Learning-Augmented Tube Model Predictive Control for Autonomous Vehicle Trajectory Tracking

Bohao He[1] and Ling Zheng[2]

***Abstract*— Trajectory tracking for autonomous ground vehicles near handling limits is affected by tire nonlinearities and transient changes in road friction. Tube-based robust model predictive control (MPC) can maintain constraint satisfaction, but large worst-case disturbance bounds may lead to conservative steering commands. This paper proposes a learning-augmented tube MPC strategy in which a feedforward deep neural network (DNN) estimates the residual mismatch between a nominal linear time-varying vehicle model and the nonlinear plant. A first-order low-pass filter is applied to the residual sequence before it is used over the prediction horizon. The tube is then constructed from the bounded filtered prediction error, and an LQR feedback law regulates the remaining residual error. In two simulation scenarios, the proposed Tube-DNN-MPC reduces lateral RMSE by 71.8% and 51.7% relative to Tube-MPC, while the average computation time remains below the 50-ms sampling interval used in the tests.**

I. Introduction

Ensuring safe and accurate trajectory tracking is a key requirement for autonomous ground vehicles operating near handling limits [1]. In this paper, autonomous vehicles (AVs) refer to road vehicles rather than aerial platforms; related tube-MPC studies on UAVs address different uncertainty sources and are therefore outside the present vehicle-dynamics scope. During high-speed cornering or maneuvers on low-friction road surfaces, vehicle dynamics exhibit pronounced nonlinear behavior. Unmodeled chassis responses, rapid load transfers, and tire-force saturation induce discrepancies between simplified predictive models and the physical plant [2]. These mismatches can degrade path-tracking accuracy and driving stability [3].

To manage bounded external disturbances and internal model mismatches, robust control strategies, including sliding mode control [4] and robust Model Predictive Control (MPC) [5], have been widely studied for handling bounded disturbances and model mismatch. Specifically, Tube-based MPC (TMPC) mathematically confines operational uncertainties within a precomputed robust positive invariant set, ensuring strict constraint satisfaction [6]. However, conventional TMPC designs rely on offline-determined, worst-case physical bounds. This rigidity can introduce conservatism, reduce the available control authority, and limit tracking performance during transient maneuvers [7].

[1]Bohao He is with the College of Mechanical and Vehicle Engineering, Chongqing University, Chongqing 400044, China (E-mail: bhhe@stu.cqu.edu.cn)

[2]Ling Zheng is with the National Key Laboratory of High-end Equipment Mechanical Transmission and the College of Mechanical and Vehicle Engineering, Chongqing University, Chongqing 400044, China (E-mail: zling@cqu.edu.cn)

Learning-based models have been introduced into model-based vehicle control to improve prediction accuracy under nonlinear tire-road conditions [8]–[10]. However, when a learned model is directly embedded in MPC, two issues remain important: the learned residual sequence should not introduce high-frequency numerical fluctuations, and the remaining prediction error should be explicitly connected to the tightened constraints used by robust MPC.

Compared with NN-MPC formulations that mainly use learned dynamics to improve nominal prediction [9], [10], this work keeps a linear QP structure and uses the filtered residual prediction error to size the tube. The contributions are two-fold. First, a nominal speed-varying linear vehicle model is augmented by a feedforward DNN that predicts residual dynamics along the MPC horizon, while a discrete low-pass filter smooths the residual sequence used by the optimizer. Second, the tube is constructed from the bounded filtered prediction error rather than from a broad physical disturbance envelope. This reduces conservatism in the tested scenarios while retaining the usual tube-MPC constraint-tightening structure under the assumed residual-error bound.

Notation

Scalars, vectors, and matrices are denoted by lower-case, bold-free lower-case, and upper-case symbols, respectively. Predicted nominal variables use an overbar, e.g., $\bar{x}$, while residual variables use a tilde, e.g., $\tilde{x} = x - \bar{x}$. Estimated quantities are marked by a hat. The operators $\oplus$ and $\ominus$ denote Minkowski summation and Pontryagin set difference. Calligraphic symbols denote constraint or error sets, and stacked prediction-horizon vectors are denoted by capital letters.

