller's adaptability to dynamic operating conditions [4, 5].

2. Vehicle trajectory tracking system modeling

2.1. Degree-of-freedom vehicle dynamic model

2.1.1. Model assumptions

1. Ignore the longitudinal, vertical and roll motions of the vehicle, and only consider two degrees of freedom including lateral displacement and yaw motion;

2.The longitudinal vehicle speed is approximately constant with small velocity disturbances allowed in simulation;

3.The tire operates in the linear sideslip range, and the tire sideslip force has a linear relationship with the sideslip angle;

4.Ignore nonlinear factors such as air resistance, suspension deformation and steering clearance;

5.The vehicle sideslip angle remains small.

2.1.2. Dynamic differential equations

The vehicle lateral dynamic equilibrium equations are as follows:

$$m(\dot{y}_v + v_x r) = -C_f\alpha_f - C_r\alpha_r \tag{1}$$

$$I_z\dot{r} = l_f C_f\alpha_f - l_r C_r\alpha_r \tag{2}$$

The tire sideslip angle is expressed as:

$$\alpha_f = -(v_y + l_f r)/v_x + \delta_f \tag{3}$$

$$\alpha_r = -(v_y - l_r r)/v_x \tag{4}$$

Where: m is the vehicle mass(kg); I_z is the moment of inertia about the z-axis(kg·m²); l_f and l_r are the distances from the front and rear axles to the vehicle centroid(m); C_f and C_r are the front and rear tire cornering stiffness(N/rad); v_x and v_y are the longitudinal and lateral velocities(m/s); r is the yaw rate(rad/s); δ_f is the front wheel steering angle(rad); α_f and α_r are the front and rear tire sideslip angles(rad) [6].

2.1.3. Continuous-time state-space equation

The state vector and control input are defined as:

$$x = [y, \psi, v_y, r]^T$$

DOI: 10.54254/2753-8818/2026.34703

$$u = \delta_f$$

The continuous state-space equation is established as:

$$\dot{x} = A_c x + B_c u \tag{5}$$

Where x refers to the state vector (dimensionless); y refers to the lateral displacement (meters, m); ψ refers to the yaw angle (radians, rad); u refers to the control input (radians, rad); A_c refers to the continuous system matrix (dimensionless); B_c refers to the continuous input matrix (dimensionless) [7]

The system matrix and input matrix are expressed as:

$$A_c = \begin{bmatrix} 0 & v_x & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{(C_f+C_r)}{(mv_x)} & \frac{(l_rC_r-l_fC_f)}{(mv_x)} - v_x \\ 0 & 0 & \frac{(l_rC_r-l_fC_f)}{(I_zv_x)} & -\frac{\left(l_f^2C_f+l_r^2C_r\right)}{(I_zv_x)} \end{bmatrix}$$

$$B_c = \begin{bmatrix} 0 \\ 0 \\ \frac{C_f}{m} \\ \frac{l_f C_f}{I_z} \end{bmatrix}$$

2.2. System discretization

The forward Euler method is adopted for system discretization with the sampling time T_s :

$$x(k+1) = A_d x(k) + B_d u(k) \tag{6}$$

$$A_d = I + A_c T_s$$

$$B_d = B_c T_s$$

Where k refers to the discrete time step (dimensionless); A_d refers to the discrete system matrix (dimensionless); B_d refers to the discrete input matrix (dimensionless); I refers to the identity matrix (dimensionless); T_s refers to the sampling time (seconds, s) [8, 9]

The predicted state vector is defined as:

DOI: 10.54254/2753-8818/2026.34703

$$X = \begin{bmatrix} x(k+1) \\ x(k+2) \\ \vdots \\ x(k+ N_p) \end{bmatrix}$$

The control input sequence is defined as:

$$u = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+ N_c - 1) \end{bmatrix}$$

2.3. MPC prediction model

Given the prediction horizon N_p and control horizon N_c, the matrix form of the prediction model is established:

$$X(k) = \Phi x(k) + F\Delta U(k) \tag{7}$$

$$\Phi = \left[A_d; A_d^2; \ldots; A_d^{N_p}\right]^T$$

$$F = \begin{bmatrix} B_d & 0 & \cdots & 0 \\ A_d B_d & B_d & \cdots & 0 \\ \vdots & & & \vdots \\ A_d^{(Np-1)} B_d & A_d^{(Np-2)} B_d & \cdots & A_d^{(Np-Nc)} B_d \end{bmatrix}$$

