technologies

Article

Path Tracking Control for Four-Wheel Independent Steering and Driving Vehicles Based on Improved Deep Reinforcement Learning

Xia Hua, Tengteng Zhang, Xiangle Cheng * and Xiaobin Ning

College of Mechanical Engineering, Zhejiang University of Technology, 18 Chaowang Road, Hangzhou 310014, China; huaxia@zjut.edu.cn (X.H.); zhangtt0331@163.com (T.Z.); nxb@zjut.edu.cn (X.N.)
* Correspondence: chengxl@zjut.edu.cn

Abstract: We propose a compound control framework to improve the path tracking accuracy of a four-wheel independent steering and driving (4WISD) vehicle in complex environments. The framework consists of a deep reinforcement learning (DRL)-based auxiliary controller and a dual-layer controller. Samples in the 4WISD vehicle control framework have the issues of skewness and sparsity, which makes it difficult for the DRL to converge. We propose a group intelligent experience replay (GER) mechanism that non-dominantly sorts the samples in the experience buffer, which facilitates within-group and between-group collaboration to achieve a balance between exploration and exploitation. To address the generalization problem in the complex nonlinear dynamics of 4WISD vehicles, we propose an actor-critic architecture based on the method of two-stream information bottleneck (TIB). The TIB method is used to remove redundant information and extract high-dimensional features from the samples, thereby reducing generalization errors. To alleviate the overfitting of DRL to known data caused by IB, the reverse information bottleneck (RIB) alters the optimization objective of IB, preserving the discriminative features that are highly correlated with actions and improving the generalization ability of DRL. The proposed method significantly improves the convergence and generalization capabilities of DRL, while effectively enhancing the path tracking accuracy of 4WISD vehicles in high-speed, large-curvature, and complex environments.

Keywords: 4WISD vehicles; path tracking; deep reinforcement learning; experience replay; information bottleneck; actor-critic architecture; policy quality

Citation: Hua, X.; Zhang, T.; Cheng, X.; Ning, X. Path Tracking Control for four-wheel Independent Steering and Driving Vehicles Based on Improved Deep Reinforcement Learning. *Technologies* **2024**, *12*, 218. https://doi.org/10.3390/technologies12110218

Academic Editor: Juan Gabriel Avina-Cervantes

Received: 28 August 2024
Revised: 29 September 2024
Accepted: 30 September 2024
Published: 4 November 2024

1. Introduction

Four-wheel independent steering and driving (4WISD) vehicles have gained considerable research attention owing to their highly flexible path tracking control capabilities [1–5]. This control independence enhances the vehicle's adaptability and robustness in complex environments, enabling it to effectively respond to variable road conditions and disturbances [6–8]. However, the complex nonlinear characteristics inherent to 4WISD vehicles present significant challenges in designing path tracking controllers that exhibit fast response, high tracking accuracy, and strong disturbance rejection [9].

In recent years, various control algorithms have been proposed to address these challenges in the path tracking control of 4WISD vehicles. One widely used approach is motion decoupling, which separates lateral and longitudinal motions through independent control loops to track different targets [10–12]. This method simplifies the design of the control system, allowing for more precise and efficient control in each motion direction. Another key strategy is the hierarchical control structure, where the upper-layer controller computes the generalized forces for the vehicle, and the lower-layer controller optimally distributes these forces to each tire [13–16]. This hierarchical structure effectively manages the complex vehicle dynamics, improving both the robustness and response speed of the control system.

In terms of control algorithms, sliding mode control (SMC) has been widely applied due to its robustness against system uncertainties and external disturbances [17,18]. This introduces a sliding surface along which the system states move, to achieve effective control of nonlinear systems. Model predictive control (MPC) has also gained significant attention [19,20] for its ability to handle multivariable coupling and constraint conditions by predicting future system behaviors to optimize current control inputs. Proportional-integral-derivative (PID) control and its improved versions remain vital in engineering practice due to their simplicity and effectiveness [21,22]. However, these traditional path tracking control methods have limitations in fully addressing the complex challenges posed by 4WISD vehicles. Traditional model-based control methods pose challenges in managing unknown and highly dynamic external disturbances. These methods rely on a predefined vehicle dynamics model, which may fail to converge under drastic or unexpected environmental changes. As a result, these methods lack real-time adaptability to nonlinear and unpredictable conditions in 4WISD vehicles. In addition, the computational demands can become prohibitive when adjusting to such complex scenarios. Numerical errors tend to accumulate and propagate with each recursive step, which might not only decrease the computational accuracy of the system but also lead to divergence in control algorithms. In path tracking control, as the iteration number increases, the errors are gradually magnified, reducing control precision, especially when the unknown external disturbances strengthen the error accumulation effect. The randomness and unpredictability of external disturbances force the system to adjust control signals, further amplifying numerical errors and reducing the system's stability. Ultimately, the continuous accumulation of these errors could destabilize the system, significantly reducing path tracking accuracy and increasing the operational risk in dynamic environments.

Model-free methods for vehicle control have been applied to address the above-mentioned challenges [23–27]. Deep reinforcement learning (DRL) combines the advantages of reinforcement learning and deep learning, demonstrating great capability in real-time control [28,29]. Within the DRL framework, agents are modeled using deep neural networks and continuously interact with the environment and control system. Each action is rewarded based on feedback from the environment, and the training objective is to maximize the cumulative reward over time. Through interaction with continuous state-action spaces [30], DRL can adapt to the decision-making requirements of 4WISD vehicles under different external disturbances. While DRL provides flexibility and adaptability by interacting with continuous state-action spaces, it faces substantial challenges in control stability, convergence speed, and generalization to varying environments. These challenges stem from the tendency of DRL agents to overfit to the specific conditions on which they are trained, limiting their ability to generalize to diverse scenarios. Furthermore, DRL's dependence on large datasets for training can result in instability during the learning process. This issue is especially pronounced for 4WISD vehicles in dynamic or high-dimensional environments.

Based on the literature review, we propose a compound control framework for path tracking of 4WISD vehicles that uses an improved DRL approach. The main novelties of this study are as follows. Firstly, we propose a compound control framework that enhances DRL's adaptability and generalization by integrating a model-free DRL auxiliary controller with a model-based dual-layer controller. This compound approach capitalizes on the DRL's strengths in managing complex, high-dimensional environments, while the model-based controller ensures stability and improves decision-making in dynamic or unforeseen scenarios. As a result, the system is capable of generalizing across diverse operational conditions, maintaining accurate path tracking even under highly uncertain external disturbances. Secondly, we propose a group intelligent experience replay (GER) mechanism that treats the experience buffer as an intelligent entity. The experience buffer is categorized into three groups based on the prioritization of samples: discover, joiner, and risker. Coordination within and between groups is performed using training progress and non-dominated sorting, enabling adaptive balancing of exploration and exploitation.

This categorization enables the DRL agent to efficiently explore new strategies in unknown environments, while refining its learned policies in known situations. This leads to faster convergence, enabling the system to manage diverse operational scenarios with higher accuracy and stability. Thirdly, an actor-critic architecture based on the two-stream information bottleneck (TIB) is proposed. An information bottleneck approach is introduced into the critic network to minimize the mutual information between high-dimensional features and the target Q-value, thereby improving the critic's feature extraction ability and reducing generalization error. Meanwhile, a reverse information bottleneck approach is applied to the actor network to maximize the mutual information between features and actions. This approach balances learning the most compact state representation and preserving highly discriminative state-action correlations, and ensures that the DRL agent can generalize its learned policies to a wide range of unknown scenarios, improving its ability to adapt to diverse high-dimensional environments.

The remainder of this paper is structured as follows. The research background and related work are introduced in Section 2. The problem formulation is presented in Section 3. The implementation of the improved DRL-based compound control framework is elaborated in Section 4. The simulation results are compared in Section 5. The conclusions are summarized in Section 6.

2. Background and Related Work

2.1. DRL-Based Vehicle Control

Liu et al. compared the decision-making strategies for autonomous driving on a highway using different deep reinforcement learning algorithms, focusing on their implementation methods, performance metrics, and impact on driving efficiency and safety [31]. In terms of longitudinal control, Lin et al. found that DRL performs better as errors based on model predictive control increase [32]. Chen et al. integrated the methods of deep reinforcement learning and model predictive control for adaptive cruise control (ACC) of tracked vehicles, which enhanced the performance and efficiency of the ACC system [33]. Selvaraj et al. developed a DRL framework that accounts for passengers' safety and comfort and road usage efficiency [34]. For lateral control, Li et al. broke the vision-based lateral control system into a perception module and a control module, with the latter being trained with a reinforcement learning controller to improve performance on different tracks [35]. Peng et al. combined a DRL strategy with graph attention networks for autonomous driving planning [36]. In complex urban traffic scenarios, Li et al. used a DRL-based eco-driving strategy to optimize economy and travel efficiency. The authors also integrated safety measures and addressed additional challenges arising from real-time traffic elements, such as varying road conditions [37]. Sallab et al. proposed a DRL framework for autonomous driving to handle complex interactions with other vehicles and roadworks [38].

The complex dynamics in 4WISD vehicles make it difficult for existing DRL methods to achieve stable and efficient control. Current approaches often face challenges such as instability and poor convergence, due to the vehicle's high-dimensional state and action spaces. Therefore, developing more robust and effective DRL-based compound control frameworks is essential to enhance the path tracking performance of 4WISD vehicles.

2.2. Experience Replay Mechanism in DRL

Recent advancements in experience replay mechanisms for deep reinforcement learning have concentrated on several key areas. For experience selection and sampling strategy optimization, various methods have been proposed to enhance learning efficiency using intelligent selection mechanisms. Wei et al. and Li et al. presented quantum-inspired experience replay (QER) to balance the importance and diversity [39,40]. Zhu et al. developed prioritized experience replay (PER) to adjust sampling probabilities based on temporal difference (TD) errors [41]. Na et al. proposed emphasized experience replay (EER) to prioritize experiences that significantly impact algorithm performance [42]. Ye et al. introduced classified experience replay (CER) to adjust sampling ratios for different types

of experiences by classifying them into successful and failed attempts. This enhancement improved the training process, enabling the DRL model to learn more effectively from both positive and negative outcomes [43]. Regarding experience timeliness and freshness, Ma et al. incorporated freshness discount factors to increase the sampling probability of recent experiences [44]. Wang et al. employed annealed biased prioritized experience replay to account for experience timeliness [45]. To improve experience storage and memory management, researchers have focused on the effective utilization of limited experience storage space. Osei et al. proposed an enhanced sequential memory management (ESMM) to optimize replay memory usage by improving experience retention strategies [46]. Liu et al. developed two-dimensional replay buffers to enhance storage structures [47].