II. Nominal Vehicle Dynamics Modeling

A. Vehicle Dynamics and Preview Kinematics

For computational tractability in real-time predictive optimization, a 3-DOF single-track vehicle model is adopted. Longitudinal velocity tracking is independently governed by a discrete Proportional-Integral-Derivative (PID) controller, directing the MPC framework to focus entirely on lateral displacement and yaw stabilization. The lateral governing equations are defined as:

$$\dot{v}_y + r v_x = \frac{F_{yf} + F_{yr}}{m}, \tag{1a}$$

$$\dot{r} = \frac{F_{yf} l_f - F_{yr} l_r}{I_z}, \tag{1b}$$

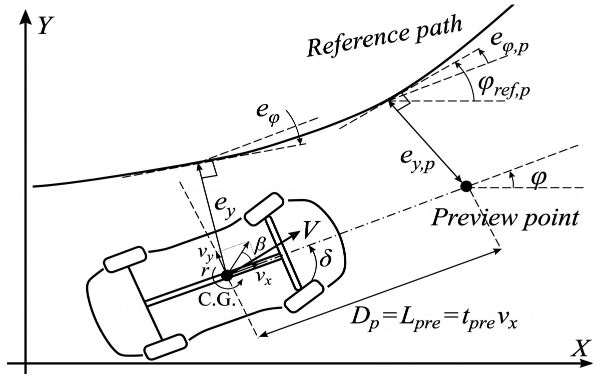


Fig. 1. Schematic of the single-track vehicle model and preview-based trajectory tracking errors.

where v_x and v_y denote the longitudinal and lateral velocities; r represents the yaw rate. The vehicle mass is m, and I_z characterizes the yaw moment of inertia. The geometric parameters l_f and l_r define the distances from the center of mass to the front and rear axles. The lateral tire forces F_{yf} and F_{yr} rely on linear approximations $F_{yf} = C_f \alpha_f$ and $F_{yr} = C_r \alpha_r$, with C_f, C_r denoting the nominal cornering stiffness coefficients.

To execute precise trajectory tracking, the dynamic equations are coupled with a preview-based kinematic model, as illustrated in Fig. 1. The control objective targets a preview point located at a longitudinal distance $L_{pre} = t_{pre} v_x$ ahead of the vehicle's center of mass, where t_{pre} is the preview time. The lateral tracking error $e_{y,p}$ and heading error $e_{\varphi,p}$ at this specific preview point evolve according to the following differential equations:

$$\dot{e}_{y,p} = v_y + v_x e_{\varphi,p}, \tag{2a}$$

$$\dot{e}_{\varphi,p} = r - v_x \rho, \tag{2b}$$

where ρ defines the continuous path curvature evaluated at the preview location.

B. State-Space Formulation

Defining the tracking state vector as $x = [e_{y,p}, e_{\varphi,p}, v_y, r]^\top$ and the control input as $u = \delta_f$, the continuous-time LTV state-space representation $\dot{x} = A_c x + B_c u$ is formulated with the following Jacobians:

$$A_c = \begin{bmatrix} 0 & v_x & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{C_f + C_r}{m v_x} & \frac{C_f l_f - C_r l_r}{m v_x} - v_x \\ 0 & 0 & \frac{C_f l_f - C_r l_r}{I_z v_x} & \frac{C_f l_f^2 + C_r l_r^2}{I_z v_x} \end{bmatrix}, \tag{3}$$

$$B_c = \begin{bmatrix} 0 & 0 & -\frac{C_f}{m} & -\frac{C_f l_f}{I_z} \end{bmatrix}^\top. \tag{4}$$

Discretization is executed via the zero-order hold method at a sampling interval T_s, yielding the baseline discrete nominal system:

$$\bar{x}_{k+1} = A_d \bar{x}_k + B_d \bar{u}_k. \tag{5}$$

This linear approximation inevitably deviates from the physical plant under high lateral accelerations, dictating the necessity for deep residual compensation.

III. Deep Residual Modeling

A. Network Architecture and Error Bounding

The absolute evolution of the vehicle tracking states decomposes into the known nominal LTV component and the unknown nonlinear residual disturbance w_k:

$$x_{k+1} = A_d x_k + B_d u_k + w_k. \tag{6}$$

The residual term w_k represents the part of the one-step state transition not captured by the nominal LTV model, including tire-force saturation, transient load transfer, and friction variation. A feedforward DNN is trained to map the operating condition to the residual compensation vector $\hat{w}_{\text{DNN}}$.