Where X refers to the predicted state vector (dimensionless); Φ refers to the state prediction matrix (dimensionless) ;F refers to the control input matrix (dimensionless); ΔU refers to the control increment vector (dimensionless); N_p refers to the prediction horizon (dimensionless); N_c refers to the control horizon (dimensionless) [10, 11]

2.4. MPC optimization objective and constraints

The objective function is designed as:

$$J = (\Phi x(k) + F\Delta U(k) - X_{ref})^T Q_{bar} \times (\Phi x(k) + F\Delta U(k) - X_{ref}) + U^T R_{bar} U$$

DOI: 10.54254/2753-8818/2026.34703

Converted to the standard Quadratic Programming form:

$$J = \frac{1}{2}\Delta U\left(k\right)^{T} H \Delta U\left(k\right) + f^{T} \Delta U\left(k\right)$$

$$H = F^{T} Q_{bar} F + R_{bar}$$

$$f = \left(\Phi x\left(k\right) - X_{ref}\right)^{T} Q_{bar} F$$

Input constraints:

$$U_min \leq \Delta U(k) \leq U_max$$

Where J refers to the objective function (dimensionless); X_{ref} refers to the reference state vector (dimensionless); Q_{bar} refers to the extended state weight matrix (dimensionless); R_{bar} refers to the extended control weight matrix (dimensionless) ;H refers to the Hessian matrix (dimensionless);f refers to the gradient vector (dimensionless)

3. Adaptive weight design for MPC

$$\boldsymbol{e_{total}} = \left|\boldsymbol{y_{ref}}\left(\boldsymbol{k}\right) - \boldsymbol{y}\left(\boldsymbol{k}\right)\right| + \left|\boldsymbol{\psi_{ref}}\left(\boldsymbol{k}\right) - \boldsymbol{\psi}\left(\boldsymbol{k}\right)\right|$$

$$\alpha_{err} = 1 + 1.8tanh\left(e_{total}\right)$$

$$\alpha_{vx} = v_{x_{nominal}} / v_{x}\left(k\right)$$

$$Q_{adaptive} = Q_{base} \cdot diag\left(\alpha_{err}\alpha_{vx}, \alpha_{err}\alpha_{vx}, 0.2, 0.2\right)$$

$$R_{adaptive} = R_{base} / \left(\alpha_{err}\alpha_{vx}\right)$$

Where:
e_{total} refers to the total tracking error (meters or radians, m/rad)
y_{ref} refers to the reference lateral displacement (meters, m)

DOI: 10.54254/2753-8818/2026.34703

ψ_{ref} refers to the reference yaw angle (radians, rad)
α_{err} refers to the error correction factor (dimensionless)
α_{vx} refers to the velocity correction factor (dimensionless)
$v_{x_{nominal}}$ refers to the nominal longitudinal velocity (meters per second, m/s)
v_x (k) refers to the actual longitudinal velocity (meters per second, m/s)
Q_{base} refers to the basic state weight matrix (dimensionless)
R_{base} refers to the basic control weight (dimensionless)
$Q_{adaptive}$ refers to the adaptive state weight matrix (dimensionless)
$R_{adaptive}$ refers to the adaptive control weight (dimensionless)

3.1. Basic controller parameters

The adaptive weight takes the fixed weight as the benchmark to ensure control stability and comparison fairness.