The non-stationary environments of these vehicles hinder the efficient utilization of training samples by existing methods. Insufficient adaptability to changing conditions and inefficiency in handling high-dimensional state-action spaces typically result in poor learning performance. It is necessary to develop more adaptive and robust experience replay strategies to optimize sample efficiency and enhance control performance in these complex systems.

2.3. Representation Learning in DRL

Recent advancements in information processing and representation learning are briefly introduced. For information compression and extraction, Xiang et al. employed variational information bottleneck techniques to infer fundamental tasks and learn essential skills [48]. Zou et al. developed the InfoGoal method, utilizing information bottleneck to learn compact goal representations, thereby improving policy optimality and generalization in goal-conditioned reinforcement learning [49]. Schwarzer et al. introduced the Self-Predictive Representations (SPR) approach. And the future state prediction and data augmentation were used to markedly increase sample efficiency [50]. Zhang et al. applied dual similarity metrics to learn robust latent representations that encode only task-relevant information, which demonstrated efficacy across various visual tasks [51]. In the domain of contrastive learning and self-supervision, Laskin et al. developed the CURL methods to extract high-level features from raw pixels, enhancing performance across multiple benchmarks [52]. Stooke et al. proposed the Augmented Temporal Contrast (ATC) task to decouple representation learning and policy learning, surpassing end-to-end reinforcement learning performance in most environments [53]. Regarding structured representations, Wei et al. introduced graph representation learning as an effective method for DRL agents to learn network entity relationships, enhancing path selection performance in network routing problems [54]. Qian et al. developed the DR-GRL framework, and combined disentangled representation learning with goal-conditioned visual reinforcement learning to improve sample efficiency and policy generalization [55]. For model-free reinforcement learning with high-dimensional image inputs, Yarats et al. proposed techniques to enhance training stability, demonstrating robustness to observational noise in control tasks [56].

Existing approaches often have difficulties in effectively capturing the complex dynamics or incur high computational costs, limiting their practical applicability. Developing more efficient state representation learning and improved sample information processing methods is crucial for achieving high DRL performance in controlling 4WISD vehicles.

3. Problem Formulation and Analysis

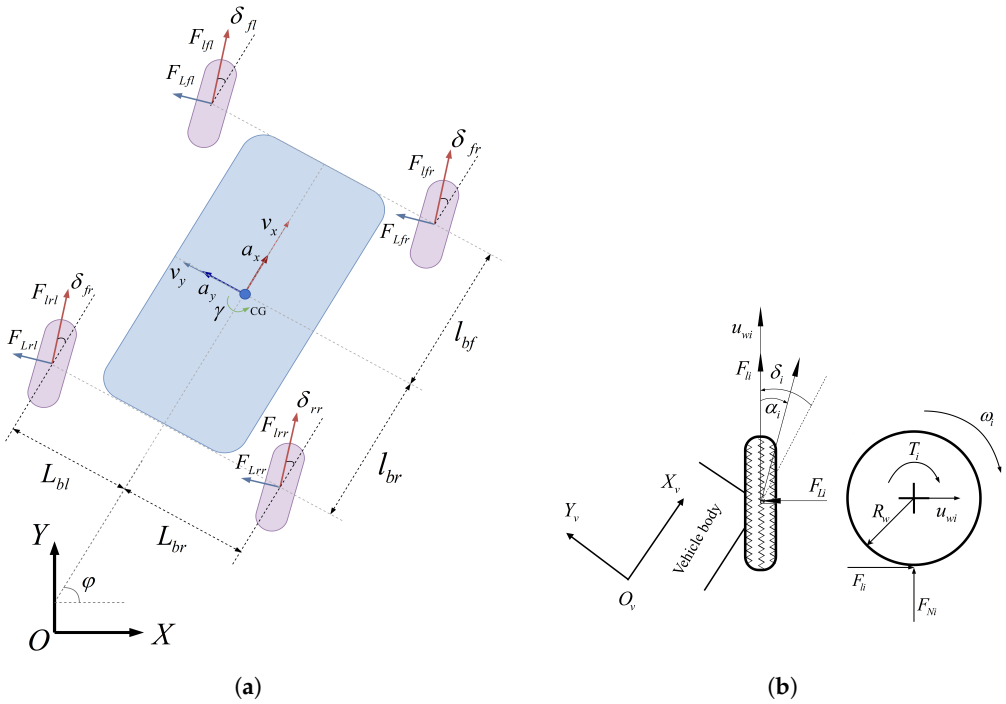
3.1. 4WISD Vehicle Dynamics Model

The 4WISD vehicle dynamics model comprises a vehicle body dynamics model and a tire model, as shown in Figure 1. The inertial reference frame and the vehicle body frame are represented as $OXYZ$ and $O_vX_vY_vZ_v$, respectively. Assume that the path tracking occurs on even road. The yaw, pitch, and roll motions of the vehicle body are mainly controlled by the longitudinal and lateral forces at each tire. For ease of notation, the four tires are indexed as $i = \mathrm{fl}, \mathrm{fr}, \mathrm{rl}, \mathrm{rr}$, which represent the front-left, front-right, rear-left, and rear-right tires, respectively, as represented in Figure 1a. In the tire coordinate system, the

longitudinal force, lateral force, vertical force, and steering angle of each tire are denoted by F_{li},F_{Li},F_{Ni}, and δ_i, respectively. The dynamics of the vehicle body, which are analyzed in the vehicle coordinate $O_vX_vY_vZ_v$, can be written as

$$\begin{cases} M(\dot{v}_x - v_y\gamma) = \sum_{i}^{\text{fl, fr, rl, rr}} (F_{li}\cos\delta_i - F_{Li}\sin\delta_i) \\ M(\dot{v}_y + v_x\gamma) = \sum_{i}^{\text{fl, fr, rl, rr}} (F_{li}\sin\delta_i + F_{Li}\cos\delta_i) \\ I_\gamma\dot{\gamma} = \sum_{i}^{\text{fl, fr, rl, rr}} [L_{bi}(F_{li}\cos\delta_i - F_{Li}\sin\delta_i) + l_{bi}(F_{li}\sin\delta_i + F_{Li}\cos\delta_i)] \end{cases} \quad (1)$$

where M and I_γ represent the vehicle mass and the moment of inertia; (L_{bi}, l_{bi}) denote the location of each tire in the coordinate $O_vX_vY_vZ_v$, with the center tied to the center of gravity of the vehicle; and $\dot{v}_x$, $\dot{v}_y$, and $\dot{\gamma}$ represent the vehicle's longitudinal acceleration, lateral acceleration, and yaw angle acceleration, respectively.



(**a**) (**b**)

Figure 1. 4WISD vehicle dynamics model [57]. (**a**) Vehicle body dynamics model. (**b**) Tire lateral dynamics model (**left**) and tire longitudinal dynamics model (**right**).

The vehicle accelerations in the vehicle body coordinate system are calculated through the forces exerted by the tires. In the design of a path tracking controller, the longitudinal and lateral forces at each tire are determined by controlling the steering angles and wheel torques, as presented in the tire dynamics model in Figure 1b.

In the tire coordinate system, we utilize the Magic Formula tire model to accurately capture the complex nonlinear characteristics of forces under varying slip ratios and slip angles. These nonlinear characteristics arise from the relationships between the longitudinal and lateral tire forces and their slip ratios and slip angles, which are influenced by factors such as tire deformation, rubber hysteresis effects, and variations in the contact patch.

The Magic Formula model employs a set of empirical equations to effectively describe these nonlinear relationships, thereby enhancing the performance of the vehicle control system and aiding in the optimization of the vehicle's handling and stability. The Magic Formula model allows for adjustments according to different tire types and operating

conditions, providing strong adaptability. The Magic Formula is well suited for real-time control systems in complex dynamic environments and can be expressed as [58]

$$y = D\sin(C\arctan\{Hx - E[Hx - \arctan(Hx)]\}) \tag{2}$$

where y represents either the longitudinal force F_{li} or the lateral force F_{Li}, and x represents either the longitudinal slip ratio λ_i or the slip angle α_i, respectively. The empirical equation is based on the following curve-fitting coefficients: the peak factor D, the stiffness factor H, the shape factor C, and the curvature factor E.

The longitudinal slip ratio λ_i is calculated as

$$\lambda_i = \begin{cases} (R_w\omega_i - u_{wi})/(R_w\omega_i) & R_w\omega_i \geq u_{wi} \\ (R_w\omega_i - u_{wi})/u_{wi} & R_w\omega_i < u_{wi} \end{cases} \tag{3}$$

where R_w and ω_i represent the dynamic tire radius and the angular velocity, and u_{wi} represents the actual speed at the center of the i^{th} tire. Note that $R_w\omega_i \geq u_{wi}$ and $R_w\omega_i < u_{wi}$ indicate the forward acceleration and braking of the tire, respectively. The tire speed u_{wi} can be calculated as

$$\begin{cases} u_{w,\,\text{fl}} = (v_x - L_{bl}\gamma)\cos(\delta_i) + (v_y + l_{bf}\gamma)\sin(\delta_i) \\ u_{w,\,\text{fr}} = (v_x + L_{br}\gamma)\cos(\delta_i) + (v_y + l_{bf}\gamma)\sin(\delta_i) \\ u_{w,\,\text{rl}} = (v_x - L_{bl}\gamma)\cos(\delta_i) + (v_y - l_{br}\gamma)\sin(\delta_i) \\ u_{w,\,\text{rr}} = (v_x + L_{br}\gamma)\cos(\delta_i) + (v_y - l_{br}\gamma)\sin(\delta_i) \end{cases} \tag{4}$$

The longitudinal and lateral dynamics models of the tire can be written as

$$\begin{cases} T_i = (F_{li} - F_{Ni}f_w)R_w + \omega_i I_w \\ \delta_i = \arctan((v_y \pm \omega_i l_{bi})/(v_x \pm \omega_i L_{bi})) - \alpha_i \end{cases} \tag{5}$$

where T_i denotes the drive torque, f_w denotes the rolling friction coefficient, I_w denotes the moment of inertia of each tire, and F_{Ni} denotes the tire's vertical force.