The input feature vector $\xi_k \in \mathbb{R}^7$ includes the variables that directly affect the residual dynamics in the simulations:

$$\xi_k = [e_{y,p}, e_{\varphi,p}, v_y, r, \delta_f, v_x, \mu]^\top. \tag{7}$$

In the implementation used for the reported results, the residual model is an MLP with $L = 3$ hidden layers and 256 neurons per layer. ReLU activations are used in the hidden layers, and a linear output layer returns the four-dimensional residual vector. The inputs and residual targets are standardized using the training-set mean and standard deviation. The network weights Θ are optimized with Adam using a learning rate of $\eta = 0.001$, a batch size of $B = 256$, and $N_{\text{epoch}} = 1000$ training epochs. These details clarify that "deep" refers to multiple hidden nonlinear layers, not to a recurrent or convolutional architecture. The training loss is

$$\mathcal{L}(\Theta) = \frac{1}{B} \sum_{j=1}^{B} \| w_j - \hat{w}_{\text{DNN}}(\xi_j) \|^2, \tag{8}$$

where B denotes the batch size.

B. Low-Pass Filtering and Sequence Evolution

In preliminary simulations, directly using the raw DNN output in the prediction loop led to less smooth control increments. A discrete first-order low-pass filter (LPF) is therefore applied to the residual prediction:

$$\hat{w}_f(k) = \alpha \hat{w}_f(k-1) + (1-\alpha) \hat{w}_{\text{DNN}}(k), \tag{9}$$

where $\alpha \in [0, 1)$ is the filter coefficient, and $\hat{w}_f$ denotes the filtered residual compensation vector. The filter attenuates high-frequency components in the learned residual sequence; closed-loop robustness is handled separately through the residual tube and LQR feedback in Section IV.

The filtered residual sequence is projected along the prediction horizon N_p. For $i \in \{0, 1, \ldots, N_p - 1\}$, the augmented nominal state evolves as

$$\bar{x}_{k+i+1|k} = A_d \bar{x}_{k+i|k} + B_d \bar{u}_{k+i|k} + \hat{w}_f(k+i|k). \tag{10}$$

To evaluate $\xi_{k+i|k}$ without making the QP nonlinear, the required future states are taken from the shifted optimal trajectory computed at the previous control step. Thus, $\hat{W}_{f,k}$ is fixed before the current QP is solved and enters the optimization as an affine term. The accuracy of this sequence is assessed through the filtered residual error on the validation/test data, which is later used to define $\mathcal{W}_{\text{err}}$.

IV. Control Strategy Formulation

A. LQR-Based Residual Stabilization

Defining the residual state error between the actual physical plant and the hybrid prediction as $\tilde{x}_k = x_k - \bar{x}_k$, the error dynamics are isolated by subtracting (10) from (6):

$$\tilde{x}_{k+1} = A_d\tilde{x}_k + B_d\tilde{u}_k + \tilde{w}_k, \tag{11}$$

where $\tilde{w}_k = w_k - \hat{w}_f$ is the remaining prediction error after filtering. Its componentwise empirical bound w_{err} is estimated from the filtered residual error on the validation/test set. To regulate this residual error, an LQR feedback gain is designed offline from the nominal matrices:

$$K = -(R + B_d^\top PB_d)^{-1}B_d^\top PA_d, \tag{12}$$

where $P \succ 0$ solves the discrete algebraic Riccati equation. Under the standard stabilizability assumption for (A_d, B_d) and positive-definite input weighting, the resulting closed-loop matrix $A_K = A_d + B_dK$ is Schur stable for the local linear model.

B. Optimization Problem Design

The disturbance boundary established by the DNN's maximum prediction error constructs a fixed zonotopic set $\mathcal{W}_{\text{err}} = \{w \in \mathbb{R}^{n_x} : |w| \le w_{\text{err}}\} = \langle 0, G_{w_{\text{err}}}\rangle$. Driven by the stabilizing gain K, the residual reachable set $\tilde{\mathcal{X}}$ over the horizon expands via Minkowski summation:

$$\tilde{\mathcal{X}}_{k+i|k} = \bigoplus_{j=0}^{i-1} A_K^j\mathcal{W}_{\text{err}}. \tag{13}$$

The state and input constraints are subsequently tightened using Pontryagin set differences to maintain constraint satisfaction under the assumed residual-error bound: $\bar{\mathcal{X}}_{k+i|k} = \mathcal{X} \ominus \tilde{\mathcal{X}}_{k+i|k}$ and $\bar{\mathcal{U}}_{k+j|k} = \mathcal{U} \ominus K\tilde{\mathcal{X}}_{k+j|k}$.