The core parameters are set as follows:

• Basic state weight matrix: $Q_{base} = diag\,(100,200,21,10)$
• Basic control weight: $R_{base} = 0.18$
• Nominal longitudinal velocity: $v_{x_{nominal}} = 10m/s$
• Sampling time: $T_s = 0.04s$
• Prediction horizon: $N_p = 50$; Control horizon: $N_c = 25$

3.2. Real-time tracking error calculation

Lateral position error: $e_y = y_{ref}\,(k) - y\,(k)$

Yaw angle error: $e_{psi} = psi_{ref}\,(k) - psi\,(k)$

Total absolute tracking error: $e_{total} = |e_y| + |e_{psi}|$

3.3. Design of dual dynamic correction factors

3.3.1. Velocity correction factor

Low vehicle speed leads to slow steering response, which requires increased tracking weight. The velocity factor is calculated as:

$$v_{x_{factor}} = v_{x_{nominal}} / v_{x_{actual(k)}}$$

To avoid sudden weight mutation, the factor is limited within:

$$v_{x_{factor}} \in [0.9,1.2]$$

3.3.2. Error correction factor

The hyperbolic tangent function is adopted to adjust the error factor smoothly. Larger tracking errors correspond to higher tracking weights, while infinite weight growth is avoided:

DOI: 10.54254/2753-8818/2026.34703

$$error_{factor} = 1 + 1.8 \times tanh\left(e_{total}\right)$$

The factor is limited within:

$$error_{factor} \in [1,3]$$

3.4. Adaptive weight update rules

3.4.1. Adaptive state weight matrix $Q_{adaptive}$

Dynamic adaptive adjustment is only implemented for two core tracking states including lateral position and yaw angle, while low fixed weights are maintained for lateral velocity and yaw rate to focus on tracking targets:

$$Q_{adaptive} = Q_{base} \times diag\begin{bmatrix} v_{x_{factor}} error_{factor} \\ v_{x_{factor}} error_{factor} \\ 0.2 \\ 0.2 \end{bmatrix}$$

3.4.2. Adaptive control weight $R_{adaptive}$

The control weight is inversely proportional to the comprehensive correction factor. When the error is large and the vehicle speed is low, the control weight is reduced to allow greater steering correction; when the error is small and the vehicle speed is high, the control weight is increased to ensure smooth control:

$$R_{adaptive} = R_{base} / \left(error_{factor} \times v_{x_{factor}}\right)$$

To guarantee system stability, the control weight is limited within:

$$R_{adaptive} \in [0.01,0.5]$$

3.5. Control constraints and optimization solution

Maximum front wheel steering angle: $delta_{max} = pi/5rad$.

To improve driving stability, the actual control constraint is set as:

$$delta_f \in [-0.6delta_{max}, 0.6delta_{max}]$$

DOI: 10.54254/2753-8818/2026.34703

The Quadratic Programming problem is solved by MATLAB, and the optimal front wheel steering angle is output through rolling optimization.

3.6. Control flow of adaptive MPC

• Collect the actual vehicle speed and vehicle state at moment ;
- Calculate lateral tracking error, yaw angle error and total tracking error;
- Compute velocity correction factor and error correction factor;
- Update adaptive weights $Q_{adaptive}$ and $R_{adaptive}$;
- Reconstruct optimization matrices and solve the optimal control variable;
- Impose steering constraints and update vehicle state;
- Enter the next sampling cycle to complete rolling optimization.

4. Simulation results and analysis

4.1. Simulation parameter settings

The vehicle and controller parameters are consistent with Section 3.1. The total simulation time is 30 seconds. The reference trajectory adopts a composite sine–cosine curve, and small disturbances are applied to the longitudinal velocity. A comparative simulation between fixed-weight MPC and adaptive-weight MPC is conducted.

4.2. Comparative analysis of tracking performance

Based on the vehicle dynamic model, the vehicle behavior can be simulated to verify the performance of the designed controller [2], Quantitative indicators are adopted to compare the trajectory tracking performance of the two controllers, and the results are shown in the following table and figure:

Table 1. Quantitative comparison of MPC control performance

Variable Meaning	Fixed MPC	Adaptive MPC	Change Rate	Trend
Lateral error convergence time /s	0.33	0.17	↓48.5%	Improved
Maximum lateral absolute error /m	0.089	0.054	↓39.1%	Improved
Average lateral absolute error /m	0.020	0.020	↓0.5%	Improved
Maximum yaw rate /(°/s)	14.3	14.2	↓1.0%	Improved
Maximum sideslip angle / (°)	1.01	0.69	↓31.3%	Improved

