

Article

Coordinated Control of Intelligent Vehicle Stability and Trajectory Tracking Based on Stability-Region Identification

Danhua Chen ^{1,2,3}, Jie Hu ^{1,2}, Yuting Liu ^{1,2}, Kaige Shen ^{1,2}, Tie Xu ^{1,2,3}, Yuanyi Huang ³ and Pei Zhang ^{1,2,*} 

¹ State Key Laboratory of Light Superalloys, Wuhan University of Technology, Wuhan 430070, China; danhua.chen@sgmw.com.cn (D.C.); auto_hujie@whut.edu.cn (J.H.); 347804@whut.edu.cn (Y.L.); shenkaige@whut.edu.cn (K.S.); tie.xu@sgmw.com.cn (T.X.)

² Hubei Key Laboratory of Advanced Technology for Automotive Components, Wuhan University of Technology, Wuhan 430070, China

³ SAIC GM-Wuling Automobile Co., Ltd., Liuzhou 545007, China; yuanyi.huang@sgmw.com.cn

* Correspondence: zhangpei@whut.edu.cn

Abstract

To solve the conflict between trajectory tracking and stability control for distributed drive electric vehicles under complex driving conditions, an integrated longitudinal and lateral coordinated control strategy based on a hierarchical architecture is proposed in this paper. First, a stability-boundary dataset is constructed using an improved sum-of-squares programming (ISOSP) algorithm. Then, a long short-term memory (LSTM) network is optimized with the sparrow search algorithm (SSA). Finally, a prediction model is established to identify the dynamic stability region in real time. A hierarchical architecture is adopted for the control strategy. The upper layer integrates longitudinal–lateral tracking and stability control to describe the desired motion states accurately. In the middle layer, a stability margin is defined based on the stability region, and a risk factor is introduced to reconstruct the optimization objective. Through this design, the coordinated control of trajectory tracking and vehicle stability is achieved. In the lower layer, the minimization of the tire-workload rate is taken as the objective, and the optimal allocation of four-wheel torque is realized through quadratic programming. Hardware-in-the-loop (HIL) tests based on an NI PXIe-1078 real-time simulator, a host computer, and a domain controller indicate that the proposed strategy achieves good control performance under both variable-speed high-adhesion and high-speed low-adhesion double lane change (DLC) conditions. Especially in the extreme condition of high speed and low adhesion, the root-mean-square errors (RMSEs) of lateral displacement and sideslip angle are controlled within 0.4125 m and 0.0365 rad, respectively. Consequently, the high-precision tracking capability and real-time stability maintenance of the coordinated control strategy under extreme conditions are successfully verified.

Keywords: distributed drive vehicles; stability region; trajectory tracking; stability; sum-of-squares programming

MSC: 93C10



Received: 12 June 2026

Revised: 16 July 2026

Accepted: 23 July 2026

Published: 6 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The structure of distributed drive electric vehicles (DDEVs) is shown in Figure 1. Each wheel of a DDEV is equipped with an independently controllable in-wheel or wheel-side motor, which provides inherent advantages for actuator-level motion control in intelligent

vehicles [1]. However, under complex and extreme driving conditions, such as high speed and low adhesion, the upper tracking controller lacks the perception of physical limits and the vehicle stability region, and the generated control commands easily exceed the safety boundary. Meanwhile, the lower stability controller is passively activated, and the intervention of the lower-level stability controller may degrade trajectory tracking performance [2]. Therefore, the construction of a coordinated control mechanism that can identify vehicle states with high precision and ensure the fast response of the lower layer is urgently needed. Currently, the main research methods for the stability region include the phase plane method and the Lyapunov function method.

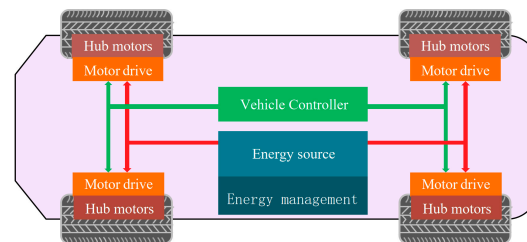


Figure 1. Schematic diagram of the DDEV structure.

The phase plane method provides an intuitive and scientific criterion for the extraction of the vehicle stability region by monitoring the dynamic evolution of the system phase trajectory. Various types of phase planes have been employed, including “sideslip angle–yaw rate” [3], “yaw rate–lateral velocity” [4], “sideslip angle–sideslip angular velocity” [5], “front tire slip angle–rear tire slip angle” [6], and “lateral acceleration–longitudinal acceleration” [7]. To achieve the parameterized expression of the stability boundary, numerous scholars have proposed multiple geometric division schemes, such as the parallel line method [8], dual line method [9], polyline method [10], diamond method [11], elliptic method [12], and circular method [13]. Different categories of phase planes correspond to distinct division methods [14]. Although current phase-plane-based vehicle stability-region division methods are diverse, how to ensure the accuracy of boundary description while maintaining simplicity and efficiency in the partitioning process remains a bottleneck in existing research.

The Lyapunov criterion solves the stability boundaries of low-degree-of-freedom systems without relying on tedious phase trajectory plotting. The region of attraction (ROA) characterizes the set of initial states that can asymptotically converge to a stable equilibrium state; therefore, it can be used to define the vehicle stability region. Lyapunov functions usually depend on manual construction, and the function form directly limits the shape and size of the estimated ROA. The SOSP algorithm improves the approximation accuracy of the stability region by introducing high-order polynomial Lyapunov functions. Tamba et al. [15] estimated the ROA of tire sideslip dynamics using SOSP combined with an approximate tire force model. Ribeiro et al. [16] applied SOSP to solve the maximum safe state-space region to ensure the lateral stability of closed-loop vehicle systems. Shen et al. [17] adopted high-order Lyapunov functions and proposed a novel dynamic circle shape function to further reduce conservatism. Hu et al. [18] optimized the computational efficiency of SOSP through feasibility pre-judgment and dynamic search strategies, but they also increased the conservatism of the estimation. With the development of data-driven modeling techniques, learning-based methods have recently been introduced to improve the online availability of vehicle dynamic stability-region information. Liu et al. [11] proposed an SVR-based dynamic stable-region regression model using vehicle phase-plane information, enabling real-time stability judgment under varying vehicle states. Li et al. [19] developed a path-tracking control strategy considering data-driven

dynamic stable-region constraints, where a stable-region dataset was constructed based on the sideslip angle–sideslip angular velocity phase plane and a BP neural network was trained offline to predict the dynamic stable-region constraints online. Wang et al. [20] further proposed a real-time multidimensional vehicle dynamic stability-domain calculation method, in which a multi-input–multi-output neural network was employed to map time-varying operating factors to stability-domain characteristics.

In the field of coordinated control of trajectory tracking and stability, Wang et al. [21] utilized an improved sliding mode control to optimize lateral control to coordinate tracking errors and stability while simplifying longitudinal motion control. However, in the nonlinear region where tire forces tend to saturate, longitudinal and lateral decoupled control is difficult to maintain while tracking accuracy and posture stability. To this end, Liu et al. [22] adopted model predictive control based on a longitudinal–lateral-coupled vehicle model to design coordinated control rules, which improved the trajectory tracking accuracy and stability under variable curvature conditions. Zhai et al. [23] proposed an integrated control strategy based on MPC, improving the trajectory tracking accuracy and handling stability of DDEVs. Zeng et al. [24] further introduced a variable-step MPC framework to balance stability and tracking accuracy by deeply mining the physical limits of longitudinal and lateral tire forces, reducing collision risks in interactive scenarios with a milder control strategy. To accurately capture the switching timing of control authority, Zhang et al. [25] introduced a normalized stability indicator, thereby designing a weight-adaptive criterion to formulate a synergistic strategy for trajectory tracking and stability control. Zhang et al. [26] proposed a dual weight-distributed cooperative sliding mode control method to achieve yaw angle tracking compensation and yaw stability control. To achieve the synergistic optimization of trajectory tracking accuracy and vehicle yaw stability, some studies also coordinated the active front steering (AFS) system and direct yaw control (DYC). Lu et al. [27] dynamically adjusted the control weights of DYC and AFS based on the stability region to realize the hierarchical regulation of actuators in stable, critical, and unstable regions, which reduced tracking errors and guaranteed stability under extreme conditions. Zhang et al. [28] investigated region stability analysis and precise tracking control for linear stochastic systems, showing that region-based stability criteria can be combined with tracking control to regulate both transient and steady-state performance.

Nevertheless, several limitations remain in the existing literature. Phase-plane-based methods are intuitive and computationally efficient, but their stability boundaries are commonly constructed using predefined geometric forms or empirical thresholds. Such boundaries may be insufficient to characterize the nonlinear, asymmetric, and operating-condition-dependent stability limits of vehicles under high-speed and low-adhesion conditions. SOSP-based methods provide a rigorous Lyapunov-theoretic framework for stability-region estimation; however, the iterative search for high-order Lyapunov functions and level sets is computationally demanding, which restricts their direct implementation in real-time vehicle control. Moreover, the conventional convergence criterion based on the variation in the shape factor may not be fully consistent with the objective of maximizing the stability-region area when the shape function and shape factor are updated simultaneously, potentially leading to conservative boundary estimation. Purely data-driven predictors can improve online computational efficiency, but the predicted stability boundary may lack physical interpretability and stability-theoretic meaning if the training labels are not derived from a Lyapunov-based formulation. Meanwhile, many existing coordinated control strategies still determine intervention timing and control authority using fixed thresholds, empirical stability indicators, or locally defined phase-plane criteria. Consequently, stability-region information is often used only for state assessment or region

division, rather than being sufficiently transformed into control-oriented variables for objective reconstruction and actuator authority coordination.

To overcome these limitations, this paper proposes a stability-region-aware hierarchical coordinated control framework that tightly integrates dynamic stability-region identification with control objective reconstruction. In the proposed framework, the ISOSP-LSTM module serves as the stability-awareness module of the controller. Specifically, ISOSP is employed offline to generate stability-region boundary parameters with Lyapunov-theoretic interpretability, while the SSA-optimized LSTM network predicts the current stability-boundary coefficients online from sequential vehicle-state information. In this way, computationally intensive online SOSP optimization is replaced by fast neural-network inference, while the stability-theoretic basis of the predicted boundary is retained. Based on the predicted stability boundary, a stability margin and an instability risk factor are constructed and embedded into the coordination layer to reconstruct the trajectory tracking and yaw stability control objectives, coordinate AFS/DYC authority, and constrain the longitudinal acceleration demand. Therefore, the proposed framework establishes a direct connection between stability-region identification and coordinated control decision-making, enabling adaptive coordination between tracking accuracy and vehicle stability under different risk levels.

The main contributions of this paper are summarized as follows:

- (1) A control-oriented vehicle model with dynamic stability-region identification is established. The predicted stability boundary provides online stability information for vehicle-state assessment and subsequent coordinated control.
- (2) A coordination mechanism based on the stability margin and risk factor is proposed. The relationship between the current vehicle state and the stability boundary is quantified as a stability margin and further mapped to an instability risk factor, providing a continuous basis for coordinating trajectory tracking and stability control.
- (3) A hierarchical coordinated control architecture is designed by integrating stability-region identification with control objective reconstruction. The risk factor is used to adjust tracking and yaw stability weights, AFS/DYC authority, and longitudinal acceleration constraints, enabling tracking-priority control under low-risk conditions and stability-priority control under high-risk conditions.

The subsequent sections of this paper are organized as follows. Section 2 establishes the vehicle dynamics model. Section 3 designs the vehicle stability-region identification model based on the ISOSP-LSTM hybrid architecture. Section 4 proposes the integrated longitudinal and lateral trajectory tracking and stability coordinated control strategy. Section 5 presents the HIL tests and discusses the test results. Finally, the conclusions are given in Section 6.