Based on the above analysis, we have established a complex vehicle dynamics model with seven degrees of freedom, which has a maximum of eight controllable inputs (i.e., four drive torques and four steering angles). By properly allocating the input parameters, we aim to effectively control the vehicle state to yield high path tracking performance.

The control input matrix of the vehicle dynamics model is given by

$$\mathbf{U} = \begin{bmatrix} T_{\text{fl}} & T_{\text{fr}} & T_{\text{rl}} & T_{\text{rr}} & \delta_{\text{fl}} & \delta_{\text{fr}} & \delta_{\text{rl}} & \delta_{\text{rr}} \end{bmatrix} \tag{6}$$

where the matrix **U** represents the torque and steering angle of the front left, front right, rear left, and rear right wheels, respectively. The vehicle state matrix is defined as **X**:

$$\mathbf{X} = \begin{bmatrix} v_x & v_y & \gamma \end{bmatrix} \tag{7}$$

where v_x, v_y, and γ denote the vehicle's longitudinal velocity, lateral velocity, and yaw velocity. To facilitate the design of DRL-based auxiliary controllers, the dynamic equations of the 4WISD vehicle can be rewritten into an affine nonlinear matrix form:

$$\begin{cases} \dot{\mathbf{X}} = \hat{\mathbf{A}}_n + \mathbf{B}\mathbf{u} \\ \mathbf{u} = \mathbf{C}\mathbf{U} \end{cases} \tag{8}$$

where **U** represents the control output matrix of the dual-layer controller;

$$\hat{\mathbf{A}}_n = \begin{bmatrix} \dot{v}_{nx} & \dot{v}_{ny} & \dot{\gamma}_n \end{bmatrix} \tag{9}$$

denotes the acceleration disturbance matrix of the vehicle body; **C** indicates the mapping matrix from the control quantities of the upper-layer controller to the control quantities of the lower-layer controller; and **B** indicates the inversion of vehicle mass matrix.

Since the vehicle's motion behavior is primarily determined by the interaction forces between the tires and the road surface, as the vehicle is a multi-body system where each tire may experience different external disturbances, it is critical to consider disturbances at the individual tire level to accurately capture the vehicle's dynamics. To realistically reflect the impact of external disturbances, we introduce a control-end force disturbance matrix:

$$\hat{\mathbf{F}}_n = \begin{bmatrix} \hat{F}_{l,\mathrm{fl}} & \hat{F}_{l,\mathrm{fr}} & \hat{F}_{l,\mathrm{rl}} & \hat{F}_{l,\mathrm{rr}} & \hat{F}_{L,\mathrm{fl}} & \hat{F}_{L,\mathrm{fr}} & \hat{F}_{L,\mathrm{rl}} & \hat{F}_{L,\mathrm{rr}} \end{bmatrix} \tag{10}$$

which corresponds to the perturbed lateral and longitudinal forces acting on each tire. Note that $\hat{F}_{l,i}$ and $\hat{F}_{L,i}$ represent the longitudinal disturbance force and the lateral disturbance force acting on each tire, respectively. The external disturbances in Equation (10) can propagate through the tire dynamics model to the vehicle body dynamics model, resulting in the acceleration disturbance matrix $\hat{\mathbf{A}}_n$. This method allows for a direct and accurate representation of how external disturbances affect the vehicle's dynamics and supports the development of more robust controllers to enhance system resilience against external uncertainties.

3.2. Transition Model for DRL

The state variable **s** of the 4WISD vehicle includes measurable states **o** and unmeasurable disturbances **d**. We employ DRL as an auxiliary controller rather than for direct end-to-end vehicle control, and assume that the disturbance **d** is known, thus utilizing the Markov decision process $p(\dot{\mathbf{s}}|\mathbf{s} = \mathbf{o}, \mathbf{a})$ to facilitate the training of the DRL [47]. In terms of the auxiliary controller in path tracking control of 4WISD vehicles, the optimization problem can be formulated as follows:

$$\begin{aligned} \mathbf{a}^* = \quad & \arg\min \sum_{i=1}^{N} \|\mathbf{X}^* - f_H(\mathbf{X}_i, f_l([\mathbf{X}_i, f_a(\mathbf{s}_i, \mathbf{a}_i) + f_{u_c}(\mathbf{e}_i, \mathbf{u}_{ci})], \mathbf{U}_i))\|^2 \\ \text{s.t.} \quad & \mathbf{a}_{\mathrm{lb}} \le \mathbf{a}_i \le \mathbf{a}_{\mathrm{ub}} \\ & \mathbf{u}_{\mathrm{lb}} \le \mathbf{u}_i \le \mathbf{u}_{\mathrm{ub}} \\ & \mathbf{U}_{\mathrm{lb}} \le \mathbf{U}_i \le \mathbf{U}_{\mathrm{ub}} \\ & \mathbf{s}_{\mathrm{lb}} \le \mathbf{s}_i \le \mathbf{s}_{\mathrm{ub}} \\ & \mathbf{e}_{\mathrm{lb}} \le \mathbf{e}_i = \|\mathbf{X}^* - f_H(\mathbf{X}_i, f_l([\mathbf{X}_i, f_a(\mathbf{s}_i, \mathbf{a}_i) + f_{u_c}(\mathbf{e}_i, \mathbf{u}_{ci})], \mathbf{U}_i))\|^2 \le \mathbf{e}_{\mathrm{ub}} \end{aligned} \tag{11}$$

where f_H denotes the dynamic function of the 4WISD vehicle; f_a denotes the auxiliary controller; f_{u_c} denotes the upper-layer controller; f_l denotes the lower-layer controller; $\mathbf{X}^* = \begin{bmatrix} v_{x_{\mathrm{ref}}} & v_{y_{\mathrm{ref}}} & \gamma_{\mathrm{ref}} \end{bmatrix}$ represents the reference path states of the longitudinal velocity, lateral velocity, and yaw velocity obtained from the lateral displacement Y and yaw angle ϕ through differential operation, respectively; and **a** represents the DRL action. Since the DRL auxiliary controller is expected to achieve real-time control of the vehicle based on the vehicle's current state, we define the state of the DRL as

$$\mathbf{s} = \begin{bmatrix} v_x & v_y & \gamma & \mathbf{e} & \mathbf{u}_c & \hat{\mathbf{A}}_n \end{bmatrix} \tag{12}$$

where $\mathbf{e} = \begin{bmatrix} e_{v_x} & e_Y & e_\phi \end{bmatrix}$ denotes the errors in longitudinal velocity, lateral displacement, and heading angle between the current and ideal vehicle states. The output action **a** of the DRL auxiliary controller has the same form as the control variable **U** of the vehicle system:

$$\mathbf{a} = \begin{bmatrix} F_{ax} & F_{ay} & M_{a\phi} \end{bmatrix} \tag{13}$$

The reward function is designed as

$$r = \begin{cases} -\left(K_r \mathbf{e}\mathbf{e}^T + K_f \dot{\mathbf{a}}\dot{\mathbf{a}}^T\right) & \|\mathbf{e}\| < \mathbf{e}_{\text{ub}} \text{ and } \mathbf{s}_{\text{lb}} \leq \mathbf{s}_i \leq \mathbf{s}_{\text{ub}} \\ -K_b & \text{otherwise} \end{cases} \tag{14}$$

where K_r is the positive gain parameter for the path tracking error, K_f is the positive gain parameter for the stability of continuous control, $\mathbf{e}_{\text{ub}}$ denotes the threshold for tracking error, K_b is the penalty term, and $\dot{\mathbf{a}}$ represents the rate of change of time in the auxiliary control variable.

4. Compound Control Framework Based on Improved DRL

The twin-delayed deep deterministic policy gradient algorithm (TD3) employs two independent Q-networks and adopts the smaller Q-value for policy updates, thereby suppressing overestimation. Moreover, TD3's deterministic policy gradient is suitable for continuous action space problems [59]. This algorithm provides enhanced stability for path tracking control of 4WISD vehicles. In what follows, the improved control framework is integrated with TD3.

4.1. Compound Control Framework

To address the issue of lateral and longitudinal coupling in path tracking control of 4WISD vehicles, we employ a dual-layer control framework in the model-based controller, which consists of an upper-layer controller based on nonlinear model predictive control (NMPC) and a lower-layer controller based on sequential quadratic programming (SQP).

The current state error $\mathbf{e}$ of the vehicle is calculated using the reference path information and the current vehicle state. In the upper-layer controller, the NMPC algorithm is applied with the cost function defined as follows:

$$\begin{aligned} J(t_k) \quad &= \min \left[\sum_{n=1}^{N} \|\mathbf{e}(t_{k+n}|t_k)\|^2_{\mathbf{Q}_{(k)}} + \sum_{n=1}^{N} \|\Delta \mathbf{u}_c(t_{k+n}|t_k)\|^2_{\mathbf{R}_{(k)}} + \rho \epsilon^2 \right] \\ \text{s.t.} \quad & F_{x,\text{lb}} \leq F_x \leq F_{x,\text{ub}} \\ & F_{y,\text{lb}} \leq F_y \leq F_{y,\text{ub}} \\ & M_{\phi,\text{lb}} \leq M_\phi \leq M_{\phi,\text{ub}} \end{aligned} \tag{15}$$

where $\mathbf{Q}_{(k)}$ and $\mathbf{R}_{(k)}$ represent the state weighting matrix and the control increment weighting matrix, ϵ denotes the relaxation factor to avoid the absence of feasible solutions, $\mathbf{e}(t_{k+n}|t_k)$ represents the error state at time t_k, and $\mathbf{u}_c(t_{k+n}|t_k)$ represents the generalized control quantities at the time t_{k+n}. Note that the generalized control quantities $\mathbf{u}_c$ are determined to calculate the vehicle state at the future moment:

$$\mathbf{u}_c = \begin{bmatrix} F_{cx} & F_{cy} & M_{c\phi} \end{bmatrix} \tag{16}$$

where F_{cx}, F_{cy}, and $M_{c\phi}$ represent the generalized longitudinal force, lateral force, and yaw moment of the vehicle body, respectively.