Let $\bar{y}_{k+i|k} = C_d\bar{x}_{k+i|k}$ denote the controlled output, where C_d selects the output states used in the tracking cost. In the implementation, $\bar{y}_{k+i|k}$ contains the lateral tracking error, heading error, and lateral-velocity-related component. The core optimization algorithm computes the optimal nominal control sequence $\bar{U}_k$ by minimizing the output tracking error, the control input, and the control increment:

$$\min_{\bar{U}_k} \quad \|\bar{Y}_k - Y_{\text{ref}}\|^2_{\mathcal{Q}_y} + \|\bar{U}_k\|^2_{\mathcal{R}_u} + \|\Delta\bar{U}_k\|^2_{\mathcal{R}_{\Delta u}} \tag{14a}$$

$$\text{s.t.} \quad \bar{x}_{k+i+1|k} = A_d\bar{x}_{k+i|k} + B_d\bar{u}_{k+i|k} + \hat{w}_f(k+i|k) \tag{14b}$$

$$\bar{x}_{k+i|k} \in \bar{\mathcal{X}}_{k+i|k}, \;\; \bar{u}_{k+j|k} \in \bar{\mathcal{U}}_{k+j|k}. \tag{14c}$$

The steering signal applied to the vehicle actuators superimposes the MPC optimization result and the LQR stabilization feedback: $u_k = \bar{u}_k^* + K(x_k - \bar{x}_k)$.

C. Quadratic Programming Transformation

To facilitate real-time feasibility in the tested setup, the MPC formulation is algebraically transformed into a standard Quadratic Programming (QP) problem. Over the prediction horizon N_p, the augmented state evolution sequence is explicitly lifted into the following global matrix formulation:

$$\bar{X}_k = \mathcal{A}\bar{x}_k + \mathcal{B}\bar{U}_k + \mathcal{B}_w\hat{W}_{f,k} \tag{15}$$

where $\bar{X}_k \in \mathbb{R}^{N_pn_x}$ and $\bar{U}_k \in \mathbb{R}^{N_cn_u}$ represent the stacked state and input vectors. The filtered DNN prediction sequence $\hat{W}_{f,k} = [\hat{w}_f(k|k)^T, \dots, \hat{w}_f(k+N_p-1|k)^T]^T$ is propagated recursively through the system via the lower-block triangular matrix $\mathcal{B}_w$:

$$\mathcal{B}_w = \begin{bmatrix} I & 0 & \dots & 0 \\ A_d & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_d^{N_p-1} & A_d^{N_p-2} & \dots & I \end{bmatrix} \tag{16}$$

Since the optimization explicitly targets the control increment sequence $\Delta\bar{U}_k$ to enforce actuator smoothness, the absolute control input is mapped via $\bar{U}_k = \Gamma\Delta\bar{U}_k + U_{k-1}$, where Γ is a lower triangular block matrix of identity matrices, and U_{k-1} expands the previous control action. Let $\mathcal{C}_y = I_{N_p} \otimes C_d$ be the stacked output-selection matrix. The stacked output prediction is then written as

$$\bar{Y}_k = \mathcal{A}_y\bar{x}_k + \mathcal{B}_y\bar{U}_k + \mathcal{B}_{w,y}\hat{W}_{f,k}, \tag{17}$$

where $\mathcal{A}_y = \mathcal{C}_y\mathcal{A}$, $\mathcal{B}_y = \mathcal{C}_y\mathcal{B}$, and $\mathcal{B}_{w,y} = \mathcal{C}_y\mathcal{B}_w$. The cost function is therefore translated into the canonical QP form:

$$\min_{\Delta\bar{U}_k} \frac{1}{2}\Delta\bar{U}_k^T H\Delta\bar{U}_k + f^T\Delta\bar{U}_k \tag{18}$$

The Hessian matrix H and the gradient vector f are derived as:

$$H = 2\big(\Gamma^T(\mathcal{B}_y^T\mathcal{Q}_y\mathcal{B}_y + \mathcal{R})\Gamma + \mathcal{R}_\Delta\big) \tag{19a}$$

$$f = 2\Gamma^T\big[\mathcal{B}_y^T\mathcal{Q}_y(\mathcal{A}_y\bar{x}_k + \mathcal{B}_{w,y}\hat{W}_{f,k} - Y_{\text{ref}}) + (\mathcal{B}_y^T\mathcal{Q}_y\mathcal{B}_y + \mathcal{R})U_{k-1}\big] \tag{19b}$$