2. Vehicle Dynamics Model

To achieve the research objective of enabling the DDEV to accurately track the desired trajectory while ensuring stability, a seven-degree-of-freedom vehicle dynamics model is first established. Subsequently, the Magic Formula tire model is introduced to accurately describe the nonlinear mechanical characteristics of tires under combined slip conditions, providing a high-fidelity tire force calculation basis for the subsequent lower-level four-wheel torque distribution. Meanwhile, to satisfy the requirement of the ISOSP algorithm for a polynomial model structure, a rational tire model is further constructed, and the vehicle dynamics are rationalized, thereby laying the model foundation for the accurate estimation of the vehicle stability region.

2.1. 7-DOF Vehicle Model

A seven-degree-of-freedom model is employed to describe the longitudinal, lateral, and yaw motions of the vehicle body, as well as the rotational motions of the four wheels, as shown in Figure 2.

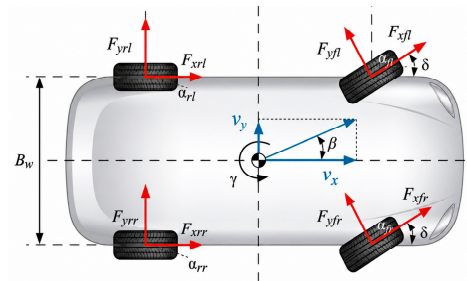


Figure 2. Seven-degree-of-freedom vehicle dynamics model.

Based on the Newton–Euler equations, the differential equations of vehicle dynamics are established:

$$\begin{cases} (F_{xfl} + F_{xfr}) \cos \delta - (F_{yfl} + F_{yfr}) \sin \delta + F_{xrl} + F_{xrr} = ma_x \\ (F_{xfl} + F_{xfr}) \sin \delta + (F_{yfl} + F_{yfr}) \cos \delta + F_{yrl} + F_{yrr} = ma_y \\ \left[(F_{xfl} + F_{xfr}) \sin \delta + (F_{yfl} + F_{yfr}) \cos \delta \right] L_f - (F_{yrl} + F_{yrr}) L_r + \\ \left[(F_{xfr} - F_{xfl}) \cos \delta + (F_{yfl} - F_{yfr}) \sin \delta \right] \frac{B_w}{2} + (F_{xrr} - F_{xrl}) \frac{B_w}{2} = I_z \dot{\gamma} \end{cases} \quad (1)$$

where L_f and L_r denote the distances from the center of gravity to the front and rear axles of the vehicle, respectively; I_z represents the moment of inertia around the z-axis; m is the vehicle mass; δ is the front-wheel steering angle; γ represents the yaw rate; a_x and a_y are the longitudinal and lateral accelerations of the vehicle, respectively; B_w is the track width; F_{xij} and F_{yij} are the longitudinal and lateral forces exerted by the ground on each tire, respectively; and $ij = fl, fr, rl$, and rr represent the front-left, front-right, rear-left, and rear-right wheels of the vehicle, respectively.

The sideslip angle is an important state variable to measure vehicle lateral stability, which theoretically can be accurately described by the arctangent function of the ratio of lateral velocity to longitudinal velocity as:

$$\beta = \arctan\left(\frac{v_y}{v_x}\right) \approx \frac{v_y}{v_x} \quad (2)$$

where v_x denotes the longitudinal velocity of the vehicle, and v_y represents the lateral velocity.

The sideslip angles of the front and rear wheels exhibit a distinct asymmetric distribution characteristic when the vehicle body undergoes yaw motion, and the specific expressions can be formulated as:

$$\begin{cases} \alpha_{fl} = \arctan\left[\frac{(v_y + L_f \gamma)}{(v_x - 0.5B_w \gamma)}\right] - \delta \\ \alpha_{fr} = \arctan\left[\frac{(v_y + L_f \gamma)}{(v_x + 0.5B_w \gamma)}\right] - \delta \\ \alpha_{rl} = \arctan\left[\frac{(v_y - L_r \gamma)}{(v_x - 0.5B_w \gamma)}\right] \\ \alpha_{rr} = \arctan\left[\frac{(v_y - L_r \gamma)}{(v_x + 0.5B_w \gamma)}\right] \end{cases} \quad (3)$$

where α_{fl} , α_{fr} , α_{rl} , and α_{rr} denote the sideslip angles of the front-left, front-right, rear-left, and rear-right tires, respectively.

2.2. Tire Model

2.2.1. Magic Formula Tire Model

As the unique mechanical interface between the vehicle and the road surface, the nonlinear characteristics of tires determine the dynamic distribution of forces and the handling stability of the entire vehicle. The Magic Formula tire model is widely applied due to its high accuracy and concise form, and the general expressions of this model for calculating longitudinal and lateral forces can be expressed as:

$$\begin{cases} y(x) = D \sin\{\text{Carctan}(BX - E(BX - \arctan(BX)))\} \\ Y(x) = y(x) + S_v \\ X = x + S_h \end{cases} \quad (4)$$

where Y is the tire longitudinal force or lateral force; x is the corresponding longitudinal slip ratio or sideslip angle; B , C , D , and E represent the stiffness, shape, peak, and curvature factors, respectively; S_v is the vertical shift, which is assumed to be zero; and S_h is the horizontal shift, which is also assumed to be zero.

2.2.2. Rational Tire Model

Considering that the subsequent SOSP requires the dynamic equations of the controlled system, these should possess a polynomial or rational function form in order to construct a Lyapunov function and solve the stability boundary. If the Magic Formula is directly adopted, the SOS optimization problem cannot be formulated, and the introduction of longitudinal forces will lead to an exponential explosion in the computational burden of SOSP. Therefore, to balance computational feasibility and model accuracy, a rational tire model based on algebraic fractions is introduced to approximate the nonlinear characteristics of the tire, and its mathematical form is expressed as:

$$F_y = \frac{\mu F_z \left[p_1 \left(\frac{\alpha}{\mu} \right)^3 + p_2 \left(\frac{\alpha}{\mu} \right) \right]}{p_3 \left(\frac{\alpha}{\mu} \right)^4 + p_4 \left(\frac{\alpha}{\mu} \right)^2 + p_5} \quad (5)$$

where $p_1, \dots, p_5$ are all parameters to be fitted. A 215/55 R17 tire model in CarSim 2024.0 is used for parameter fitting, and the fitted parameters are listed in Table 1.

Table 1. Fitting parameters of the rational tire model for lateral force.

Parameter	p_1	p_2	p_3	p_4	p_5
Value	0.1306	31.6853	0.0029	3.3757	119.9965

2.3. Rationalization of the Vehicle Model

The SOSP algorithm requires that the nonlinear terms of the system can be accurately expressed in polynomial or rational forms. Due to the numerous nonlinear links in the seven-degree-of-freedom model, the direct processing form is excessively complex; therefore, reasonable assumptions are introduced to establish a two-degree-of-freedom reference model:

$$\begin{cases} m(\dot{v}_y + v_x \gamma) = 2F_{yf} \cos \delta + 2F_{yr} \\ I_z \dot{\gamma} = 2L_f F_{yf} \cos \delta - 2L_r F_{yr} \end{cases} \quad (6)$$

where F_{yf} and F_{yr} denote the lateral tire forces of the front and rear axles, respectively.

At this point, the sideslip angles of the front and rear tires are simplified as:

$$\begin{cases} \alpha_f = \frac{v_y + L_f \gamma}{v_x} - \delta = \beta + \frac{L_f \gamma}{v_x} - \delta \\ \alpha_r = \frac{v_y - L_r \gamma}{v_x} = \beta - \frac{L_r \gamma}{v_x} \end{cases} \quad (7)$$

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

The rational tire model is correspondingly simplified into the following form:

$$F_y = \frac{n_i(\alpha_i)}{d_i(\alpha_i)}, \quad i = f, r \quad (8)$$

where $n_i(\alpha_i)$ and $d_i(\alpha_i)$ are polynomial functions with regard to the sideslip angle α_i ; and f and r denote the front and rear wheels, respectively.

By combining the vehicle body dynamics model with the rational tire model through the tire sideslip angle, the rational dynamics model of the entire vehicle is obtained. Its state-space equation can be uniformly expressed as:

$$\begin{cases} \dot{x} = f(x) = \frac{N(x)}{D(x)} \\ \text{s.t.} \begin{cases} N = \begin{bmatrix} (2n_f d_r \cos \delta + 2n_r d_f - m v_x d_f d_r \omega) I_z \\ (2n_f d_r L_f - 2n_r d_f L_r) m v_x \end{bmatrix} \\ D = d_f d_r I_z m v_x \end{cases} \end{cases} \quad (9)$$

where $x = [\beta, \gamma]^T$ is the state variable; $N(x)$ and $D(x)$ are polynomial functions comprising n_i and d_i , respectively, which represent the numerator and denominator parts of the system dynamic equations.

3. Vehicle Stability-Region Prediction Based on a Hybrid ISOSP-LSTM Architecture

The ISOSP-LSTM module is designed as the stability-awareness component of the proposed coordinated control framework. It provides a coordination layer with real-time dynamic stability-boundary information through a two-stage identification strategy. When offline, ISOSP estimates stability-region boundary parameters under representative operating conditions and generates physically interpretable training labels. When online, the SSA-optimized LSTM network predicts the current stability-boundary coefficients from sequential vehicle-state information. In this way, the coordination layer can obtain the required stability boundary through fast neural-network inference while retaining the stability-theoretic basis of the predicted result.

3.1. ISOSP-Based Offline Estimation of Vehicle Stability Region

3.1.1. Optimization Problem Formulation for Vehicle Stability-Boundary Estimation

For the entire vehicle system after the rationalization process in Section 2.3, the equilibrium point of the system is shifted to the origin of the state space. According to the Lyapunov stability theory, the system satisfies the asymptotic stability conditions at the equilibrium point, and the specific expressions are formulated as:

$$\begin{cases} V(0) = 0 \\ V(x) > 0 \text{ in } B \setminus \{0\} \\ \nabla V \cdot f(x) < 0 \text{ in } B \setminus \{0\} \end{cases} \quad (10)$$

where B is the neighborhood of the equilibrium point; and $V(x)$ is the Lyapunov function to be constructed.

The region of asymptotic stability is the neighborhood of the equilibrium point, and any system trajectories originating within this neighborhood will eventually converge to the equilibrium point. Therefore, the stability region of the system is formally defined as:

$$\Omega_c = \{x \in \mathbb{R}^2 \mid V(x) \leq c\} \quad (11)$$

where c is the maximum-level value of $V(x)$.

SOSP simultaneously searches for the Lyapunov function and its maximum-level set; thus, the traditional process of the Lyapunov function for construction and stability-region characterization, which relies on empirical trial and error, is transformed into the following optimization problem:

$$\begin{aligned} & \max\{c\} \\ \text{s.t. } & \begin{cases} V(x) > 0 \text{ and } V(0) = 0 \\ \{x \in \mathbb{R}^2 \mid V(x) < c\} \subseteq \{x \in \mathbb{R}^2 \mid \dot{V}(x) < 0\} \cup \{0\} \end{cases} \end{aligned} \quad (12)$$

To reduce the conservatism of the stability-region estimation and capture the asymmetric boundary characteristics of the vehicle nonlinear system, a scalable reference set comprising a shape function $s(x)$ and a shape factor λ is introduced under the premise of satisfying the Lyapunov theorem:

$$\{x \in \mathbb{R}^2 \mid s(x) < \lambda\} \subseteq \{x \in \mathbb{R}^2 \mid V(x) < c\} \quad (13)$$

Under the premise of strictly satisfying the Lyapunov theorem, the reference region is driven to approximate the true stability region by maximizing λ . However, the set containment constraint requires point-by-point verification for infinitely many points, which is computationally infeasible. The generalized S-procedure theory [29] can transform the infinite-dimensional geometric containment problem into a solvable sum-of-squares constrained optimization problem:

$$\begin{aligned} & \max\{\lambda\} \\ \text{s.t. } & \begin{cases} q_1(x), q_2(x) \in \Sigma \\ V(x) - l_1(x) \in \Sigma \\ q_1(x)(s(x) - \lambda) - (V(x) - c) \in \Sigma \\ -\nabla V N - D l_2(x) + D q_2(x)(V(x) - c) \in \Sigma \end{cases} \end{aligned} \quad (14)$$

where l_1 , l_2 , q_1 , and q_2 are the introduced polynomial-type slack variables; and Σ denotes the set of sum-of-squares polynomials.