Then, we allocate the generalized force $\mathbf{u}_c$ from the upper-layer controller into the lateral and longitudinal forces for each tire. A lower-layer controller based on SQP is used to achieve the optimal distribution of the generalized force $\mathbf{u}_c$. The cost function in the SQP algorithm is defined as

$$
\begin{aligned}
f \quad &= \min\left[w_1 \sum_{i}^{\text{fl,fr,rl,rr}} \frac{F_{li}^2 + F_{Li}^2}{\mu^2 F_{Ni}^2} + w_2 \left(\max\left(\frac{F_{li}^2 + F_{Li}^2}{\mu^2 F_{Ni}^2} \right) - \min\left(\frac{F_{li}^2 + F_{Li}^2}{\mu^2 F_{Ni}^2} \right) \right)^2 \right] \\
\text{s.t.} \quad & F_{l,\text{lb}} \le F_{li} \le F_{l,\text{ub}},\ F_{L,\text{lb}} \le F_{Li} \le F_{L,\text{ub}} \\
& 0 \le \frac{F_{li}^2 + F_{Li}^2}{\mu^2 F_{Ni}^2} \le 1 \\
& F_x = \sum_{i}^{\text{fl,fr,rl,rr}} \left(F_{li} \cos\delta_i - F_{Li} \sin\delta_i \right) \\
& F_y = \sum_{i}^{\text{fl,fr,rl,rr}} \left(F_{li} \sin\delta_i + F_{Li} \cos\delta_i \right) \\
& M_\phi = \sum_{i}^{\text{fl, fr, rl, rr}} \left[L_{bi} \left(F_{li} \cos\delta_i - F_{Li} \sin\delta_i \right) + l_{bi} \left(F_{li} \sin\delta_i + F_{Li} \cos\delta_i \right) \right]
\end{aligned}
\tag{17}
$$

where w_1 and w_2 represent the weighting coefficients for tire balancing. Using the tire dynamics model, the lateral and longitudinal forces on each tire can be transformed into the drive torques and steering angles, resulting in the control variable **U** at the next moment. Then, the control variable is transmitted to the vehicle body dynamics model to achieve closed-loop path tracking control of 4WISD vehicles.

Even though the aforementioned dual-layer controller can be used to decouple the longitudinal and lateral motion, the control capability for a 4WISD vehicle subjected to external disturbances is still limited. The DRL-based controller, which is a data-driven algorithm, possesses super adaptability for the improvement of control performance. As depicted in Figure 2, a compound control framework is proposed, which mainly consists of a model-based dual-layer control loop and a model-free DRL-based auxiliary control loop. The current vehicle state is input into the upper-layer controller to obtain the required generalized forces for the next time step. Simultaneously, a well-trained DRL policy network produces an extra control term **a** based on the vehicle state information **s** for compensation. As such, a new upper layer control variable **u** is obtained as

$$
\mathbf{u} = \mathbf{u}_c + \mathbf{a} = \begin{bmatrix} F_{cx} + F_{ax} & F_{cy} + F_{ay} & M_{c\phi} + M_{a\phi} \end{bmatrix} \tag{18}
$$

By integrating the DRL-based auxiliary controller with the upper-layer controller, the lower-layer controller transforms the generalized force vector **u** into an end control variable **U**. The control stability proof of the proposed compound control framework is given in Appendix A.

4.2. Group Intelligent Experience Replay

The experience replay mechanism in DRL stores transition samples obtained from the agent's interactions with the environment in a replay buffer and randomly extracts small batches of samples to update the parameters of the value network or policy network. This process can break the temporal correlation between samples to facilitate the convergence of the agent.

To improve the stability and convergence of the DRL training process and avoid becoming trapped into local optima, the principles of group intelligence optimization are incorporated into the experience replay mechanism. Data in the experience buffer are regarded as an intelligent group [60], and the samples in the experience buffer are divided into three distinct functional groups based on the TD error and advantage function value. Non-dominated sorting and training progress are then introduced for within-group and between-group collaboration to optimize data replay and storage [61]. The discover group provides a better direction for the convergence of the DRL training process by prioritizing the learning of novel state-action pairs. The joiner group focuses on replaying a higher proportion of excellent samples around the discover group, reinforcing and optimizing

known high-value policies. The risker group filters out low-quality samples to avoid learning from high-risk samples that may cause overfitting.

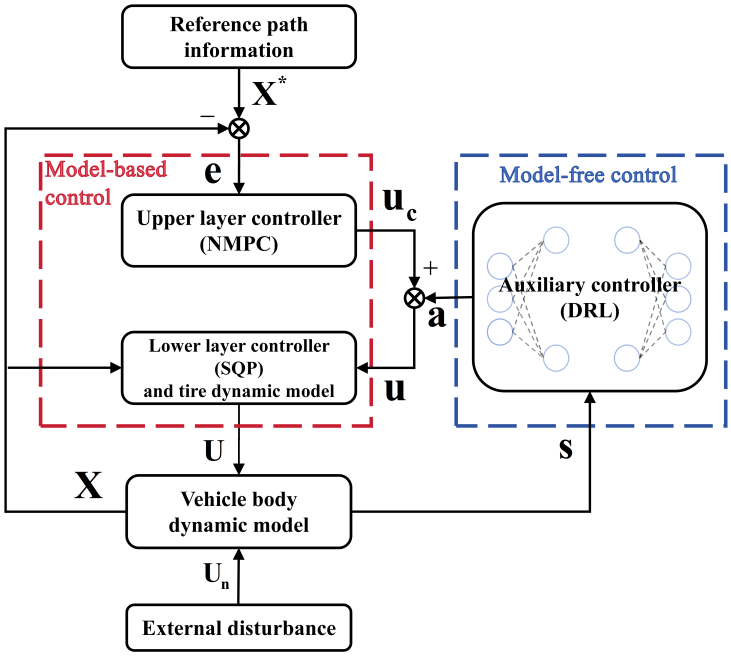


Figure 2. Schematic of the proposed compound control framework.

During the training process, a new transition sample $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$ is obtained from the interaction between the agent and the environment, from which the TD error τ_i and advantage function value A_i are calculated as

$$\tau_i = r_i + \gamma Q_{\phi'}(\mathbf{s}_{i+1}, \mu_{\theta'}(\mathbf{s}_{i+1})) - Q_{\phi}(\mathbf{s}_i, \mathbf{a}_i) \tag{19}$$

$$A_i = Q_{\phi}(\mathbf{s}_i, \mathbf{a}_i) - V_{\phi}(\mathbf{s}_i) \tag{20}$$

where Q_{ϕ} and $\mu_{\theta'}$ represent the target critic network and actor network, respectively, and V_{ϕ} represents the state value function. Subsequently, the initial priority P_i of the i^{th} sample is calculated as

$$P_i = |\tau_i| + \lambda |A_i| \tag{21}$$

The sample $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$ and the associated parameters P_i, τ_i, and A_i are stored in the experience replay buffer Ω. When the replay criteria are met, the samples in the experience replay buffer are divided into three groups based on priority P_i. As shown in Figure 3, the sample ratio coefficients for each group are denoted as $S_{\alpha 0}$, $S_{\beta 0}$, and $S_{\gamma 0}$, respectively, and they must satisfy the following conditions for $S_{\alpha 0}, S_{\beta 0}, S_{\gamma 0} \in [0, 1]$, and $S_{\alpha 0} + S_{\beta 0} + S_{\gamma 0} = 1$:

(1) Group D_{discover}: Samples with priorities in the top $S_{\alpha 0}$ percentile.
(2) Group D_{joiner}: Samples with priorities in the top $S_{\alpha 0} + S_{\beta 0}$ percentile but not in the top $S_{\alpha 0}$ percentile.
(3) Group D_{risker}: Samples with priorities in the bottom $S_{\gamma 0}$ percentile.

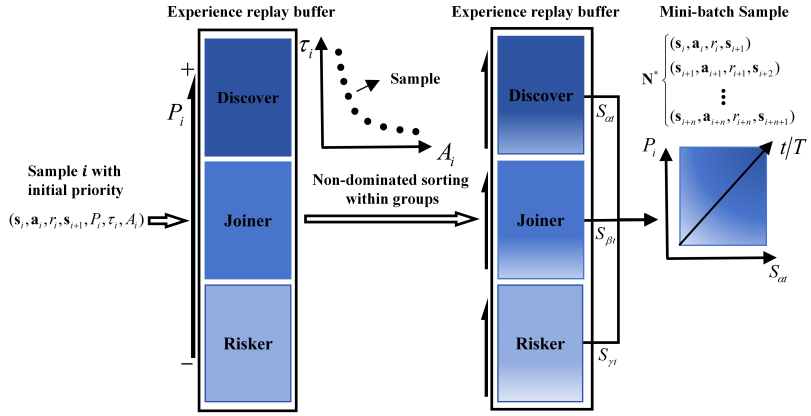


Figure 3. Schematic of the proposed group intelligent experience replay.

To effectively and accurately assess the sample quality without excessive time consumption, we select the top two mini-batch samples from each group D_k based on priority P_i. These samples are used for non-dominated sorting to obtain the Pareto frontier F_k for each group, which enables within-group collaboration:

$$F_k = \{s_i \in D_k \mid \forall s_j \in D_k, \tau_j < \tau_i \text{ and } A_j < A_i\} \tag{22}$$

where $k =$ discover, joiner, or risker.