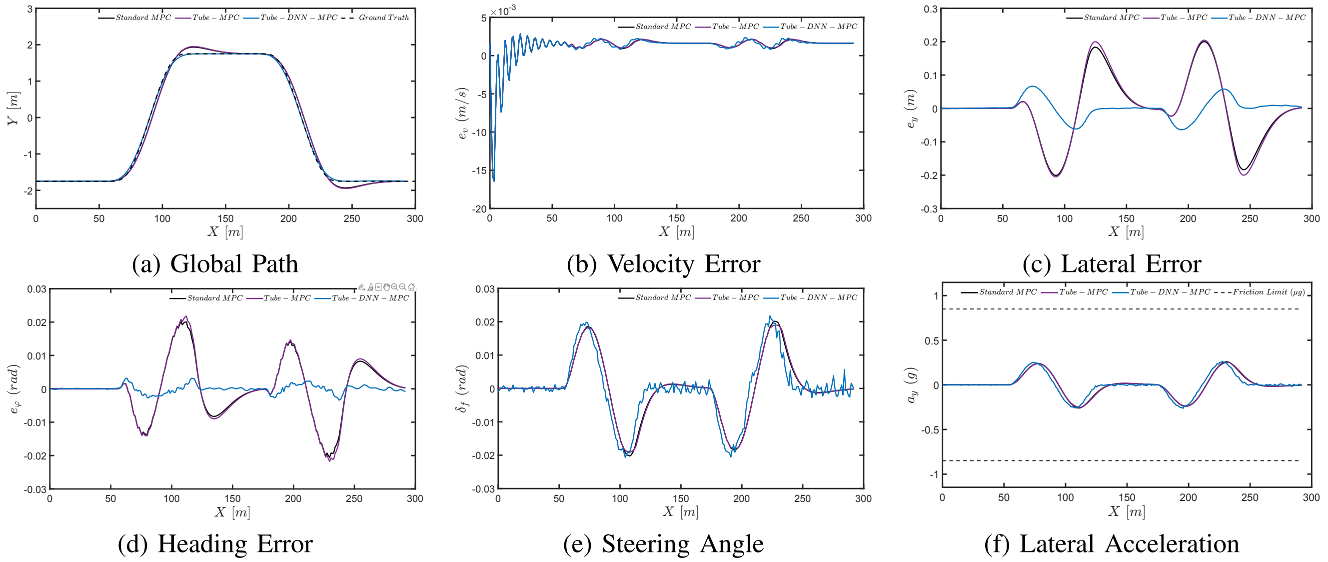
where $\mathcal{Q}_y$, $\mathcal{R}$, and $\mathcal{R}_\Delta$ are the block-diagonal weight matrices, and Y_{ref} defines the previewed reference sequence. This algebraic transformation embeds the DNN-predicted nonlinear dynamics ($\hat{W}_{f,k}$) directly into the linear gradient vector f.

V. Simulation Validation

To evaluate the tracking improvements, the proposed Tube-DNN-MPC is compared with a standard nominal MPC (Std MPC) and a Tube-MPC without DNN residual compensation.

A. Implementation Setup

The vehicle parameters and MPC tuning weights used in the closed-loop tests are summarized in Table I. The discrete sampling interval is $T_s = 0.05$ s and the prediction and control horizons satisfy $N_p = N_c = 10$, so each control step is solved within the same horizon length. The residual DNN and the LPF used in (9) adopt the hyperparameters listed in Table II; the training dataset is collected from a higher-fidelity nonlinear vehicle simulator covering the velocity and friction ranges considered in the tests, and the network is



(a) Global Path (b) Velocity Error (c) Lateral Error

(d) Heading Error (e) Steering Angle (f) Lateral Acceleration

Fig. 2. Tracking performance and vehicle state responses under conventional handling conditions (80 km/h, $\mu = 0.85$).

TABLE I

VEHICLE AND MPC TUNING PARAMETERS

Symbol	Value	Symbol	Value
m	1530 kg	T_s	0.05 s
l_f	1.10 m	N_p	10
l_r	1.68 m	N_c	10
I_z	2315.3 kg·m^2	Q_0	diag(50, 50, 1)
C_f	-5.0×10^4 N/rad	$\mathcal{R}_u$	1200 I_{N_c}
C_r	-5.0×10^4 N/rad	$\mathcal{R}_{\Delta u}$	20 I_{N_c}

$\mathcal{Q}_y = I_{N_p} \otimes Q_0$; $\otimes$ denotes the Kronecker product.