3.1.2. Vehicle Stability-Region Estimation Strategy Based on ISOSP

The traditional SOSP algorithm adopts the V-s iteration strategy, in which the variation in the shape factor below a predefined threshold is used as the termination criterion. However, since both the shape function and the shape factor vary during the iteration, the maximum shape factor does not necessarily correspond to the maximum stability region, which may increase the conservatism of the estimated boundary. To address this issue, the proposed ISOSP replaces the conventional shape-factor-based termination criterion with an area-based convergence criterion for the stability region. In addition, a feasibility-feedback-based interval updating strategy and a boundary-verification-based early termination strategy are introduced. These optimizations can reduce the conservatism of stability-boundary estimation and avoid repeated invalid searches over infeasible intervals during the stability-region estimation process. The specific flowchart is illustrated in Figure 3.

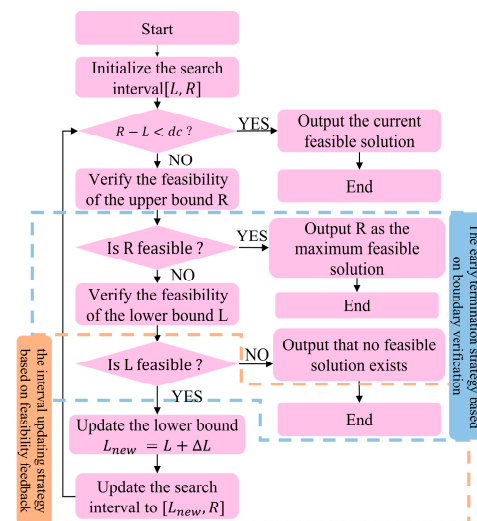


Figure 3. Flowchart of the ISOSP search and iteration strategy.

Specifically, the interval updating strategy based on feasibility feedback is implemented during the iterative search using the bisection method. First, the feasibility verification is performed on the lower bound L of the current search interval. If L is feasible, the lower bound is updated to $L + \Delta L$ while the upper bound remains unchanged, and the iterative search continues. Conversely, if L is infeasible, it indicates that no feasible solution exists within the current interval; thus, the algorithm terminates immediately and returns the conclusion of no feasible solution. By gradually advancing the lower bound, this strategy continuously approaches the maximum feasible solution during the iterative process. Meanwhile, it avoids invalid searches in the infeasible region, thereby effectively improving the solving efficiency of the algorithm.

The early termination strategy based on boundary verification is implemented after the search interval is determined. When the algorithm enters the iterative solving phase, an existence judgment of feasible solutions is performed on the upper bound of the interval. If a feasible solution exists, it indicates that the entire interval is feasible; thus, the upper bound can be directly output as the maximum feasible solution, and the iteration of this step terminates. Conversely, if the upper bound is infeasible, the feasibility of the lower bound of the interval is further verified. If the lower bound is feasible, the search process is executed within the interval to locate the maximum feasible solution. If the lower bound is also infeasible, it indicates that no feasible solution exists within the entire interval; consequently, the algorithm immediately stops iterating and returns the conclusion of no feasible solution. Through boundary feasibility determination, this flow achieves a pre-judgment of the necessity of searching, effectively avoiding redundant computation while ensuring the completeness of the results.

The saddle point is a key characteristic point on the boundary of the stability region, and substituting its coordinates into the iteratively constructed Lyapunov function can dynamically determine the search interval. ISOSP executes the early termination strategy based on boundary verification prior to searching, and implements the interval updating strategy based on feasibility feedback during searching. By integrating saddle point information with feasibility determination, this method not only guarantees theoretical rigor but also significantly improves computational efficiency.

Under the straight-line condition, the vehicle is assumed to travel at a medium speed on a road with a high-adhesion coefficient. To evaluate the reduced conservatism and improved computational efficiency of the proposed ISOSP method, the stability-region area and computation time are selected as quantitative metrics for comparison with the

conventional SOSP method. The simulations are performed on a 2.11 GHz desktop PC with 8 GB of RAM. The comparison results under the same operating condition are listed in Table 2.

Table 2. Comparison of computation time and ROA area between SOSP and ISOSP.

Degv	Time Consumption (s)			Area (rad ² /s)		
	SOSP	ISOSP	Improvement	SOSP	ISOSP	Improvement
2	97.7605	47.4734	+51.44%	1.0273	1.0478	+2.00%
4	218.2896	92.9547	+57.42%	1.3290	1.4094	+6.05%
6	497.4205	124.4300	+74.98%	2.0565	2.0561	−0.19%
8	863.8834	195.7928	+77.34%	2.8415	2.7421	−3.50%

As shown in Table 2, the proposed ISOSP significantly reduces the computation time for ROA estimation at different Lyapunov function orders. Compared with conventional SOSP, the computation time is reduced by 51.44%, 57.42%, 74.98%, and 77.34% for the second-, fourth-, sixth-, and eighth-order ROA estimations, respectively. This indicates that the feasibility-feedback-based interval updating strategy and the boundary-verification-based early termination strategy can effectively avoid repeated invalid searches and reduce the computational burden, especially for high-order Lyapunov functions. In terms of ROA area, the ISOSP method obtains results close to those of the conventional SOSP method. For the second- and fourth-order cases, the estimated ROA areas increase by 2.00% and 6.05%, respectively, whereas slight decreases of 0.19% and 3.50% are observed for the sixth- and eighth-order cases. As the order of the Lyapunov function increases, the number of monomial terms and the associated computational burden increase substantially. During the ISOSP iteration, the search interval is progressively narrowed to reduce unnecessary iterations and computational cost. For high-order cases, this narrowed search interval may prevent the algorithm from continuing to search for a larger feasible Lyapunov function value, thereby leading to a locally suboptimal solution and a slightly smaller estimated ROA. Nevertheless, once the estimated ROA has approached the true stability boundary, such as the saddle point boundary, further expansion becomes limited, and the resulting deviation in stability-region characterization remains negligible. Therefore, the slight reductions in the sixth- and eighth-order cases represent an acceptable trade-off for the substantial reduction in computation time.

3.2. Online Identification of Vehicle Stability Region Based on ISOSP-LSTM

3.2.1. Development of the Hybrid SSA-LSTM Prediction Model

The performance of the LSTM neural network [30] is highly sensitive to the values of its hyperparameters. The manual parameter tuning approach makes it difficult to obtain the global optimal configuration within a continuous and multidimensional hyperparameter domain. The SSA possesses distinct advantages, such as rapid convergence, strong robustness, and resistance to trapping in local optima. Therefore, this section adopts the SSA as the optimizer. By simulating the foraging and anti-predation behaviors of sparrow populations, the combination of hyperparameters that minimizes the LSTM prediction error is iteratively searched, thereby constructing the coupled SSA-LSTM prediction model.

The LSTM architecture used in this study consists of an input sequence layer, one LSTM layer for temporal feature extraction, a fully connected layer and a ReLU activation layer for boundary coefficient regression. A fixed time-window length of 10 is adopted so that the input sequence remains physically continuous. In each LSTM unit, the cell state stores historical information, while the hidden state is transferred to the subsequent network layer.

The sparrow search algorithm (SSA) is a swarm intelligence optimization algorithm inspired by the foraging behavior of sparrow populations [31]. According to the behavioral differences in individuals during the foraging process, sparrows can be divided into three types of roles: discoverers, followers, and sentinels.

Discoverers are responsible for searching for areas with abundant food. Each individual in the population can exhibit this behavior, and the number of discoverers accounts for a fixed proportion of the population as approximately 10–20%. The position update formula for discoverers is expressed as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(-\frac{i}{\alpha \cdot D_{\max}}\right), & \text{if } R_2 < ST \\ X_{i,j}^t + Q \cdot L, & \text{if } R_2 \geq ST \end{cases} \quad (15)$$

where $X_{i,j}^t$ denotes the position of the i -th sparrow on the j -th hyperparameter in the t -th iteration, $D_{\max}$ is the maximum number of iterations, α is a random number within the interval $[0, 1]$, the warning value R_2 and the safety threshold ST both take values within $[0, 1]$, Q is a random number following the standard normal distribution, and L is a unit element matrix. When $R_2 < ST$, it indicates that the current environment is safe, and the discoverers can perform large-scale exploration. When $R_2 > ST$, it indicates that some sparrows have detected danger and triggered an alarm, and all individuals need to immediately fly to other safe locations for foraging.

The behavioral characteristic of followers is to follow the discoverers to obtain higher-quality food sources and move their positions accordingly. The position update rule for followers is expressed as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{\text{worst}}^t - X_{i,j}^t}{i^2}\right), & \text{if } i > n/2 \\ X_P^{t+1} + |X_{i,j}^t - X_P^{t+1}| \cdot A^+ \cdot L, & \text{otherwise} \end{cases} \quad (16)$$

where X_P is the current position of the best discoverer; X_{worst} is the global worst position; and $A^+ = A^T(AA^T)^{-1}$, in which A is a matrix whose elements are 1 or -1 . When $i > n/2$, it indicates that the i -th follower fails to forage and is in a hungry state, so it needs to move to another area to search for food. When $i \leq n/2$, the follower will directly move near the current optimal position X_P for foraging.

When sentinels perceive danger signals, they terminate the current foraging process and move to other areas. The number of sentinels is maintained at a certain proportion of the population as approximately 10–20%. The position update rule for sentinels is expressed as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + v \cdot |X_{i,j}^t - X_{\text{best}}^t|, & \text{if } f_i > f_g \\ X_{i,j}^t + K \cdot \left(\frac{|X_{i,j}^t - X_{\text{worst}}^t|}{(f_i - f_w) + \varepsilon}\right), & \text{if } f_i = f_g \end{cases} \quad (17)$$

where X_{best} is the global best position; v and K are random numbers controlling the step size; and f_i , f_g , and f_w denote the individual fitness, global best fitness, and worst fitness, respectively. When $f_i > f_g$, sparrows are located at the edge position and will move closer to X_{best} . When $f_i = f_g$, sparrows located at the center of the population move in other directions to avoid danger, thereby guiding the population to escape from local optima.

When constructing the vehicle stability-region prediction model, three hyperparameters exert the most significant impact on the performance of the LSTM network, including the number of hidden layer neurons N_{hidden} , the initial learning rate Y_0 , and the L2 regularization coefficient θ . Therefore, these three parameters are selected as the optimization

variables for the SSA. Specifically, N_{hidden} determines the network capacity and nonlinear fitting capability. Too few neurons may lead to underfitting, making it difficult to capture the complex characteristics of the stability-boundary features, whereas too many neurons may cause overfitting and increase the computational burden. Therefore, the search range is set to $[10, 200]$. Y_0 determines the step size of the gradient descent. An excessively large learning rate may cause oscillation of the loss function and prevent convergence, whereas an excessively small learning rate may result in slow convergence. Therefore, the search range is set to $[0.0001, 0.01]$. θ is used to control the model complexity, prevent the weight matrix from becoming excessively large, and enhance the generalization capability of the model. Its search range is set to $[10^{-6}, 10^{-2}]$. In the process of SSA optimizing LSTM, the position vector of the i -th sparrow is defined as $X_i = [N_{\text{hidden}}, Y_0, \theta]_i$. To evaluate the quality of the hyperparameter combinations, the dataset is partitioned into a training set S_{train} and a validation set S_{test} . In each iteration of the SSA, the LSTM network is constructed using the hyperparameters represented by the current sparrow position, trained for short cycles on S_{train} , and subsequently evaluated for prediction error on S_{test} . The root mean square error (RMSE) of the validation set is selected as the fitness function.