After that, we use the Pareto frontier F_k to ensure that the following updated priorities are ranked higher than those from other samples:

$$\bar{P}_i = P_i + \min P_k, \quad i \in F_k \tag{23}$$

The sampling proportions for the three groups are dynamically adjusted for between-group collaboration according to the current training step t and the total number of training steps T. To prevent probability overflow, the priorities in each group are normalized as

$$\begin{cases} N_\alpha = \exp(\text{avg}_{\text{discover}}) \Big/ \sum_i \exp(\text{avg}_i) \\ N_\beta = \exp\left(\text{avg}_{\text{joiner}}\right) \Big/ \sum_i \exp(\text{avg}_i) \\ N_\gamma = \exp(\text{avg}_{\text{risker}}) \Big/ \sum_i \exp(\text{avg}_i) \end{cases} \tag{24}$$

where avg_i represents the average priority of samples within group $i \in$ (discover, joiner, risker), and N_α, N_β, and N_γ represent the normalization factors. Then, the updated priorities are defined as

$$\begin{cases} S_{\alpha t} = S_{\alpha 0}(1 - t/T)N_\alpha \\ S_{\beta t} = (S_{\beta 0}(1 - t/T) + S_{\alpha 0}(t/T))N_\beta \\ S_{\gamma t} = (1 - S_{\alpha 0} - S_{\beta 0})N_{\gamma 0} \end{cases} \tag{25}$$

where $S_{\alpha t}$, $S_{\beta t}$, and $S_{\gamma t}$ denote the updated sampling proportions for three groups, respectively. Samples are drawn from D_{discover}, D_{joiner}, and D_{risker} according to the values of $S_{\alpha t}$, $S_{\beta t}$, and $S_{\gamma t}$ to obtain replay samples. In the initial stage of training, $S_{\beta t}$ is large, which allows for the exploration of new and valuable state-action pairs. As training progresses, $S_{\beta t}$ gradually decreases while $S_{\alpha t}$ increases, which progressively transfers valuable samples discovered during exploration to D_{joiner} for sufficient utilization. However, $S_{\gamma t}$ remains

small to ensure that a low proportion of low-priority samples participates in training, which enhances the model's generalization capability.

To correct for biases introduced by priority sampling, it is necessary to apply importance sampling correction to the loss function during training. For each sample $(\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}_{i+1})$ in a mini-batch sample, the importance weight o_i is computed as follows:

$$o_i = (P_i/D)^{-\zeta} \tag{26}$$

where D represents the sum of priorities of all samples, and ζ controls the intensity of the correction. Applying the importance weight o_i to the loss function yields a weighted loss function:

$$L_w = \frac{1}{E}\sum_{i}^{E} o_i L_i \tag{27}$$

where L_i represents the mean squared error loss function for the i^{th} sample, and E represents the number of the mini-batch sample.

At the end of each training step, it is essential to update the sample priorities and substitute these updated priorities into the experience replay buffer. The proposed group intelligence experience replay mechanism adjusts sample priorities, balances exploration and exploitation, and enhances sample utilization efficiency. The proposed method provides greater sample utilization efficiency compared to traditional random experience replay. By employing intelligent sample classification and priority adjustment, GER adaptively prioritizes the most valuable samples for policy improvement, significantly enhancing the performance of reinforcement learning models in complex environments. This allows GER to better accelerate convergence and improve system robustness, particularly when dealing with sparse rewards or high-dimensional environments.

4.3. Actor-Critic Architecture Based on TIB

The integration of information bottleneck (IB) techniques into reinforcement learning establishes a self-supervised learning and an adaptive compression mechanism. By maximizing the mutual information achieved by arbitrary policies within short time windows, this approach generates a dense and immediate learning signal and addresses the sparse reward problem inherent in traditional reinforcement learning. The derivation of a lower bound on mutual information facilitates the adaptive adjustment of Lagrange multipliers, enabling maximal information compression while retaining sufficient task-relevant information. This methodology can provide an informative learning signal, and ultimately generate a compact and effective objective representation. Consequently, it significantly enhances the generalization capability and learning efficiency of goal-conditioned reinforcement learning [49]. This work provides theoretical guidance for integrating information theory into deep reinforcement learning.

In the standard actor-critic architecture, the critic network estimates the state-action value function, while the actor network generates policies. Building upon this framework, we employ the two-stream information bottleneck (TIB) [62] to enhance the algorithm's generalization capability and policy quality, as shown in Figure 4. In the critic network, we integrate an IB module prior to Q-value estimation, which extracts a D dimensional compact representation $\mathbf{z}_t$ from high-dimensional features $\mathbf{h}_t^c$. The representation $\mathbf{z}_t$ minimizes mutual information with high-dimensional features while preserving essential information for Q-value estimation. For the actor network, we integrate a reverse information bottleneck (RIB) module before the policy generation module, which extracts a K-dimensional expressive representation $\mathbf{u}_t$ from high-dimensional features $\mathbf{h}_t^a$. The representation $\mathbf{u}_t$ maximizes mutual information with actions and retains discriminative state-action correlations. The

extracted latent representations $\mathbf{z}_t$ and $\mathbf{u}_t$ are subsequently utilized in the critic and actor networks, respectively, to formulate the state-action value function and policy as follows:

$$Q_\phi(\mathbf{s}_t, \mathbf{a}_t) = f_\phi^{(l_c)}(\mathbf{z}_t(\mathbf{h}_t^c(\mathbf{s}_t, \mathbf{a}_t))) \tag{28}$$

$$\pi_\theta(\mathbf{s}_t) = g_\theta^{(l_a)}(\mathbf{u}_t(\mathbf{h}_t^a(\mathbf{s}_t))) \tag{29}$$

where l_c and l_a represent the number of network layers after the IB module and RIB module, respectively. The IB modules and RIB modules are simultaneously optimized in a single training session.

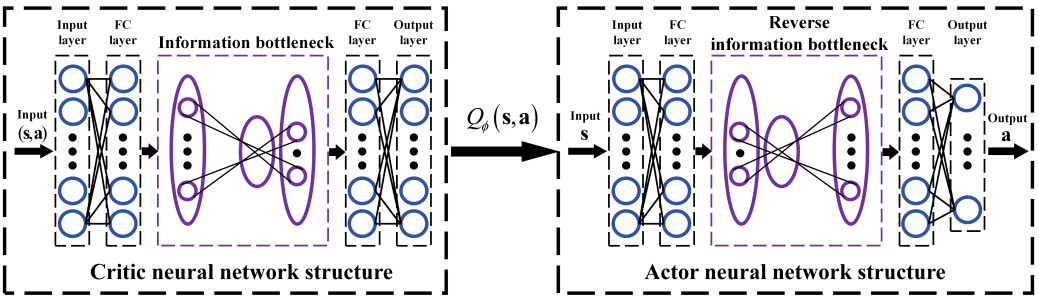


Figure 4. Diagram of the proposed actor-critic architecture based on TIB.

In the critic network, we introduce an IB module to enhance the network's generalization and sample efficiency. By compressing redundant information from high-dimensional features and preserving essential information for Q-value estimation, the IB module enables the network to learn state-action value functions more effectively. We aim to reduce the complexity of high-dimensional inputs, extract the most relevant features, and improve the accuracy of Q-value estimates to enhance the model's adaptability across complex environments. Based on these design objectives, we formulate the training loss function for the critic network as

$$L(\phi) = \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim E}\left[(Q(\mathbf{s}_t, \mathbf{a}_t) - y_t)^2\right] + \xi \mathrm{KL}(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t)) \tag{30}$$

where ϕ represents the parameters of the critic network, y_t is the target value for the TD, ξ is the balancing coefficients, $p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t)$ is the conditional distribution defined by the IB module, and $r(\mathbf{z}_t)$ is the prior distribution of $\mathbf{z}_t$. Note that the expectation $\mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim E}\left[(Q(\mathbf{s}_t, \mathbf{a}_t) - y_t)^2\right]$ measures the TD error, ensuring accurate Q-value estimation, and the expectation $\mathrm{KL}(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t))$ implements the IB by minimizing the Kullback–Leibler (KL) divergence between the conditional distribution of $\mathbf{z}_t$ and prior distribution of $r(\mathbf{z}_t)$, and constraining the information flow from high-dimensional features to achieve feature compression and selection. After applying the aforementioned design, the critic network maintains accurate Q-value estimation while improving its generalization capability in complex and dynamic environments.

In the actor network, we introduce an RIB module to enhance the quality and expressiveness of the policy. Using the RIB module, we aim to maximize the mutual information between state representations and actions, preserve critical information necessary for policy generation, enhance the discriminative power of state representations to better differentiate the value of various actions, and improve the policy's exploratory capabilities to generate more diverse and effective actions. Based on these design objectives, we formulate the training loss function for the actor network as

$$J(\theta) = \mathbb{E}_{\mathbf{s}_t \sim E}\left[\iota^t Q(\mathbf{s}_t, \mathbf{a}_t)\right] + \kappa \mathrm{KL}(p(\mathbf{u}_t|\mathbf{s}_t), r(\mathbf{u}_t)) \tag{31}$$

where θ refers to the parameters of the actor network, $\iota \in (0, 1]$ is the discount factor, κ is the balancing coefficients, $p(\mathbf{u}_t|\mathbf{s}_t)$ is the conditional distribution defined by the RIB module,

and $r(\mathbf{u}_t)$ is the prior distribution. In Equation (31), the expectation $\mathbb{E}_{\mathbf{s}_t \sim E}\left[\iota^t Q(\mathbf{s}_t, \mathbf{a}_t)\right]$ is the standard policy gradient term, aiming to maximize the expected cumulative reward to ensure the generated policy yields high returns. The expectation $\mathrm{KL}(p(\mathbf{u}_t|\mathbf{s}_t), r(\mathbf{u}_t))$ implements the RIB by maximizing the KL divergence between the conditional distribution of $\mathbf{u}_t$ and prior distribution of $r(\mathbf{u}_t)$, thereby increasing the mutual information between the state representation $\mathbf{u}_t$ and actions, preserving more policy-relevant information. Through this design, the actor network is capable of generating more precise and effective policies while maintaining sufficient exploratory capabilities to adapt to complex decision-making environments.

For the IB module in the critic network, the KL divergence can be formulated as

$$\mathrm{KL}(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t)) = \mathbb{E}_{\mathbf{z}_t}[\log(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t)) - \log(r(\mathbf{z}_t))] \tag{32}$$

where $p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t)$ represents the posterior distribution of $\mathbf{z}_t$ conditioned on the given $\mathbf{s}_t$ and $\mathbf{a}_t$.