DOI: 10.54254/2753-8818/2026.34703

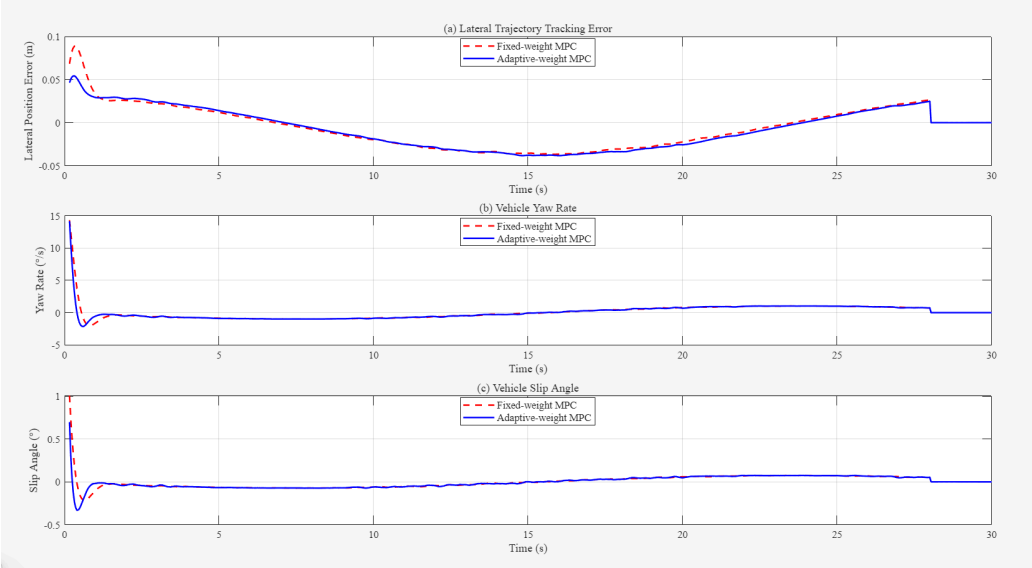

Figure 1. Comparison of performance between adaptive weight MPC and fixed weight MPC

The simulation results indicate that in the early stage with large initial errors, the proposed adaptive weight MPC significantly increases tracking weights dynamically, leading to a faster lateral error convergence rate compared with fixed-weight MPC. When the tracking error converges to a small range, the adaptive weight automatically returns to the benchmark value, and the controller performance is consistent with fixed-weight MPC, balancing tracking accuracy and driving stability.

It is worth noting that the initial transient spike is not significant for control performance evaluation. The initial transient spike is caused by the mismatch between the system initial state and the reference trajectory as well as the unconverged solution of the MPC controller. It belongs to the transient phenomenon in the start-up transition stage, which cannot reflect the steady-state control performance of the algorithm under normal working conditions and thus has no significance for comparing the advantages and disadvantages of different algorithms [3]. In accordance with vehicle dynamic control evaluation specifications, evaluation is conducted by excluding the first 0.2 seconds and ignoring the first five sampling points. The adaptive weight MPC presents superior performance in convergence speed, tracking accuracy and stability, with basically unchanged yaw rate and overall optimized control effect.

4.3. Optimization effect of control stability

The adaptive weight MPC realizes synchronous optimization of tracking accuracy and driving stability. The convergence speed of lateral error is greatly improved, and the maximum tracking error is significantly reduced. The yaw rate remains basically stable, while the maximum sideslip angle is reduced by 31.3%, which effectively suppresses vehicle sideslip, improves handling stability and ride comfort at medium and high speeds, and solves the contradiction between tracking accuracy and stability of traditional fixed-weight MPC.

4.4. Dynamic variation law of adaptive weights

Q_{11} and Q_{22} are error-driven dynamic weights, which increase with the growth of tracking errors. The hyperbolic tangent function ensures smooth weight variation without sudden mutation. Q_{33} and Q_{44} maintain constant low weights to balance core tracking objectives and vehicle stability.

DOI: 10.54254/2753-8818/2026.34703

The control weight R is negatively correlated with the state weight. When the error is large and the vehicle speed is low, R decreases to allow greater steering correction; when the error decreases, R rises to suppress excessive steering action, ensuring that the control input is always within a reasonable range and avoiding actuator saturation.