The execution flow of the designed coupled SSA-LSTM model is illustrated in Figure 4. The framework first initializes the SSA parameters and sparrow population, where each sparrow position represents a candidate LSTM hyperparameter vector $X = [N_{\text{hidden}}, Y_0, \theta]$. For each candidate vector, an LSTM network is constructed and briefly trained on the training set, and the RMSE on the validation set is used as the fitness value. The producer, scrounger, and sentinel positions are then updated iteratively, followed by boundary checking and global optimum updating. When the maximum number of iterations is reached, the optimal hyperparameters X_{best} are transferred to the final LSTM training stage. The final LSTM model is trained using X_{best} and then applied to predict the vehicle stability-boundary coefficients from new driving-condition sequences.

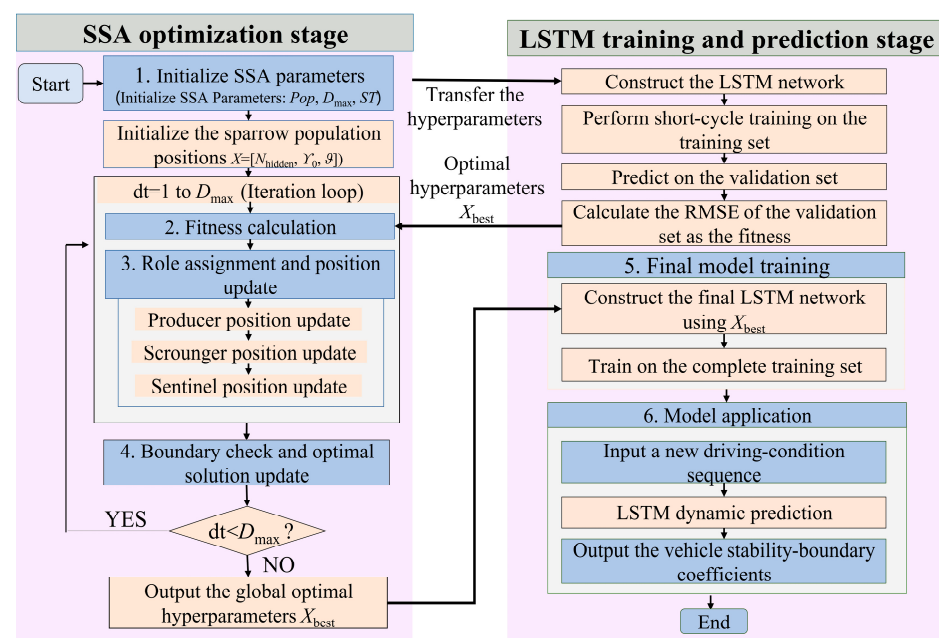


Figure 4. Execution flowchart of the coupled SSA-LSTM model.

3.2.2. Construction of the ISOSP-LSTM-Based Stability-Region Prediction Model

Under a specific driving condition, the vehicle stability region estimated by the ISOSP method is uniquely determined by the Lyapunov function $V(x)$ and its corresponding function value c . For a given order of $V(x)$, the dimension of its polynomial coefficient vector

is fixed. Therefore, the stability region under this driving condition can be characterized by the following mapping relationship:

$$(V_x, \mu, \delta) \rightarrow y(a_0, a_1, \dots, a_n, c) \quad (18)$$

where V_x , μ , and δ denote the longitudinal velocity, road adhesion coefficient, and front-wheel steering angle, respectively; and a_0 , a_1 , $\dots$, a_n , and c are the predicted stability-boundary coefficients.

The sample dataset is constructed using the CarSim–MATLAB/Simulink co-simulation platform under double-lane-change maneuvers, and the corresponding driving-condition configuration is presented in Table 3. The longitudinal vehicle speed ranges from 18 km/h to 126 km/h with an interval of 18 km/h, while the road adhesion coefficient is set to 0.3, 0.5, and 0.85 to represent low-, medium-, and high-adhesion roads, respectively. Constant speed, acceleration, and deceleration processes are considered under each adhesion level. Considering the feasible operating range of the vehicle under different road conditions, the maximum speed is reduced under low-adhesion conditions, whereas a wider speed range is covered under high-adhesion conditions. A total of 44 driving-condition groups were generated.

Table 3. Configuration of the driving-condition dataset.

Driving Process	Road Adhesion Coefficient	Initial Speed Range (km/h)
Constant speed	0.3	18–72
Acceleration	0.3	18–54
Deceleration	0.3	18–72
Constant speed	0.5	18–90
Acceleration	0.5	18–72
Deceleration	0.5	18–90
Constant speed	0.85	18–126
Acceleration	0.85	18–90
Deceleration	0.85	18–126

Each group consisted of a continuous time series containing the longitudinal velocity, road adhesion coefficient, and front-wheel steering angle. At each time step, the corresponding stability-boundary coefficient vector is calculated using ISOSP, thereby forming paired input–output samples between the operating-condition variables (V_x , μ , δ) and the stability-boundary coefficients (a_0 , a_1 , $\dots$, a_n , c).

Since the original dataset consists of multiple independent simulation conditions, discontinuities in the connections between different conditions could introduce physically meaningless temporal features. Therefore, condition-switching points are identified before sequence construction. A new subsequence is created when the speed difference between two consecutive time steps exceeds 5 km/h or when the road adhesion coefficient changes. A sliding window with a fixed sequence length of 10 is then applied within each continuous subsequence, and no input sequence is allowed to cross a condition boundary. Approximately 90,000 sequence samples were generated after data preprocessing and sliding-window construction.

To achieve high-precision prediction of the stability-region boundaries, the pre-processed sample dataset is first partitioned into a training set, a validation set, and a test set according to a ratio of 7:2:1. Figure 5 illustrates the fitness convergence curve during the SSA optimization process, where the vertical axis denotes the RMSE of the validation set. The fitness value decreases rapidly during the first five generations, indicating that SSA can quickly search for a promising hyperparameter region. After convergence, the validation

RMSE reaches 0.0217, and the global optimal hyperparameter combination is obtained. The LSTM network is trained using the Adam optimizer algorithm with a maximum of 600 epochs. A piecewise learning-rate adjustment strategy is adopted during the training process, where the learning rate is reduced by a factor of 0.1 after 450 epoch-s. The initial learning rate is determined by the SSA optimization process within the predefined search range. The training process is implemented using MATLAB R2025b. Compared with conventional grid search, which requires a large number of discrete evaluations, SSA reduces the manual tuning burden and improves the efficiency of LSTM configuration selection.

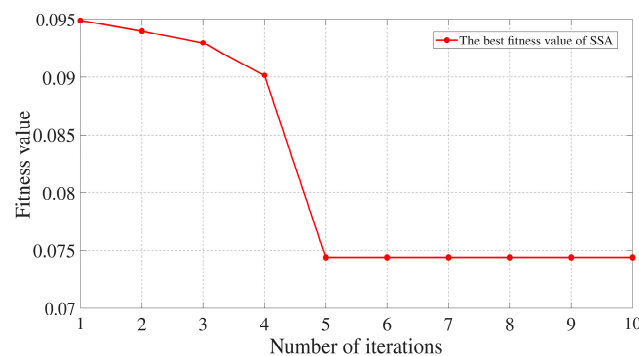


Figure 5. Fitness convergence curve of SSA during hyperparameter optimization.

Figures 6 and 7 illustrate the evolution curves of the loss values and RMSEs versus the number of epochs for the SSA-LSTM model on the training and validation sets, respectively. It can be observed that within the first 100 steps of the early training stage, both the loss value and RMSE of the SSA-LSTM model decrease rapidly. This indicates that the hyperparameters optimized by the SSA are reasonable, and the network can quickly capture the gradient direction, yielding high training efficiency. When iterating to 2500 times, the curves of the training and validation sets fit closely with no obvious overfitting observed, which demonstrates that the model possesses good generalization capability. The jagged oscillation of the training loss is a normal phenomenon characteristic of mini-batch stochastic gradient descent.

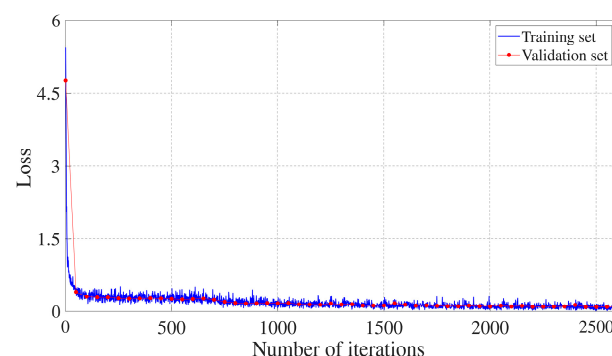


Figure 6. Loss evolution curves of the SSA-LSTM model during training.

Figure 8 displays the RMSEs of the model for each output stability-boundary coefficient in the test set. The error values for the coefficients numbered 7, 8, 11, and 14 are relatively higher. The underlying reason is that these high-order coefficients determine the sharpness and local curvature of the stability-region boundaries, making them highly sensitive to the nonlinear variations under extreme driving conditions and thus more difficult to fit. Nevertheless, the maximum error remains constrained within 0.05, which satisfies the accuracy requirements for the stability-region estimation.

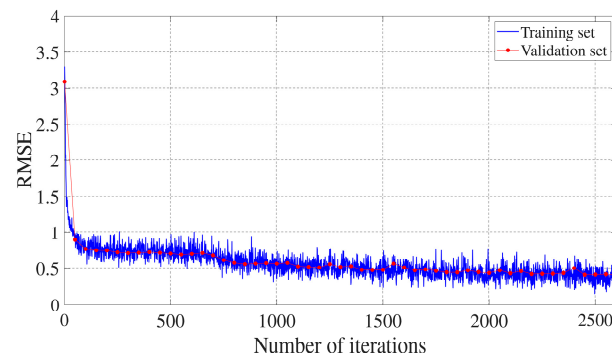


Figure 7. RMSE evolution curves of the SSA-LSTM model during training.

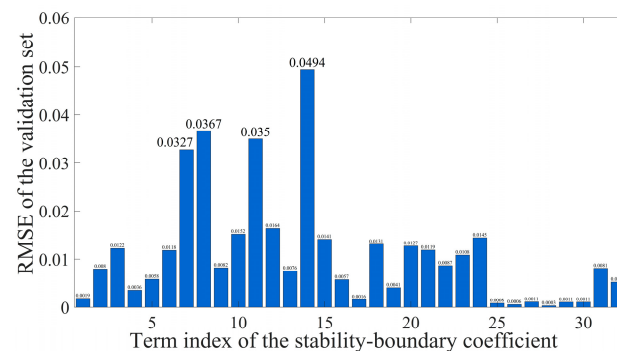


Figure 8. Predicted RMSEs of individual stability-boundary coefficients on the test set.

4. Strategy Design for Integrated Longitudinal and Lateral Trajectory Tracking and Coordinated Stability Control

To balance trajectory tracking accuracy and driving stability, a three-layer hierarchical control architecture is proposed, as shown in Figure 9. The upper layer integrates longitudinal–lateral trajectory tracking, yaw stability control, and auxiliary steering to generate desired vehicle motion states. The middle layer introduces a risk factor based on the stability margin derived from the stability boundary, reconstructing the optimization objectives and dynamically adjusting actuator authority. The lower layer minimizes tire utilization via quadratic programming to achieve optimal four-wheel torque distribution.