TABLE II

RESIDUAL DNN AND FILTER HYPERPARAMETERS

Hyperparameter	Value	Hyperparameter	Value
Architecture	MLP	Optimizer	Adam
Hidden layers L	3	Learning rate η	1×10^{-3}
Neurons/layer	256	Batch size B	256
Hidden activation	ReLU	Training epochs	1000
Output activation	Linear	LPF coefficient α	0.9
Input dim	7	Residual dim	4

trained offline. During closed-loop evaluation, the DNN is queried once per control step to construct the filtered residual sequence $\hat{W}_{f,k}$ used in the QP.

B. Evaluation Metrics

The root mean square error (RMSE) and maximum absolute error are used to evaluate lateral displacement (e_y) and heading error (e_φ).

C. Conventional Operating Condition

Under conventional conditions, the vehicle tracks a double-lane change trajectory at 80 km/h on a high-adhesion surface ($\mu = 0.85$). The temporal vehicle states are plotted in Fig. 2, and the comprehensive numerical results are documented in Table III.

TABLE III

TRACKING PERFORMANCE (CONVENTIONAL: 80 KM/H, $\mu = 0.85$)

Metric	Std MPC	Tube-MPC	**Tube-DNN-MPC**
RMSE e_y (10^{-2} m)	9.70	10.23	**2.88**
Max e_y (10^{-2} m)	20.06	20.49	**6.65**
RMSE e_φ (10^{-2} rad)	0.84	0.89	**0.11**
Avg. Time (ms)	6.45	9.97	24.02

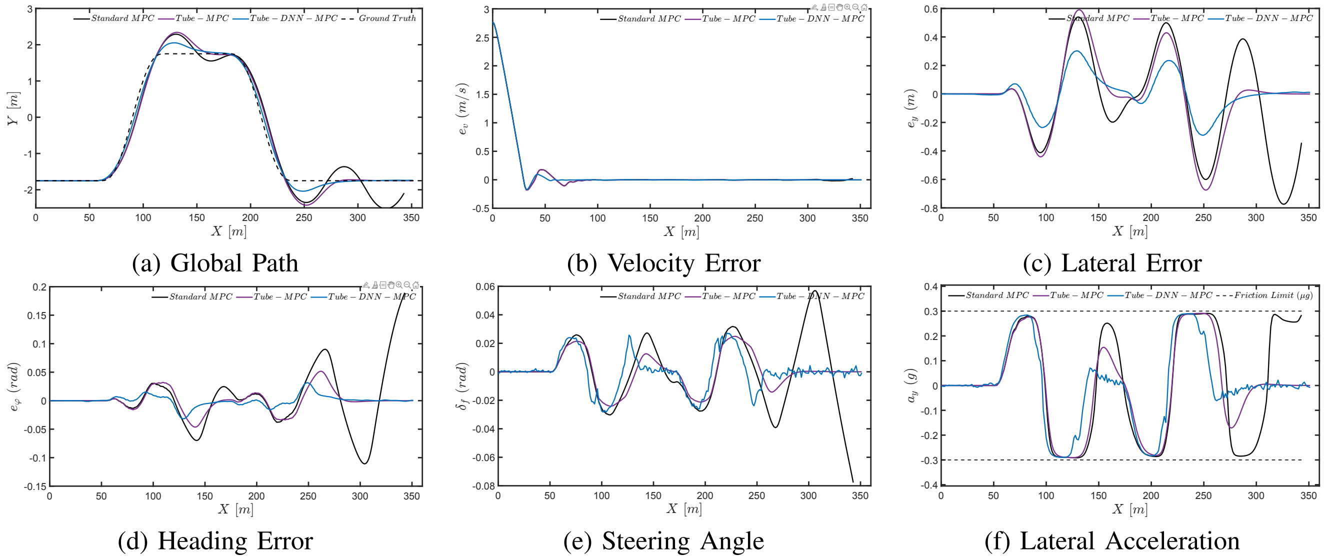
Table III indicates that both LTV-based controllers have non-negligible tracking errors under this condition. Tube-MPC yields a lateral RMSE of 10.23×10^{-2} m, whereas Tube-DNN-MPC reduces it to 2.88×10^{-2} m, corresponding to a 71.8% reduction relative to Tube-MPC. The maximum lateral error is also reduced from 20.49×10^{-2} m to 6.65×10^{-2} m.