5. Conclusion

This paper proposes an adaptive weight MPC algorithm based on joint perception of tracking error and longitudinal velocity, which solves the defect that traditional fixed-weight MPC cannot adapt to dynamic changes in vehicle speed and tracking error. Simulation results show that the proposed algorithm reduces the lateral error convergence time by 48.5%, the maximum lateral tracking error by 39.1%, and the maximum sideslip angle by 31.3%. It improves trajectory tracking accuracy, optimizes vehicle yaw and sideslip characteristics, and enhances comprehensive driving stability.

In future research, actuator dynamic characteristics and road adhesion coefficient variation will be further considered to construct a more robust adaptive MPC framework and expand the application scope of the algorithm under complex road conditions.

References

[1] Kou, F. R., Zhang, K., & Li, G. (2023). Path tracking control of unmanned vehicles based on state extended MPC and steering angle compensation. Mechanical Science and Technology for Aerospace Engineering, 42(7), 1082-1088.

[2] Zhang, H., Li, Y., & Wu, G. (2025). Tube-MPC Based Path Tracking for I Vehicle Considering Data-Driven Stable Region and External Disturbances. SAE Technical Papers. https: //doi.org/10.4271/2025-01-8262

[3] Huang, S., Sun, D., & Zhao, M. (2024). Distributed MPC-Based Hierarchical Cooperative Control for Mixed Vehicle Groups with T-CPS in the Vicinity of Traffic Signal Light. IEEE Transactions on Intelligent Transportation Systems, 25(7), 8003–8016. https: //doi.org/10.1109/TITS.2024.3390763

[4] Tang, C., Li, T., Li, T., Ma, J., & Xiong, H. (2024). All-Wheel Steering Path Tracking Control for Multi-Axle Vehicle Based on Adaptive MPC. 2024 4th International Conference on Computer, Control and Robotics, ICCCR 2024, 183–190. https: //doi.org/10.1109/ICCCR61138.2024.10585367

[5] Ge, S., Bai, Z., Wang, J., Luo, X., Li, X., Fang, X., & Yuan, H. (2024). Tube-Based Distributed MPC for Heterogeneous Vehicle Platoons Subject to External Disturbances. Proceedings - 2024 China Automation Congress, CAC 2024, 4352–4357. https: //doi.org/10.1109/CAC63892.2024.10864501

[6] Pang, L., Yue, M., Qi, G., & Zhao, X. (2025). Switched control strategy based on preview-based MPC for active suspension of heavy-duty vehicle to improve ride comfort and roll stability. Measurement Science and Technology, 36(3). https: //doi.org/10.1088/1361-6501/adafce

[7] Cen, S., Bai, Z., Wang, J., Luo, X., Li, X., Fang, X., & Yuan, H. (2024). Distributed MPC for Nonlinear Networked Vehicle Platoon System with Communication Delays. Proceedings - 2024 China Automation Congress, CAC 2024, 1576–1581. https: //doi.org/10.1109/CAC63892.2024.10865719

[8] Yang, K., Wang, H., Wang, Y., Wu, Y., & Zuo, Z. (2024). Sequential-Parallel Hybrid Distributed MPC for Vehicle Platoon with Integrated Cartesian-Frcnct Kinematic Model. 14th Asian Control Conference, ASCC 2024, 214–219.

[9] Wang, T., Yang, X., Ma, M., Cen, H., & Deng, Z. (2025). Game theory-based MPC control strategy for path following of reconfigurable unmanned ground vehicle: A multi-agent control method. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering. https: //doi.org/10.1177/09544070241312954

[10] Zhou, Z., Tao, M., Qiu, J., Zhang, P., Xu, M., & Chen, J. (2025). Autonomous Vehicle Path Tracking Using Event-Triggered MPC With Switching Model: Methodology and Real-World Validation. IET Control Theory and Applications, 19(1). https: //doi.org/10.1049/cth2.70046

[11] Zhang, C., & Wu, M. (2023). Trajectory Tracking Control of Automatic Guided Soft Target Vehicle Based on MPC. Proceedings - 2023 2nd International Symposium on Control Engineering and Robotics, ISCER 2023, 18–21. https: //doi.org/10.1109/ISCER58777.2023.00009