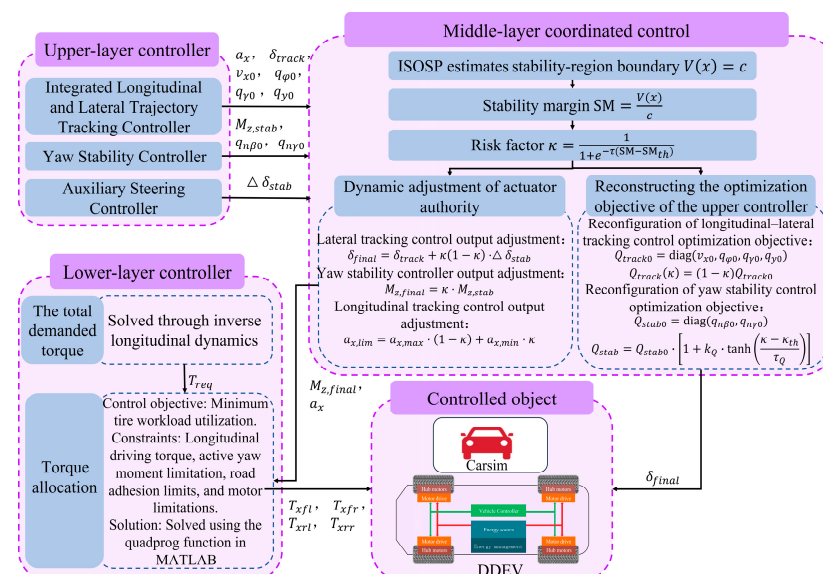


Figure 9. Hierarchical architecture diagram of the coordinated control system.

4.1. Design of the Upper-Level Controller

4.1.1. Integrated Longitudinal–Lateral Trajectory Tracking Controller Design

In the MPC-based integrated longitudinal and lateral tracking control, the algorithm takes the target trajectory sequence as input and estimates future states using the predictive model. By constructing an objective function that incorporates tracking errors, control variables, and constraints on state variables, the optimal control sequence is solved in real time. Finally, the expected values of the front-wheel steering angle and longitudinal acceleration/deceleration are generated as outputs.

The system equations can be expressed in a general nonlinear form as:

$$\dot{x} = f(x, u) \quad (19)$$

where $x = [v_y, v_x, \varphi, \gamma, X, Y]^T$ is state vector of the predictive model, and $u = [a_x, \delta]^T$ is the control vector.

Given the reference point (x_r, u_r) at the current time step, the continuous-time predictive model is linearized by performing a Taylor series expansion around this point as follows:

$$\begin{aligned} \dot{\tilde{x}} &= A\tilde{x} + B\tilde{u} \\ \text{s.t. } \tilde{x} &= x - x_r, \tilde{u} = u - u_r, A = \left. \frac{\partial f}{\partial x} \right|_{\substack{x=x_r \\ u=u_r}}, B = \left. \frac{\partial f}{\partial u} \right|_{\substack{x=x_r \\ u=u_r}} \end{aligned} \quad (20)$$

Discretization yields the discrete-time state-space equation:

$$\tilde{x}(k+1) = A_d \tilde{x}(k) + B_d \tilde{u}(k) \quad (21)$$

where $A_d = I + TA$ and $B_d = TB$, in which I denotes the identity matrix and T represents the sampling period.

To establish a unified description of the system states and the control input from the previous time step, a new augmented state vector $\xi(k)$ is defined as:

$$\xi(k) = [\tilde{x}(k) \ \tilde{u}(k-1)]^T \quad (22)$$

By introducing the augmented system output $\eta = [v_x, \varphi, \gamma, Y]^T$, the augmented linear discrete-time state-space model can be formulated as:

$$\begin{cases} \xi(k+1) = \tilde{A}_d \xi(k) + \tilde{B}_d \Delta \tilde{u}(k) \\ \eta(k+1) = C \xi(k+1) \end{cases} \quad (23)$$

$$\text{s.t. } \tilde{A}_d = \begin{bmatrix} A_d & B_d \\ 0_{2 \times 6} & I_{2 \times 2} \end{bmatrix}, \tilde{B}_d = \begin{bmatrix} B_d \\ I_{2 \times 2} \end{bmatrix}$$

By assuming the prediction horizon is N_p and the control horizon is N_c , this satisfies $N_c < N_p$. Beyond the control horizon N_c , the increments of the control inputs become zero, meaning that the controller outputs remain constant. Consequently, the length of the optimization variable sequence ΔU to be solved is determined solely by N_c :

$$\Delta U = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix}_{N_c \times 1} \quad (24)$$

Meanwhile, the predictive output sequence Y of the system still maintains a length of N_p :

$$Y = \begin{bmatrix} \eta(k+1) \\ \eta(k+2) \\ \vdots \\ \eta(k+N_p) \end{bmatrix}_{N_p \times 1} \quad (25)$$

Further iterative derivation and matrix assembly based on the augmented state-space model yield the final predictive output equation:

$$Y = \Psi \xi(k) + \Theta \Delta U$$

$$\text{s.t. } \Psi = \begin{bmatrix} C\tilde{A}_d \\ C\tilde{A}_d^2 \\ \vdots \\ C\tilde{A}_d^{N_c} \\ \vdots \\ C\tilde{A}_d^{N_p} \end{bmatrix}, \Theta = \begin{bmatrix} C\tilde{B}_d & 0 & \dots & 0 \\ C\tilde{A}_d\tilde{B}_d & C\tilde{B}_d & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C\tilde{A}_d^{N_c-1}\tilde{B}_d & C\tilde{A}_d^{N_c-2}\tilde{B}_d & \dots & C\tilde{B}_d \\ C\tilde{A}_d^{N_c}\tilde{B}_d & C\tilde{A}_d^{N_c-1}\tilde{B}_d & \dots & C\tilde{A}_d\tilde{B}_d \\ \vdots & \vdots & \ddots & \vdots \\ C\tilde{A}_d^{N_p-1}\tilde{B}_d & C\tilde{A}_d^{N_p-2}\tilde{B}_d & \dots & C\tilde{A}_d^{N_p-N_c}\tilde{B}_d \end{bmatrix} \quad (26)$$

The quadratic objective function is defined as follows:

$$J(\xi(k), \Delta U(k)) = (Y - Y_{ref})^T Q_{aug} (Y - Y_{ref}) + \Delta U^T R_{aug} \Delta U + \rho \varepsilon^2$$

$$\text{s.t. } Q_{aug} = \text{diag}(Q, Q, \dots, Q); R_{aug} = \text{diag}(R, R, \dots, R) \quad (27)$$

Within each control interval, the optimal control sequence is obtained by solving the quadratic programming solver, and the first control increment is applied to the system.

4.1.2. Yaw Stability Controller Design

For computational convenience, a linearized MPC model with constant cornering stiffness coefficients k_{cf} and k_{cr} is adopted in the yaw stability controller. Since the vehicle states are constrained by the identified stability region, the nonlinear tire characteristics captured by the rational tire model are indirectly considered in the control design [17].

Under extreme conditions with saturated tire forces, solely pursuing trajectory tracking accuracy may cause vehicle instability. Therefore, this section introduces the additional yaw moment as an independent control input into the two-degree-of-freedom model to develop a yaw stability controller that maintains vehicle stability.

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{2k_{cf}+2k_{cr}}{mv_x} & \frac{2k_{cf}L_f-2k_{cr}L_r}{mv_x^2} - 1 \\ \frac{2k_{cf}L_f-2k_{cr}L_r}{I_z} & \frac{2k_{cf}L_f^2+2k_{cr}L_r^2}{I_z v_x} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I_z} \end{bmatrix} M_z + \begin{bmatrix} \frac{-2k_{cf}}{mv_x} \\ \frac{-2k_{cf}L_f}{I_z} \end{bmatrix} \delta \quad (28)$$

We let the system state vector be $e = [\beta, \gamma]^T$ and the control input be $w = [M_z]$. The front-wheel steering angle, which is obtained from the trajectory tracking controller in the previous section, is treated as the system disturbance $o = [\delta]$. Consequently, the aforementioned model can be reconstructed as follows:

$$\dot{e} = A_n e + B_n w + G_n o$$

$$\text{s.t. } A_n = \begin{bmatrix} \frac{2k_{cf}+2k_{cr}}{mv_x} & \frac{2k_{cf}L_f-2k_{cr}L_r}{mv_x^2} - 1 \\ \frac{2k_{cf}L_f-2k_{cr}L_r}{I_z} & \frac{2k_{cf}L_f^2+2k_{cr}L_r^2}{I_z v_x} \end{bmatrix}, B_n = \begin{bmatrix} 0 \\ \frac{1}{I_z} \end{bmatrix}, G_n = \begin{bmatrix} \frac{-2k_{cf}}{mv_x} \\ \frac{-2k_{cf}L_f}{I_z} \end{bmatrix} \quad (29)$$

After discretization, $A_{nd} = I + TA_n$ and $B_{nd} = TB_n$, where I denotes the identity matrix and T represents the sampling period.

With $\omega = [\beta \ \gamma \ MZ]$ defined as the state vector and $\chi = [e]$ defined as the output state vector of the system, the augmented linear discrete-time state-space model is formulated as follows:

$$\begin{aligned}\omega(k+1) &= \tilde{A}_{nd}\omega(k) + \tilde{B}_{nd}\Delta\tilde{w}(k) + \tilde{G}_{nd}o(k) \\ \chi(k+1) &= C_n\omega(k+1) \\ \text{s.t. } \tilde{A}_{nd} &= \begin{bmatrix} A_{nd} & B_{nd} \\ 0_{1 \times 2} & 1 \end{bmatrix}, \tilde{B}_{nd} = \begin{bmatrix} B_{nd} \\ 1 \end{bmatrix}, \tilde{G}_{nd} = \begin{bmatrix} G_d & 0 \end{bmatrix}, C_n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\end{aligned}\quad (30)$$

Consequently, the prediction equation is further formulated as:

$$\begin{aligned}T &= M\omega(k) + N\Delta W + \Gamma O \\ \text{s.t. } \left\{ \begin{aligned} T &= \begin{bmatrix} \chi(k+1) \\ \chi(k+2) \\ \vdots \\ \chi(k+N_p) \end{bmatrix}, M = \begin{bmatrix} C_n\tilde{A}_{nd} \\ C_n\tilde{A}_{nd}^2 \\ \vdots \\ C_n\tilde{A}_{nd}^{N_c} \\ \vdots \\ C_n\tilde{A}_{nd}^{N_p} \end{bmatrix}, \Delta W = \begin{bmatrix} \Delta\mathbf{w}(k) \\ \Delta\mathbf{w}(k+1) \\ \vdots \\ \Delta\mathbf{w}(k+N_p-1) \end{bmatrix}, O = \begin{bmatrix} \mathbf{o}(k) \\ \mathbf{o}(k+1) \\ \vdots \\ \mathbf{o}(k+N_p-1) \end{bmatrix}, \\ N &= \begin{bmatrix} C_n\tilde{B}_{nd} & 0 & \dots & 0 \\ C_n\tilde{A}_{nd}\tilde{B}_{nd} & C_n\tilde{B}_{nd} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_n\tilde{A}_{nd}^{N_c-1}\tilde{B}_{nd} & C_n\tilde{A}_{nd}^{N_c-2}\tilde{B}_{nd} & \dots & C_n\tilde{B}_{nd} \\ C_n\tilde{A}_{nd}^{N_c}\tilde{B}_{nd} & C_n\tilde{A}_{nd}^{N_c-1}\tilde{B}_{nd} & \dots & C_n\tilde{A}_{nd}\tilde{B}_{nd} \\ \vdots & \vdots & \ddots & \vdots \\ C_n\tilde{A}_{nd}^{N_p-1}\tilde{B}_{nd} & C_n\tilde{A}_{nd}^{N_p-2}\tilde{B}_{nd} & \dots & C_n\tilde{A}_{nd}^{N_p-N_c}\tilde{B}_{nd} \end{bmatrix} \end{aligned} \right. \quad (31)\end{aligned}$$

To satisfy the vehicle stability control requirements under extreme driving conditions, the vehicle states must closely track their reference values. Accordingly, the objective function is designed as follows:

$$\begin{aligned}J &= (T - T_{ref})^T \tilde{Q}_{stab}(T - T_{ref}) + \Delta W^T \tilde{R}_{stab} \Delta W \\ \text{s.t. } \tilde{Q}_n &= \begin{bmatrix} Q_{stab} & & \\ & \ddots & \\ & & Q_{stab} \end{bmatrix}, \tilde{R}_n = \begin{bmatrix} R_{stab} & & \\ & \ddots & \\ & & R_{stab} \end{bmatrix}\end{aligned}\quad (32)$$

where Q_{stab} and R_{stab} denote the penalty weight matrices for the state tracking error and the control increment, respectively.