For the RIB module in the actor network, the KL divergence can be formulated as

$$\mathrm{KL}(p(\mathbf{u}_t|\mathbf{s}_t), r(\mathbf{u}_t)) = \mathbb{E}_{\mathbf{u}_t}[\log(p(\mathbf{u}_t|\mathbf{s}_t)) - \log(r(\mathbf{u}_t))] \tag{33}$$

where $p(\mathbf{u}_t|\mathbf{s}_t)$ represents the posterior distribution of $\mathbf{s}_t$ conditioned on the given $\mathbf{u}_t$, and $r(\mathbf{u}_t)$ denotes the prior distribution of $\mathbf{u}_t$. The standard normal distributions for $r(\mathbf{z}_t)$ and $r(\mathbf{u}_t)$ are used to simplify computation and ensure model stability.

We update the network parameters based on the loss function defined above, the general rules for updating the critic and actor networks are presented as follows:

$$\nabla_\phi L(\phi) = \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim E}\left[(Q(\mathbf{s}_t, \mathbf{a}_t) - y_t)\nabla_\phi Q(\mathbf{s}_t, \mathbf{a}_t)\right] + \tilde{\varsigma}\nabla_\phi \mathrm{KL}(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t)) \tag{34}$$

$$\nabla_\theta J(\theta) = \mathbb{E}_{\mathbf{s}_t \sim E}[\nabla_{\mathbf{a}} Q(\mathbf{s}_t, \mathbf{a}_t)\nabla_\theta \pi_\theta(\mathbf{s})] + \kappa\nabla_\theta \mathrm{KL}(p(\mathbf{u}_t|\mathbf{s}_t), r(\mathbf{u}_t)) \tag{35}$$

The IB module in the critic network and RIB module in the actor network are

$$\nabla_\psi \mathrm{KL}(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t)) = \nabla_\psi \mathbb{E}_{\mathbf{z}_t}[\log(p(\mathbf{z}_t|\mathbf{s}_t, \mathbf{a}_t)) - \log(r(\mathbf{z}_t))] \tag{36}$$

$$\nabla_\omega \mathrm{KL}(p(\mathbf{u}_t|\mathbf{s}_t), r(\mathbf{u}_t)) = \nabla_\omega \mathbb{E}_{\mathbf{u}_t}[\log(p(\mathbf{u}_t|\mathbf{s}_t)) - \log(r(\mathbf{u}_t))] \tag{37}$$

where ∇_ϕ, ∇_θ, ∇_ψ, and ∇_ω represent the gradients with respect to the critic network, actor network, IB module, and RIB module, respectively.

To enhance the robustness and efficiency of our improved actor-critic architecture, we integrate the TD3 algorithm to mitigate overestimation bias, improve stability, and enhance exploration in complex reinforcement learning environments. We employ dual critic networks Q_{ϕ_1} and Q_{ϕ_2} with parameters ϕ_1 and ϕ_2, respectively, to combat overestimation bias. The loss functions for these networks are defined in Equation (30), the target Q-value can be formulated as

$$y_t = r_t + \min(Q'_{\phi_1}(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}), Q'_{\phi_2}(\mathbf{s}_{t+1}, \mathbf{a}_{t+1})) \tag{38}$$

where Q'_{ϕ_1} and Q'_{ϕ_2} represent target networks, and $\mathbf{a}_{t+1}$ is generated by the target policy network with added noise:

$$\mathbf{a}_{t+1} = \pi'_\theta(\mathbf{s}_{t+1}) + N_t \tag{39}$$

where π'_θ denotes the target actor network and N_t is clipped Gaussian noise.

We implement delayed policy updates to optimize the actor network every d_{delay} iterations for stabilization. The objective function for the actor network is Equation (31).

To ensure stability, we employ soft updates for target network parameters:

$$\phi_j' \leftarrow \sigma\phi_j + (1-\sigma)\phi_j', \quad j = 1,2 \tag{40}$$

$$\psi_j' \leftarrow \sigma\psi_j + (1-\sigma)\psi_j', \quad j = 1,2 \tag{41}$$

$$\theta' \leftarrow \sigma\theta + (1-\sigma)\theta' \tag{42}$$

$$\omega' \leftarrow \sigma\omega + (1-\sigma)\omega' \tag{43}$$

where σ is the soft update coefficient.

The proposed method can outperform other self-supervised and contrastive learning methods in enhancing generalization capability; while traditional methods aim to improve generalization by increasing sample diversity. They often introduce redundant information, leading to greater model complexity. In contrast, TIB can simplify feature representations by retaining only task-relevant information, which simultaneously improves generalization performance and reduces generalization errors. This enables the model to perform more stably and robustly in complex, dynamic environments. Finally, the complete framework of the improved GT-TD3 will interact with the path tracking self-disturbance rejection control of the 4WISD vehicle. The diagram and the pseudocode are presented in Figure 5 and Algorithm 1, respectively.

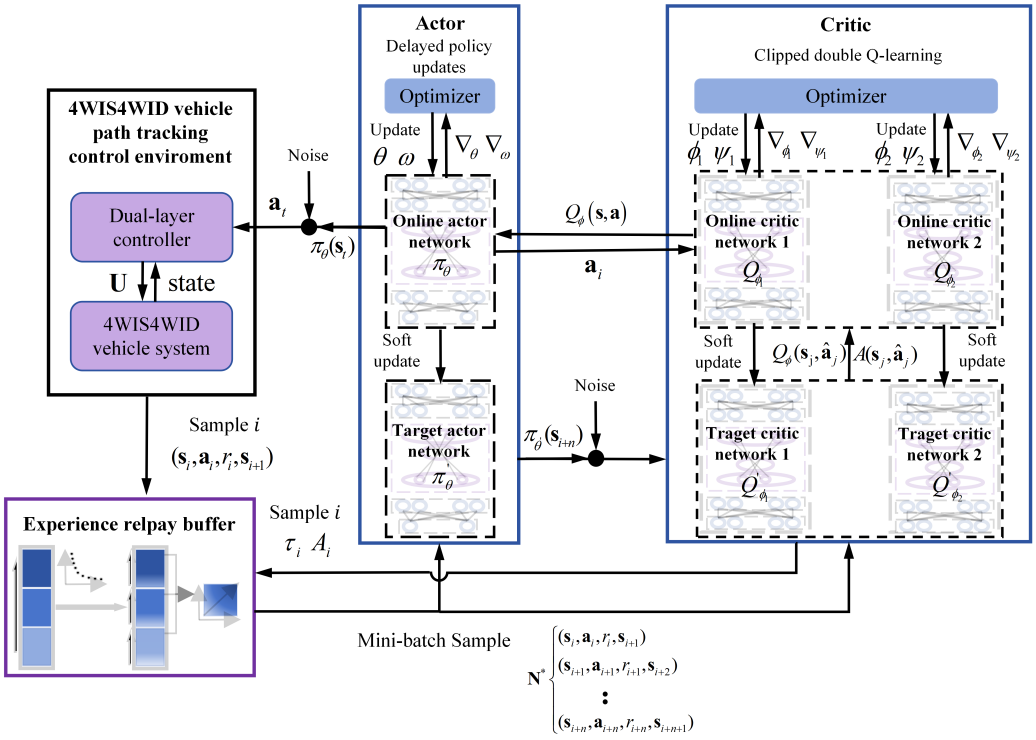


Figure 5. Diagram of the proposed path tracking control framework for 4WISD vehicles based on improved DRL.

The framework mainly consists of three components: the 4WISD vehicle path tracking control environment, the actor-critic network, and the experience replay buffer. The 4WISD vehicle path tracking control environment is composed of a dual-layer controller and a dynamics model of the 4WISD vehicle, which is designed to interact with the DRL to generate training data. The DRL actor consists of an on-line actor neural network and a target actor neural network, designed to generate appropriate actions for the 4WISD vehicle. The DRL critic includes two online critic neural networks and two target critic networks based on TIB, aiming to guide the actor network updates. The extended experience replay buffer, designed based on GER, is intended to store historical data for training the critic and actor.

Algorithm 1 Proposed GT-TD3

Initialize critic networks Q_{ϕ_1}, Q_{ϕ_2} and actor network π_θ with random parameters ϕ_1, ϕ_2, θ;
Initialize target networks $Q'_{\phi_1}, Q'_{\phi_2}, \pi'_\theta$ with parameters $\phi'_1 \leftarrow \phi_1, \phi'_2 \leftarrow \phi_2, \theta' \leftarrow \theta$;
Initialize experience replay buffer Ω and sampling proportion coefficients for three groups $S_{\alpha 0}$, $S_{\beta 0}$, $S_{\gamma 0}$;
for *episode* $= 1$ to M **do**
    Initialize a random process R_N for action exploration;
    Receive initial observation state $\mathbf{s}_1$;
    for $t = 1$ to E **do**
        Select action $\mathbf{a}_t = \pi_\theta(\mathbf{s}_t) + N_t$ according to the current policy and exploration noise;
        Execute action $\mathbf{a}_t$ in the environment;
        Observe reward $\mathbf{r}_t$, new state $\mathbf{s}_{t+1}$;
        Calculate TD error τ_t , advantage function A_t , priority P_t of sample;
        Store transition sample $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1}, P_t, \tau_t, A_t)$ in experience replay buffer;
        Divide the samples in the experience replay buffer into three groups based on priority P_t;
        Non-dominated sorting within groups based on τ_t and A_t:

$$F_k = \left\{ s_i \in S_k \mid \forall s_j \in S_k, \tau_j < \tau_i \text{ and } A_j < A_i \right\}$$

        Update sampling proportion coefficients for three groups $S_{\alpha t}$, $S_{\beta t}$, $S_{\gamma t}$:

$$\begin{cases} S_{\alpha t} = S_{\alpha 0}(1 - t/T)N_\alpha \\ S_{\beta t} = (S_{\beta 0}(1 - t/T) + S_{\alpha 0}(t/T))N_\beta \\ S_{\gamma t} = (1 - S_{\alpha 0} - S_{\beta 0})N_\gamma \end{cases}$$

        Update critic network by policy gradient:

$$L(\phi) = \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim E}\left[(Q(\mathbf{s}_t, \mathbf{a}_t) - y_t)^2 \right] + \xi \mathrm{KL}(p(\mathbf{z}_t | \mathbf{s}_t, \mathbf{a}_t), r(\mathbf{z}_t))$$

        if t mod $d_{\text{delay}} == 0$ **then**
            Update actor network by maximizing the loss:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\mathbf{s}_t \sim E}[\nabla_{\mathbf{a}} Q(\mathbf{s}_t, \mathbf{a}_t) \nabla_\theta \pi_\theta(\mathbf{s})] + \kappa \nabla_\theta \mathrm{KL}(p(\mathbf{u}_t | \mathbf{s}_t), r(\mathbf{u}_t))$$

            Update target networks:

$$\begin{aligned} \phi'_j &\leftarrow \sigma \phi_j + (1 - \sigma)\phi'_j, \quad j = 1, 2 \\ \psi'_j &\leftarrow \sigma \psi_j + (1 - \sigma)\psi'_j, \quad j = 1, 2 \\ \theta' &\leftarrow \sigma \theta + (1 - \sigma)\theta' \\ \omega' &\leftarrow \sigma \omega + (1 - \sigma)\omega' \end{aligned}$$

        end if
    end for
end for

5. Numerical Simulations

To validate the effectiveness of the proposed enhanced DRL algorithm and the DRL-based path tracking controller for 4WISD vehicles, we conducted extensive simulation experiments. These experiments were performed on a hardware platform equipped with an Intel Core i9-13900K processor and an NVIDIA GeForce RTX 4080 graphics card.