The LPF helps regularize the learned residual sequence, and no visibly oscillatory steering response is observed in this scenario. The average execution time is 24.02 ms, which remains below the 50-ms sampling interval used in the simulations.

D. Extreme Operating Condition

To push the vehicle dynamics to their handling limits, an extreme test is executed at 100 km/h with a severely reduced friction coefficient ($\mu = 0.3$). This extreme regime induces tire saturation and intensive lateral load transfers. Figure 3 illustrates the dynamic responses.

Table IV shows that the low-friction case is more challenging for the two baseline controllers. The standard nominal MPC reaches a maximum lateral deviation of 77.46×10^{-2} m, while Tube-MPC reduces this value to 67.47×10^{-2} m and obtains a lateral RMSE of 26.36×10^{-2} m. Tube-DNN-MPC further reduces the lateral RMSE to 12.72×10^{-2} m, a



(a) Global Path (b) Velocity Error (c) Lateral Error

(d) Heading Error (e) Steering Angle (f) Lateral Acceleration

Fig. 3. Tracking performance and vehicle state responses under extreme limit-handling conditions (100 km/h, $\mu = 0.3$).

TABLE IV
TRACKING PERFORMANCE (EXTREME: 100 KM/H, $\mu = 0.3$)

Metric	Std MPC	Tube-MPC	**Tube-DNN-MPC**
RMSE e_y (10^{-2} m)	33.15	26.36	**12.72**
Max e_y (10^{-2} m)	77.46	67.47	**30.26**
RMSE e_φ (10^{-2} rad)	5.08	1.90	**0.95**
Avg. Time (ms)	5.81	10.36	25.15

51.7% reduction relative to Tube-MPC. The heading RMSE decreases from 1.90×10^{-2} rad to 0.95×10^{-2} rad, and the maximum lateral error decreases from 67.47×10^{-2} m to 30.26×10^{-2} m. The steering-angle and lateral-acceleration responses remain bounded in simulation, and the average execution time is 25.15 ms, below the 50-ms sampling interval.

VI. CONCLUSION

This paper presented a Tube-DNN-MPC framework for autonomous ground-vehicle trajectory tracking under nonlinear tire-road effects. The DNN residual model improves the nominal LTV prediction, while the low-pass filter smooths the residual sequence used by the QP. The remaining filtered prediction error is used to construct the tube, and an LQR feedback law regulates the residual state error under the assumed error bound. In the two simulated scenarios, the method reduced lateral RMSE by 71.8% and 51.7% relative to Tube-MPC, with average computation times below the 50-ms sampling interval. Future work will evaluate the residual-error bound under broader road-friction variations and experimental vehicle tests.

REFERENCES

[1] J. Zhao *et al.*, "Autonomous driving system: A comprehensive survey," *Expert Systems with Applications*, vol. 242, 2024.

[2] K. Yang *et al.*, "Uncertainties in onboard algorithms for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 8963–8987, 2023.

[3] P. Stano *et al.*, "Model predictive path tracking control for automated road vehicles: A review," *Annual Reviews in Control*, vol. 55, pp. 194–236, 2023.

[4] G. Tagne *et al.*, "Higher-order sliding mode control for lateral dynamics of autonomous vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 678–683.

[5] X. Wu *et al.*, "Fault-tolerant control for path-following of independently actuated autonomous vehicles using tube-based model predictive control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20 282–20 297, 2022.

[6] S. Mata *et al.*, "Robust tube-based model predictive control for lateral path tracking," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 4, pp. 569–577, 2019.

[7] H. Zheng *et al.*, "Varying zonotopic tube RMPC with switching logic for lateral path tracking of autonomous vehicle," *Journal of the Franklin Institute*, vol. 359, no. 7, pp. 2759–2787, 2022.

[8] A. M. Bokor *et al.*, "Tube-based model predictive control with random Fourier features for nonlinear systems," *arXiv preprint arXiv:2511.16425*, 2025.

[9] M. Rokonuzzaman *et al.*, "Model predictive control with learned vehicle dynamics for autonomous vehicle path tracking," *IEEE Access*, vol. 9, pp. 128 233–128 249, 2021.

[10] N. A. Spielberg *et al.*, "Neural network model predictive motion control applied to automated driving with unknown friction," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 5, pp. 1934–1945, 2021.