To incorporate the offline stability region obtained via ISOSP into the constraints, the gradient of the stability boundary at the current operating point is first calculated as:

$$\nabla V(e(k+i)) = \left[\frac{\partial V}{\partial \beta}, \frac{\partial V}{\partial \gamma} \right]^T \bigg|_{e=e(k+i|k)} \quad (33)$$

Consequently, the linearized constraints are constructed at $e_s = (\beta_s, \gamma_s)$:

$$\nabla V(e(k+i))^T \cdot (e(k+i) - e_s) + V(e_s) \leq c \quad (34)$$

Ultimately, the optimization problem is solved via a quadratic programming solver, and the first control increment from the obtained optimal control sequence is applied to the controlled system.

4.1.3. Auxiliary Steering Controller Design

To fully utilize the lateral tire potential and assist the yaw stability controller in correcting the vehicle posture, an auxiliary active steering controller is designed, which

takes the yaw-rate error and sideslip-angle error as inputs. A proportional–derivative control law is employed to calculate the theoretical corrective steering angle:

$$\Delta\delta_{stab0} = -k_{\gamma}(\gamma - \gamma_{ref}) - k_{\beta}(\beta - \beta_{ref}) \quad (35)$$

where $\Delta\delta_{stab0}$ represents the theoretical front-wheel steering angle correction required to maintain stability, and k_{γ} and k_{β} denote the tunable feedback gain coefficients.

Considering the physical limitations of the steering actuator motor, a saturation constraint is applied to the calculated theoretical corrective steering angle:

$$\Delta\delta_{stab} = \text{sat}(\Delta\delta_{stab0}, -\delta_{\max}, \delta_{\max}) \quad (36)$$

where $\text{sat}(\cdot)$ denotes the saturation function, and $\delta_{\max}$ is the maximum allowable additional steering correction magnitude for the AFS system.

Since this command may conflict with the driver's intent or trajectory tracking requirements, it is not directly used as the input to the coordinated control strategy, and the coordination layer determines its specific acceptance weight.

4.2. Design of Middle-Level Coordination Control Strategy

4.2.1. Definitions of Stability Margin and Risk Factor

The value of the Lyapunov function directly reflects the degree to which the vehicle system states deviate from the equilibrium point. Specifically, as the function value approaches zero, the vehicle system transitions into a stable state. Conversely, when the function value approaches c , the vehicle system nears the stability boundary, leading to a sharp increase in instability risk. To facilitate the normalized scheduling of controller weights, the stability margin SM is defined based on the vehicle stability region as follows:

$$\text{SM} = \frac{V(x)}{c} \quad (37)$$

Evidently, $\text{SM} \in [0, 1]$ indicates that the vehicle is operating within the stability region, whereas $\text{SM} > 1$ signifies that the vehicle has exceeded the stability boundary.

To achieve a smooth and continuous transition between the trajectory tracking controller and the stability controller, a nonlinear mapping from the stability margin to the instability risk factor κ is constructed. The mapping relationship is depicted in Figure 10, and the mapping function is formulated as:

$$\kappa = \frac{1}{1 + e^{-\tau(\text{SM} - \text{SM}_{th})}} \quad (38)$$

where τ denotes the sensitivity adjustment coefficient, and SM_{th} represents the stability threshold. This implies that when the Lyapunov function value reaches SM_{th} times the boundary value, a significant trend towards instability is acknowledged. In this study, SM_{th} is set to 0.8 because the vehicle is still far from the boundary when SM is much lower than 0.8, whereas the instability risk increases rapidly as SM approaches one. The coefficient τ is selected to balance response sensitivity and smoothness; an excessively small value causes delayed intervention, while an excessively large value makes the transition close to hard switching. In the present implementation, $\tau = 3$ is adopted, which enables κ to increase smoothly around SM_{th} and approach saturation near the stability boundary.

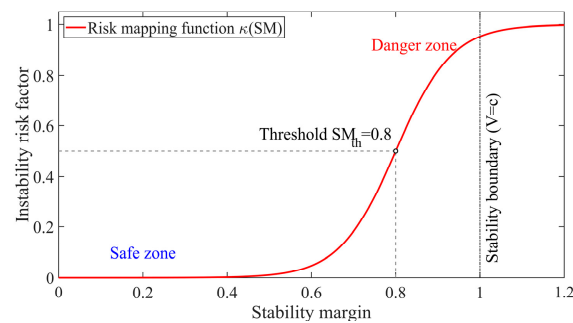


Figure 10. Mapping relationship between stability margin and risk factor.

4.2.2. Dynamic Reconfiguration of Optimization Objective Based on Risk Factor

To avoid excessive front-wheel steering angles and tire force saturation caused by aggressive tracking under extreme driving conditions, this section proposes a strategy for the dynamic reconstruction of optimization objectives based on the risk factor.

The weights in the objective function of the trajectory tracking controller are block-diagonal matrices comprising Q and R . To prevent the controller from issuing aggressive commands that induce oversteering in its attempt to strictly track the trajectory under extreme conditions, the tracking weight matrix Q is designed as a continuous function of the instability risk factor κ :

$$\begin{aligned} Q_{track} &= \text{diag}(q_{vx0}, q_{\varphi0}, q_{\gamma0}, q_{y0}) \\ Q(\kappa) &= (1 - \kappa)Q_{track} \end{aligned} \quad (39)$$

where q_{vx0} , $q_{\varphi0}$, $q_{\gamma0}$, and q_{y0} denote the default tracking weights configured for nominal driving conditions.

The primary objective of the yaw stability MPC controller is to ensure that the vehicle states remain within the stability boundary. Under low-risk conditions, it operates in a low-priority supervisory mode with minimal intervention. Conversely, under high-risk scenarios, it rapidly gains top priority to execute aggressive intervention. To this end, leveraging the smooth saturation characteristics of the hyperbolic tangent function, a nonlinear weight adjustment mechanism is designed as follows:

$$\begin{aligned} Q_{stab0} &= \text{diag}(q_{n\beta0}, q_{n\gamma0}) \\ Q_{stab}(\kappa) &= Q_{stab0} \cdot \left[1 + k_Q \cdot \tanh\left(\frac{\kappa - \kappa_{th}}{\tau_Q}\right) \right] \end{aligned} \quad (40)$$

where Q_{stab0} denotes the baseline weight matrix, k_Q represents the maximum amplification factor, and τ_Q signifies the transition slope coefficient.

4.2.3. Dynamic Actuator Authority Adjustment

To address the operating-condition dependency where the control effectiveness of AFS and DYC varies with the tire sideslip characteristics, an adaptive control allocation strategy based on the physical characteristics of the actuators is proposed. This strategy dynamically allocates the control authority and target objectives of both actuators based on the risk factor, ensuring that the system invokes the most efficient actuator combination to maintain stability across all operating conditions.

The final control weight factor for AFS is defined as a function of the risk factor:

$$\sigma_{AFS} = \kappa(1 - \kappa) \quad (41)$$

Ultimately, the actual steering angle command of the AFS actuator is obtained through a weighted fusion of the steering command δ_{track} from the trajectory tracking controller and the steering correction command $\Delta\delta_{stab}$ from the stability controller:

$$\delta_{final} = \delta_{track} + \sigma_{AFS} \cdot \Delta\delta_{stab} \quad (42)$$

The additional yaw-moment demand calculated by the stability controller is directly executed based on the risk factor κ , and its final command is formulated as:

$$M_{z,final} = \kappa \cdot M_{z,stab} \quad (43)$$

where $M_{z,final}$ is the final torque output of the actuator, and $M_{z,stab}$ represents the torque output by the stability controller.

Furthermore, the longitudinal acceleration constraint is established based on the risk factor:

$$a_{x,lim} = a_{x,max} \cdot (1 - \kappa) + a_{x,min} \cdot \kappa \quad (44)$$

where $a_{x,max}$ denotes the maximum acceleration expected by the upper-layer controller, and $a_{x,min}$ is the minimum allowable acceleration.

As κ increases from zero to one, the upper limit of the longitudinal acceleration decreases linearly from $a_{x,max}$ to $a_{x,min}$. Under extremely high-risk scenarios, the vehicle system enforces active deceleration or even emergency braking, thereby guaranteeing that the vehicle recovers its stability.

4.3. Lower-Layer Four-Wheel Torque Allocation

The total tire-workload function of the vehicle is defined as the optimization objective for torque allocation, which is formulated as follows:

$$\min J = \min \sum_{ij=fl,fr,rl,rr} \ell_{ij} \frac{T_{xij}^2}{(\mu R_{eff} F_{zij})^2} \quad (45)$$

where J represents the total tire workload of the vehicle, and ℓ_{ij} denotes the weighting coefficient for each wheel.

Subject to constraint conditions, including the longitudinal driving torque demand, the supplementary yaw-moment target, the tire adhesion limit, and the maximum output torque of the in-wheel motors, the lower-layer four-wheel torque allocation problem is transformed into a quadratic programming model for solution.

5. Simulation Analysis and Test Verification

To verify the effectiveness of the proposed coordinated control strategy under varying vehicle speeds and adhesion limits, HIL tests are conducted under variable-speed high-adhesion and high-speed low-adhesion conditions. The former introduces time-varying longitudinal acceleration at high initial speeds, demonstrating the controller's performance under strong longitudinal–lateral coupling. The latter tests the strategy's ability to ensure driving safety under extreme constraints. Comparative simulations are performed between integrated tracking control and coordinated control. Tracking accuracy is measured by maximum absolute error (MaxAE) and RMSE of lateral displacement and longitudinal speed, and stability by MaxAE/RMSE of yaw rate and sideslip angle.

5.1. Construction of the HIL Test Platform

The developed HIL test platform primarily consists of three components, including an NI PXIe-1078 real-time simulator, a host computer, and a domain controller, as illustrated

in Figure 11. The Simulink model of the control algorithm is converted into embedded C code via a code generation tool and subsequently flashed into the domain controller. Concurrently, a high-fidelity vehicle model is constructed in CarSim on the host computer, with the steering wheel angle and four-wheel drive torque configured as input interfaces and vehicle states as output interfaces. This vehicle model is then compiled, deployed to NI VeriStand, and downloaded into the PXIe-1078 real-time simulator for execution. The PXIe-1078 solves the vehicle dynamics equations at a fixed hard real-time step and transmits status signals to the domain controller via the CAN bus. Based on the received states, the domain controller calculates the target steering angle and the desired torque for each wheel in real time. These control commands are then fed back to the real-time simulator, thereby establishing a closed-loop vehicle dynamic control system. The overall system architecture is depicted in Figure 12.

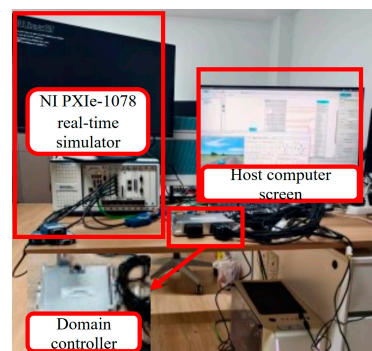


Figure 11. HIL platform for experiments.

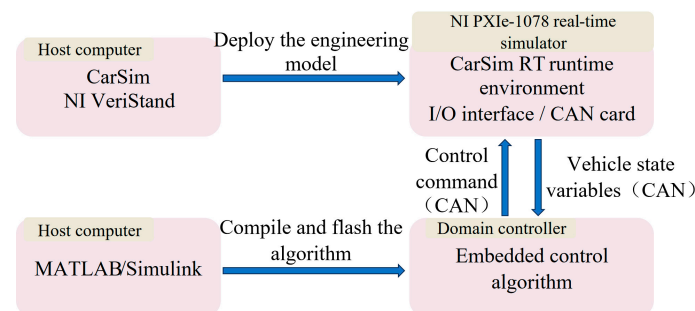


Figure 12. Overall architecture diagram of the HIL test.