5.1. Convergence and Generalization Analyses

The simulation parameters for different DRL algorithms are presented in Table 1. The discount factor determines the extent of the future rewards relative to the immediate future. The soft update coefficient governs the convergence rate of the target network toward the current network. The exploration noise reflects the changes in exploratory behaviors, which is determined by the clipping noise. The policy network refresh frequency determines the update rate of the associated parameters. To make the proposed algorithms comparable, the hyperparameters are consistently set for all compared algorithms.

Table 1. Simulation parameters used in different DRL algorithms.

DRL Algorithm	AER-TD3	PER-TD3	GER-TD3
Hidden layer dimension	256	256	256
Batch size	256	256	256
Discount factor	0.99	0.99	0.99
Soft update coefficient	0.05	0.05	0.05
Policy noise	0.2	0.2	0.2
Noise clipping range	256	256	256
Policy update frequency	2	2	2
Priority exponent	×	0.6	0.6
Group proportion coefficients	×	×	0.2, 0.7, 0.1
Learning rate	1×10^{-4}	1×10^{-4}	1×10^{-4}

We establish a high-speed path tracking task for a 4WISD vehicle in the double-shift line condition. Random parameters are used to simulate the various conditions of vehicle velocities, curvatures, and external disturbances, such that the DRL training environment encompasses a wide range of path tracking scenarios, as specified in Table 2. And the simulation parameters for a C-class vehicle are listed in Table 3.

Table 2. Simulation parameters used in the DRL training environment.

DRL Training Environment	Parameters	Unit
Shift line longitudinal position	[40,180]	m
Shift line transition length	[25,75]	m
Longitudinal velocity	[15,20]	m/s
Longitudinal velocity variation range	[0,20]	m/s
Smooth disturbance amplitude	[−100,100]	N
Sudden disturbance amplitude	[−100,100]	N
Smooth disturbance duration	[5,10]	s

Table 3. System parameters of the vehicle.

Vehicle Parameter	Parameters	Unit
Vehicle mass	1477	kg
Vehicle yaw inertia	1536.7	$kg \cdot m^2$
Track width	1.675	m
Distance from CG to front axle	1.015	m
Distance from CG to rear axle	1.895	m
Wheel radius	0.325	m
Wheel mass	22	kg
Wheel moment of inertia	0.8	$kg \cdot m^2$

In the established training environment, experiments on each path tracking control task are repeated 10 times to demonstrate the efficacy of the DRL algorithms. During training, the agent is evaluated every 10,000 time steps, with one evaluation report sent every 100 evaluations. In the following figures, the solid lines represent the average results

over 10 trials, while the shaded regions indicate the variance of predictions around that mean across 10 experiments.

To show the effect of the methods proposed in Sections 4.2 and 4.3 on DRL performance, we conducted two convergence experiments. In the first experiment, reward curves obtained from three different experience replay mechanisms are compared in Figure 6, i.e., average experience replay (AER), prioritized experience replay (PER), and the proposed group intelligent experience replay (GER), with the remaining parameters unchanged. The reward curve of AER tends to converge in the early stages of iterations, but remains with a small reward value at the final stage, showing an inefficiency of convergence. Due to the unevenly distributed samples in the training environment, the low probability of occurrence leads to sample sparsity. These factors make it difficult for AER to effectively replay high-quality samples. The PER can prioritize samples based on the TD error, assigning higher sampling probabilities, which allows for the utilization of information-rich samples to yield more frequent updates for the network, thereby mitigating the impact of uneven and sparse distribution of samples. However, the PER-TD3 algorithm is trapped in local optima in the later training stages, resulting in almost constant reward curve variation. In comparison with simulation results from AER and PER, the reward curve from GER converges more stably, due to higher sample utilization efficiency and better balance between exploration and exploitation. The comparison results from the different DRL algorithms are presented in Table 4. The proposed GER improves convergence performance by 59% compared to AER and by 25% compared to PER.

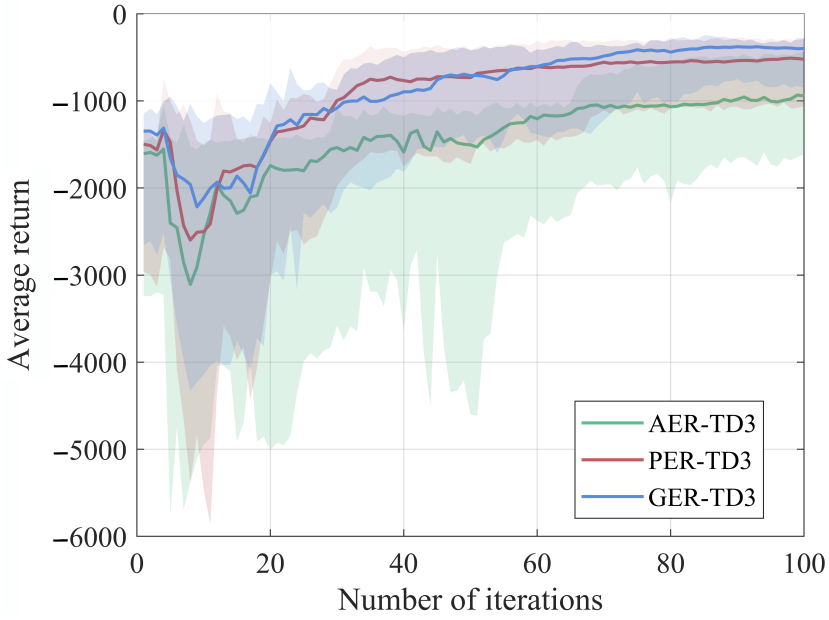


Figure 6. Comparisons of the average return obtained from AER-TD3, PER-TD3, and the proposed GER-TD3.

In the second experiment, we maintained a fixed experience replay buffer across deep deterministic policy gradient (DDPG), TD3, and the proposed TIB-TD3 to evaluate convergence. As shown in Figure 7, the TD3 demonstrates stronger convergence capabilities compared to DDPG, due to the application of double Q-networks, delayed policy updates, and target policy smoothing, which improve learning stability and performance in continuous control. The reward curve of proposed TIB-TD3 shows minimal decline in the early stages of training. So the TIB-TD3 converges more quickly and maintains a stable trend in the reward curve in the subsequent training process. This improvement can be attributed to the removal of irrelevant information with IB, enabling the agent to learn higher-quality information. Additionally, the agent enhances the quality and expressiveness of its policy

through the persistent application of the RIB, which is capable of preventing overfitting and reducing the influence of lacking unknown information.

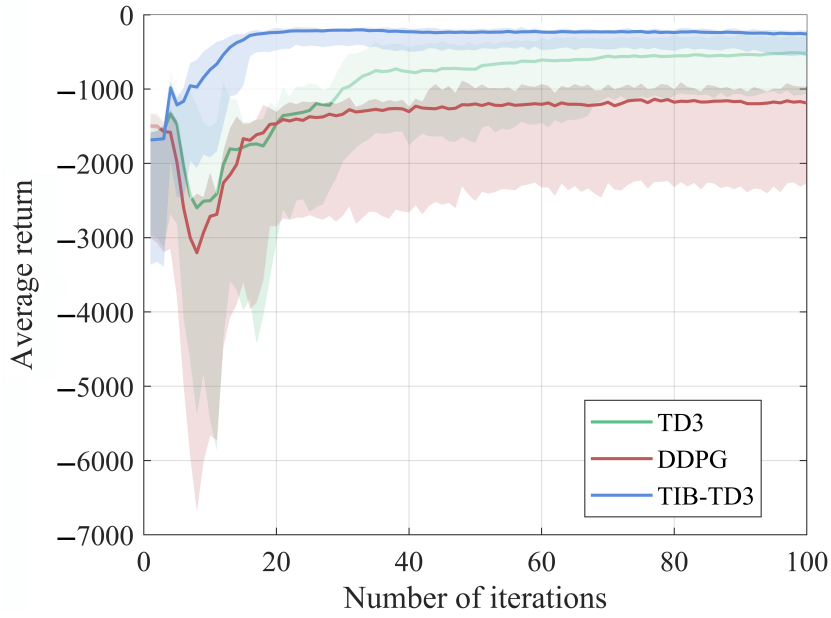


Figure 7. Comparisons of the average return obtained from DDPG, TD3, and the proposed TIB-TD3.

Table 4. Maximum average return obtained from different TD3-based DRL algorithms.

DRL Algorithm	AER-TD3 [63]	PER-TD3 [41]	GER-TD3
Maximum average return	−933.3041	−509.6428	−378.1214

The additive external disturbances, which consist of persistent and abrupt disturbances acting on each tire along the lateral and longitudinal directions, were varied from −100 N to 100 N to evaluate the generalization efficiency of different DRL algorithms. The percentage improvement represents the performance improvement of the DRL-based compound controller compared to the dual-layer controller in the path tracking control. As illustrated in Figure 8, the proposed TIB-TD3 exhibits enhanced optimization performance compared with DDPG and TD3, with the results presented in Table 5. The experimental results demonstrate that TIB improves the generalization efficiency and convergence performance of the original TD3 algorithm by 60% and 26%, respectively, by filtering irrelevant information and preserving critical information necessary for action generation. This improvement facilitates the application of DRL models for the path tracking control of a 4WISD vehicle with high uncertain disturbances and unmodeled dynamics.