5.2. Validation of Coordinated Control Strategy on the HIL Platform

5.2.1. Variable-Speed and High-Adhesion DLC Conditions

The parameters for the simulation test under the variable-speed, high-adhesion DLC condition are configured as follows: the vehicle uniformly accelerates from an initial speed of 60 km/h to 90 km/h within a travel distance of 25 m to 225 m, and the road adhesion coefficient is set to 0.85. The corresponding test results and quantitative evaluation metrics under this condition are presented in Figure 13 and Table 4, respectively.

In terms of trajectory tracking accuracy, as shown in Figure 13a,b, the pure tracking controller produces a noticeable lateral overshoot near the peak displacement during the DLC maneuver. This phenomenon can be attributed to the fact that the pure tracking controller mainly prioritizes path-error reduction, while its consideration of the available lateral tire force reserve is limited under strongly coupled longitudinal–lateral motion. In contrast, the proposed coordinated controller suppresses the lateral overshoot by integrating trajectory tracking, yaw-moment regulation, and tire-workload optimization. As listed in Table 4, the maximum lateral error is reduced from 0.8327 m to 0.4551 m, and the

RMSE of lateral displacement decreases from 0.2756 m to 0.1737 m. The longitudinal speed responses in Figure 13c,d show that both controllers can follow the desired acceleration process, while the coordinated controller exhibits smaller speed fluctuation. From the perspective of stability maintenance, Figure 13f shows that the yaw-rate response under pure tracking control fluctuates more sharply, whereas the coordinated controller reduces the yaw-rate RMSE by 44.1%. Figure 13e,h further indicate that the sideslip angle and phase-plane trajectory are more concentrated under the coordinated controller, suggesting a lower tendency toward sideslip instability during the high-speed lane-changing process. In Figure 13g, the risk factor increases during the intensive steering stages, indicating that the proposed stability-margin-based mechanism can capture the rise in instability risk and trigger the dynamic adjustment of tracking and stability weights before severe sideslip occurs.

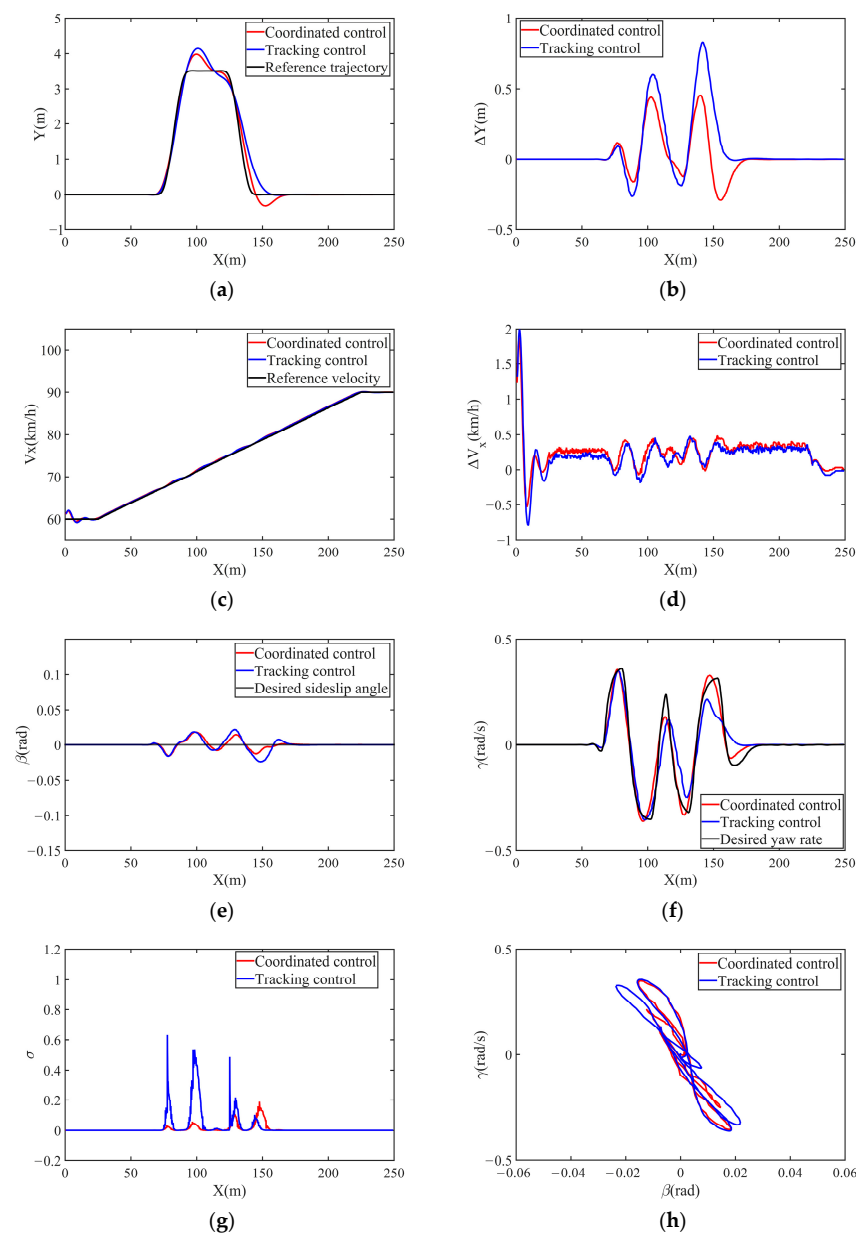


Figure 13. Simulation results under variable-speed high-adhesion DLC conditions. (a) Lateral displacement. (b) Lateral displacement error. (c) Longitudinal velocity. (d) Longitudinal velocity error. (e) Sideslip angle. (f) Yaw rate. (g) Risk factor. (h) Phase plane.

Table 4. Simulation comparison under variable-speed high-adhesion DLC conditions.

Evaluation Metrics	MaxAE				RMSE			
	ΔY (m)	Δv_x (km/h)	β (rad)	γ (rad/s)	ΔY (m)	Δv_x (km/h)	β (rad)	γ (rad/s)
Coordinated Control	0.4551	1.8750	0.0183	0.1243	0.1737	0.2927	0.0058	0.0449
Tracking Control	0.8327	1.9950	0.0236	0.2366	0.2756	0.3198	0.0062	0.0803

5.2.2. High-Speed and Low-Adhesion DLC Conditions

The parameter configuration for the simulation test under the high-speed, low-adhesion DLC condition is as follows: the vehicle speed is set to 90 km/h, and the road adhesion coefficient is selected as 0.3. The corresponding simulation results and quantitative evaluation metrics under this condition are presented in Figure 14 and Table 5, respectively.

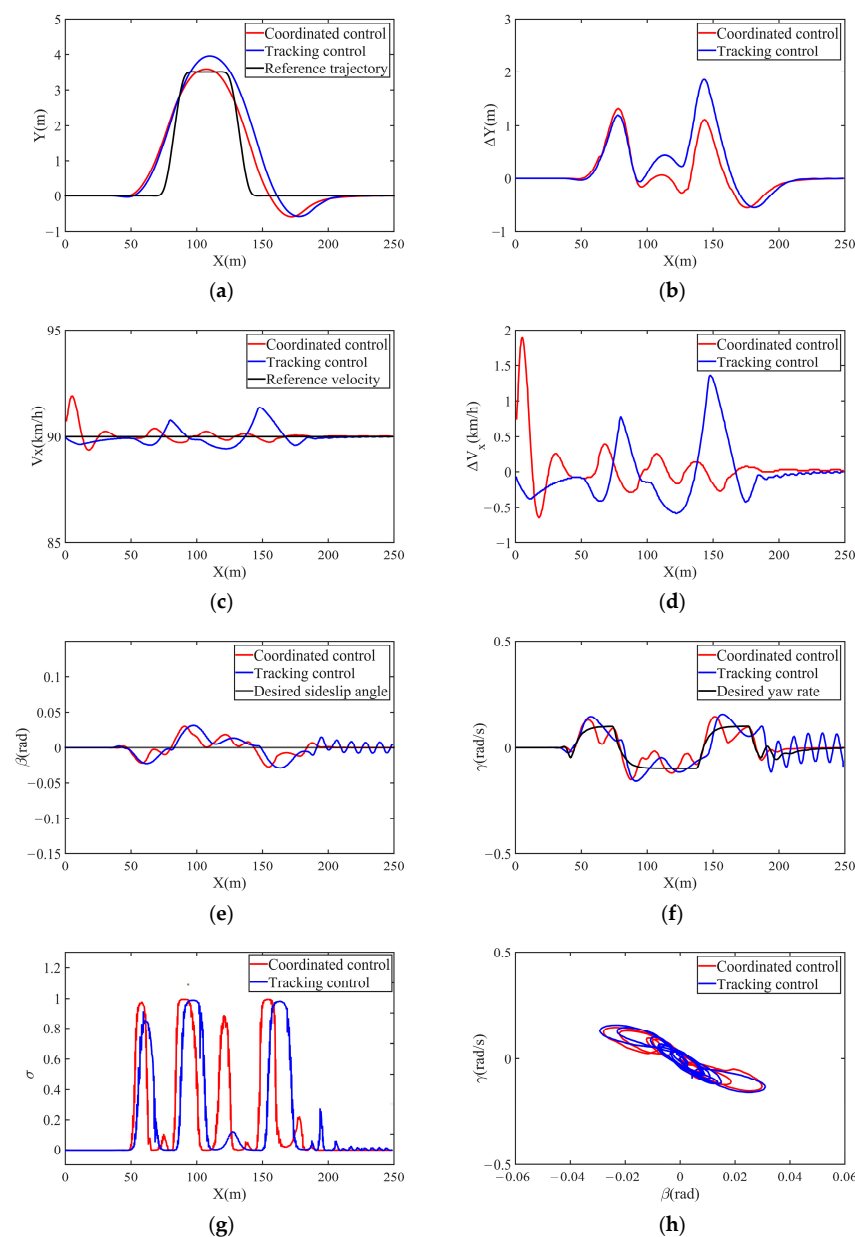


Figure 14. Simulation results under high-speed low-adhesion DLC conditions. (a) Lateral displacement. (b) Lateral displacement error. (c) Longitudinal velocity. (d) Longitudinal velocity error. (e) Sideslip angle. (f) Yaw rate. (g) Risk factor. (h) Phase plane.

Table 5. Simulation comparison under high-speed low-adhesion DLC conditions.

Evaluation Metrics	MaxAE				RMSE			
	ΔY (m)	Δv_x (km/h)	β (rad)	γ (rad/s)	ΔY (m)	Δv_x (km/h)	β (rad)	γ (rad/s)
Coordinated Control	1.3204	1.8900	0.0299	0.0825	0.4125	0.2397	0.0365	0.0123
Tracking Control	1.8595	1.3650	0.0248	0.1702	0.5569	0.4027	0.0637	0.0092

Under the extreme conditions of high speed and low adhesion, preventing vehicle instability becomes the primary control objective. As shown in Figure 14a,b and Table 5, the maximum lateral error of the coordinated controller is 1.3204 m, which is lower than the 1.8595 m obtained under pure tracking control. This result indicates that blindly pursuing path accuracy under low-adhesion conditions may aggravate nonlinear sideslip and lead to greater trajectory deviation. In contrast, when the risk factor increases, the coordinated controller appropriately relaxes the strict tracking demand and introduces additional yaw-moment regulation and torque reallocation to maintain the vehicle posture within a recoverable range, thereby improving both stability and DLC tracking performance. From the perspective of stability maintenance, Figure 14e,f,h show that the pure tracking controller leads to outward diffusion of the phase trajectory, larger sideslip-angle fluctuation, and high-frequency yaw-rate oscillation after the second steering transition. Specifically, the sideslip-angle RMSE under pure tracking control is 0.0637 rad, whereas the coordinated controller reduces it by 42.7%. The maximum yaw-rate error of pure tracking control reaches 0.1702 rad/s, indicating a severe oscillatory response, while the coordinated controller reduces this value to 0.0825 rad/s through active stability intervention, corresponding to a reduction of 51.5%. Combined with Figure 14c,d,g, the risk factor under pure tracking control remains close to one around the two steering peaks, suggesting that the vehicle operates in a sustained high-risk region. Although the risk factor of the coordinated controller also reaches one at several instants, its duration is shorter and it decreases rapidly, demonstrating that the stability-domain-aware coordination mechanism can intervene promptly and drive the vehicle state away from the instability-prone region. The speed fluctuation observed under coordinated control can be mainly attributed to the longitudinal acceleration constraint and torque redistribution under high-risk conditions. This local degradation in speed smoothness can be regarded as a reasonable trade-off, as it effectively improves yaw stability and avoids sustained high-risk operation under the high-speed low-adhesion condition.

6. Conclusions

This study proposed an ISOSP-LSTM-based stability-region identification model and a hierarchical coordinated control strategy for DDEVs to balance trajectory tracking accuracy and yaw stability under complex driving conditions. HIL results show that under the variable-speed high-adhesion DLC condition, the RMSEs of longitudinal velocity and lateral displacement are 0.2927 km/h and 0.1737 m, respectively. Under the high-speed low-adhesion DLC condition, compared with pure tracking control, the maximum yaw-rate error is reduced by 51.5%, the RMSE of sideslip angle is decreased by 42.7%, and the maximum lateral error is reduced from 1.8595 m to 1.3204 m. These results verify that the proposed strategy can adjust control authority according to the stability margin, thereby improving yaw stability while maintaining acceptable tracking accuracy.

Although the proposed strategy demonstrates promising performance, several limitations should be noted. The stability-region estimation relies on a simplified 2-DOF rational vehicle model, which does not fully capture suspension dynamics, load transfer,

actuator delay, and other high-order nonlinear effects. In addition, the rational tire model depends on fitted parameters, and its accuracy may vary with tire type, vertical load, and road adhesion. The SSA-LSTM predictor is trained on selected simulation data, so its generalization ability under unseen driving conditions still requires further validation. In the present study, the vehicle states are assumed to be ideally measurable. Future work will include real-vehicle testing, robustness analysis under vehicle-parameter perturbations, robust state estimation for practical implementation, adaptive tire-parameter identification, and dataset expansion across more speeds, adhesion levels, and maneuvers. In addition, the SSA-LSTM inference time and the overall coordinated controller execution time will be measured independently on the target domain controller and compared with the prescribed control period to provide a quantitative evaluation of real-time performance.

Author Contributions: D.C.: Methodology, writing—original draft preparation, writing—review and editing, and software. J.H.: Writing—review and editing, software, and supervision. Y.L.: Writing—original draft preparation and writing—review and editing. K.S.: Writing—review and editing. T.X.: Writing—review and editing and software. Y.H.: Supervision, funding acquisition, and resources. P.Z.: Writing—review and editing, supervision, funding acquisition, and resources. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Guangxi Science and Technology Major Program (AA23062056), the Technology Innovation Program of Hubei Province (2024BAB077), the Fundamental Research Funds for the Central Universities (104972026KFYjlb0012), and the Guangxi Special Hired Expert Program.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Danhua Chen, Tie Xu, and Yuanyi Huang are affiliated with SAIC GM-Wuling Automobile Co., Ltd., Liuzhou, China. The authors declare that these affiliations did not influence the results or interpretation of this study, and no conflicts of interest exist regarding the publication of this manuscript.

References

1. Sun, X.; Wang, Y.; Hu, W.; Cai, Y.; Huang, C.; Chen, L. Path tracking control strategy for the intelligent vehicle considering tire nonlinear cornering characteristics in the PWA form. *J. Frankl. Inst.* **2022**, *359*, 2487–2513. [\[CrossRef\]](#)
2. Zhang, H.; Liu, C.; Zhao, W. Segmented trajectory planning strategy for active collision avoidance system. *Green Energy Intell. Transp.* **2022**, *1*, 100002. [\[CrossRef\]](#)
3. Huang, Y.; Chen, Y. Vehicle lateral stability control based on shiftable stability regions and dynamic margins. *IEEE Trans. Veh. Technol.* **2020**, *69*, 14727–14738. [\[CrossRef\]](#)
4. Huang, Y.; Liang, W.; Chen, Y. Stability regions of vehicle lateral dynamics: Estimation and analysis. *J. Dyn. Syst. Meas. Control* **2021**, *143*, 051002. [\[CrossRef\]](#)
5. Wang, F.; Shen, T.; Zhao, M.; Ren, Y.; Lu, Y.; Feng, B.; Yin, G. Lane-change trajectory planning and control based on stability region for distributed drive electric vehicle. *IEEE Trans. Veh. Technol.* **2024**, *73*, 504–521. [\[CrossRef\]](#)
6. Liang, J.; Wang, F.; Feng, J.; Zhao, M.; Fang, R.; Pi, D.; Yin, G. A hierarchical control of independently driven electric vehicles considering handling stability and energy conservation. *IEEE Trans. Intell. Veh.* **2024**, *9*, 738–751. [\[CrossRef\]](#)
7. Daher, A.M.; Bardawil, C.R.; Daher, N.A. Vehicle stability based on g–g diagram through braking and driveline. In *Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017*; IEEE: New York, NY, USA, 2017; pp. 309–314. [\[CrossRef\]](#)
8. Liu, J.; Song, J.; Li, H.; Huang, H. Direct yaw-moment control of vehicles based on phase plane analysis. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2022**, *236*, 2459–2474. [\[CrossRef\]](#)
9. Wu, X.; Zhou, B.; Wen, G.; Long, L.; Cui, Q. Intervention criterion and control research for active front steering with consideration of road adhesion. *Veh. Syst. Dyn.* **2018**, *56*, 553–578. [\[CrossRef\]](#)
10. Bobier, C.G.; Joe, S.; Gerdes, J.C. Sliding surface envelope control: Keeping the vehicle within a safe state-space boundary. In *Proceedings of the ASME 2010 Dynamic Systems and Control Conference, Cambridge, MA, USA, 12–15 September 2010*; ASME: New York, NY, USA, 2010; Volume 2, pp. 149–156. [\[CrossRef\]](#)

11. Liu, Z.; Li, Y.; Li, W.; Li, Z.; Zhang, H.; Tan, X.; Wu, G. Research on control strategy of vehicle stability based on dynamic stable region regression analysis. *Front. Neurobot.* **2023**, *17*, 1149201. [[CrossRef](#)] [[PubMed](#)]
12. Chen, W.; Liang, X.; Wang, Q.; Zhao, L.; Wang, X. Extension coordinated control of four-wheel-independent-drive electric vehicles by AFS and DYC. *Control Eng. Pract.* **2020**, *101*, 104504. [[CrossRef](#)]
13. Wang, Q.; Wang, Z.; Wang, H.; Shang, Z.; Xu, W. Coordinated control of ARS and DYC for full X-by-wire distributed drive electric vehicles based on stability domain division in extension phase plane. *Control Eng. Pract.* **2025**, *162*, 106344. [[CrossRef](#)]
14. Zhu, Z.; Tang, X.; Qin, Y.; Huang, Y.; Hashemi, E. A survey of lateral stability criterion and control application for autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 10382–10399. [[CrossRef](#)]
15. Tamba, T.A.; Nazaruddin, Y.Y. Nonlinear stability analysis of vehicle side-slip dynamics using SOS programming. In *Proceedings of the 2018 5th International Conference on Electric Vehicular Technology (ICEVT), Surakarta, Indonesia, 30–31 October 2018*; IEEE: New York, NY, USA, 2018; pp. 1–6. [[CrossRef](#)]
16. Ribeiro, A.M.; Fioravanti, A.R.; Moutinho, A.; de Paiva, E.C. Nonlinear state-feedback design for vehicle lateral control using sum-of-squares programming. *Veh. Syst. Dyn.* **2022**, *60*, 743–769. [[CrossRef](#)]
17. Shen, T.; Wang, F.; Yan, Y.; Zhao, M.; Lu, Y.; Feng, B.; Yin, G. Stability analysis and control validation of DDEV in handling limit via SOS: A strategy based on stability region. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 17240–17252. [[CrossRef](#)]
18. Hu, X.; Chen, H.; Ren, Q.; Gong, X.; Wang, P.; Hu, Y. Estimation and expansion of vehicle stability region with sums of squares programming. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 2820–2831. [[CrossRef](#)]
19. Li, Y.; Wu, G.; Liu, K. Intelligent vehicle path tracking control strategy considering data-driven dynamic stable region constraints. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2024**, *238*, 3758–3770. [[CrossRef](#)]
20. Wang, C.; Zhang, Y.; Hu, X.; Qin, H.; Wang, G.; Qin, Y. *Real-Time Multidimensional Vehicle Dynamic Stability Domain Calculation and Its Application in Intelligent Vehicles*; SAE Technical Paper 2025-01-7324; SAE International: Warrendale, PA, USA, 2025.
21. Wang, W.; Ma, T.; Yang, C.; Chen, W.; Hu, Y. A path following lateral control scheme for four-wheel independent drive autonomous vehicle using sliding mode prediction control. *IEEE Trans. Transp. Electr.* **2022**, *8*, 3192–3207. [[CrossRef](#)]
22. Liu, K.; Gong, J.; Kurt, A.; Chen, H.; Ozguner, U. Dynamic modeling and control of high-speed automated vehicles for lane change maneuver. *IEEE Trans. Intell. Veh.* **2018**, *3*, 329–339. [[CrossRef](#)]
23. Zhai, L.; Wang, C.; Hou, Y.; Liu, C. MPC-based integrated control of trajectory tracking and handling stability for intelligent driving vehicle driven by four hub motor. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2668–2680. [[CrossRef](#)]
24. Zeng, X.H.; Duan, C.S.; Song, D.F.; Qian, Q.F.; Yang, D.P.; Zhang, X.M. Integrated motion control of interaction safety, stability, and path-tracking for front-and-rear-independent-drive electric vehicle. *IEEE Trans. Transp. Electr.* **2025**, *11*, 8573–8583. [[CrossRef](#)]
25. Zhang, Z.; Zheng, L.; Li, Y.; Li, S.; Liang, Y. Cooperative strategy of trajectory tracking and stability control for 4WID autonomous vehicles under extreme conditions. *IEEE Trans. Veh. Technol.* **2023**, *72*, 3105–3118. [[CrossRef](#)]
26. Zhang, L.; Sun, L.; Ding, X.; Wang, Z. Cooperative trajectory tracking and stability control with improved stability criterion for intelligent four-wheel-independent-drive electric vehicles. *Automot. Innov.* **2025**, *8*, 187–204. [[CrossRef](#)]
27. Lu, J.; Wang, R.; Wan, C. Intelligent vehicle path tracking and stability control method based on extension coordinated control of AFS and DYC. *IEEE Access* **2025**, *13*, 7353–7365. [[CrossRef](#)]
28. Zhang, H.; Dai, Y.; Zhu, C. Region stability analysis and precise tracking control of linear stochastic systems. *Appl. Math. Comput.* **2024**, *465*, 128402. [[CrossRef](#)]
29. Meng, F.; Wang, D.; Yang, P.; Xie, G.; Guo, F. Application of sum-of-squares method in estimation of region of attraction for nonlinear polynomial systems. *IEEE Access* **2020**, *8*, 14234–14243. [[CrossRef](#)]
30. Shen, T.; Yin, G.; Ren, Y.; Wang, F.; Feng, B.; Liang, J. Stability and maneuverability guaranteed torque distribution strategy of DDEV in handling limit: A novel LSTM-LMI approach. *IEEE/ASME Trans. Mechatron.* **2022**, *27*, 5647–5658. [[CrossRef](#)]
31. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **2020**, *8*, 22–34. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.