Table 5. Summary of the simulation results obtained from different DRL algorithms.

DRL Algorithm	DDPG [64]	TD3 [59]	TIB-TD3
Maximum average return	−1136.5222	−509.6428	−202.4948
Mean optimization	34.3016	49.3389	66.4575

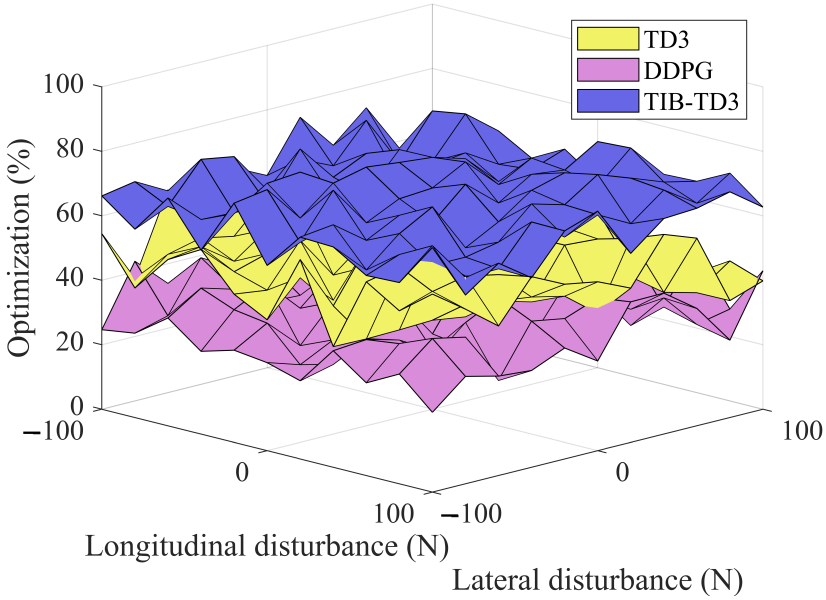


Figure 8. Comparisons of the generalization efficiency obtained from DDPG, TD3, and the proposed GER-TD3.

5.2. Performance Analysis of an Improved DRL-Based Path Tracking Controller for 4WISD Vehicles

To validate the performance of the proposed GT-TD3-based compound control framework (GTC) for path tracking control of 4WISD vehicles subjected to complex external disturbances, we compared it with the NMPC-SQP dual-layer controller (MLC) [16] and the TD3-based [59] compound control framework (TDC). In both the GTC and TDC, the DRL auxiliary controller is established in the training environment using the parameters specified in Section 5.1. The trained model is combined with the MLC method described in Section 3.2 to form the compound control framework.

The control performance of the aforementioned control methods is compared using the mean absolute error (MAE) and the maximum error metric (MAX), where MAE represents the average deviation from the desired path in this experiment, and MAX indicates the highest deviation observed at any point in the experiment, as shown in Figure 9. MAE reflects the overall steady-state control performance of the system, while the MAX indicates the transient performance. We can observe that the dual-layer controller has limited path tracking control capabilities for 4WISD vehicles subjected to complex external disturbances. The TDC can improve the path tracking performance due to the application of the TD3-based model-free auxiliary controller. The proposed GTC can adapt to abrupt changes in complex nonlinear vehicle dynamics with external disturbances, which improve the path tracking performance further. The proposed GTC reduces MAE error and MAX error by 68% and 63%, respectively, compared to MLC. The proposed GTC reduces MAE error and MAX error by 28% and 9%, respectively, compared to TDC.

To demonstrate the control performance of the three control methods, four groups of parameters for the double-shifted path and longitudinal velocity were established to simulate the vehicle's driving environment under high-speed and large-curvature conditions. Figure 10 presents a comparison of the actual lateral displacement, yaw angle, and longitudinal velocity of the controlled vehicle using the three control methods. Correspondingly, the control errors with respect to the ideal states for the three control methods are presented in Figure 11. From Figures 10 and 11, we can observe that when the tracking path is relatively stable, each control method can track the vehicle motion with high accuracy. However, when the tracking target undergoes significant changes, the dual-layer controller faces difficulties in mitigating external disturbances, resulting in substantial deviations in the vehicle state. In comparison, the TD3-based compound control framework can

further address the impact of external disturbances on path tracking control. The proposed GT-TD3 exhibits the highest capabilities of controlling the external disturbances and the complex tracking targets, keeping the tracking errors within a sufficiently low range. This is because the proposed auxiliary controller based on GT-TD3 is capable of conducting dynamic compensation for the control variables from the upper-layer controller, as shown in Figure 12. Through allocating the compensated upper-layer control variables to the tire dynamics model, we finally obtain eight end-effector control variables, that is, the steering angle and torque of each tire, as shown in Figure 13.

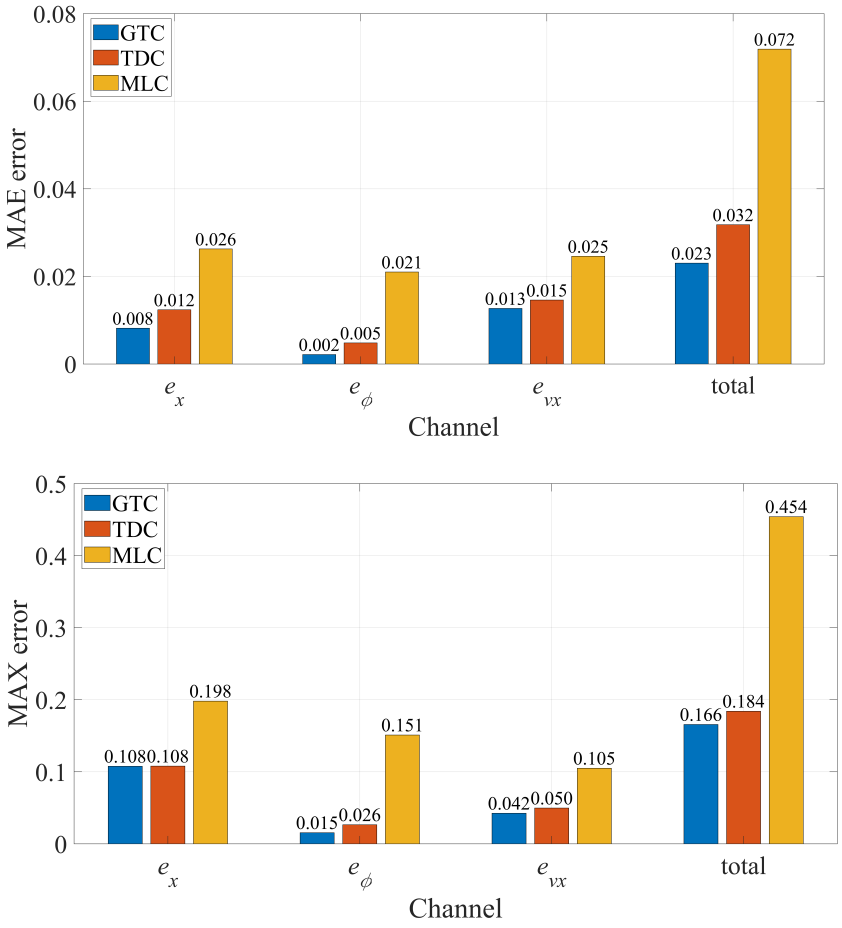


Figure 9. Comparisons of the MAE error and the MAX error using the MLC, TDC, and GTC.

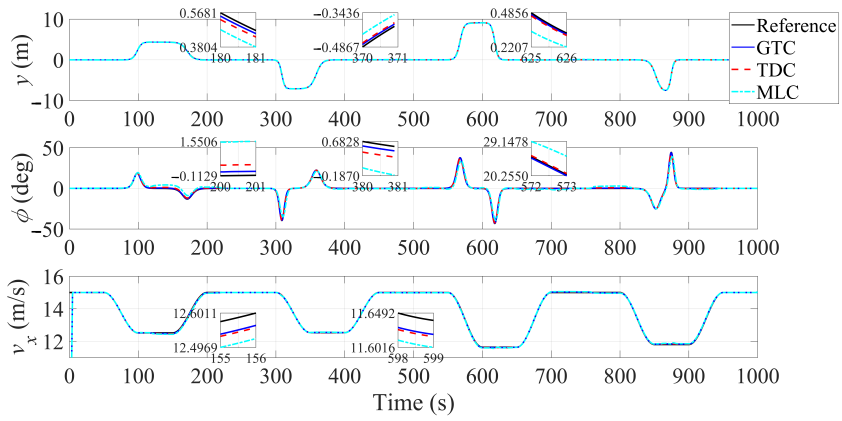


Figure 10. Comparisons of the path tracking performance from the MLC, TDC, and GTC.

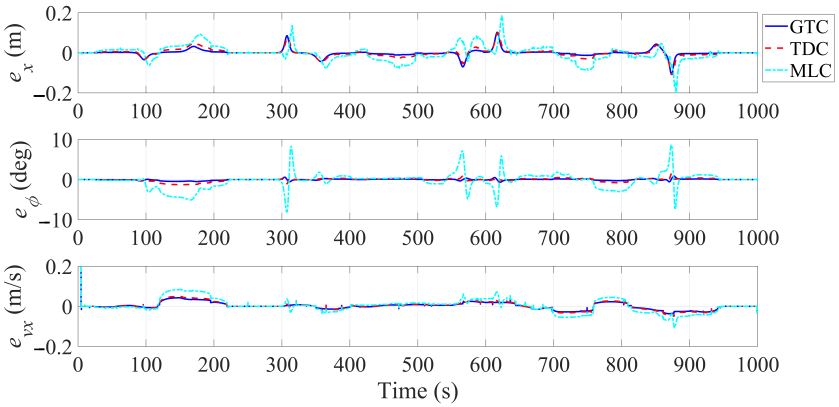


Figure 11. Comparisons of the path tracking errors from the MLC, TDC, and GTC.

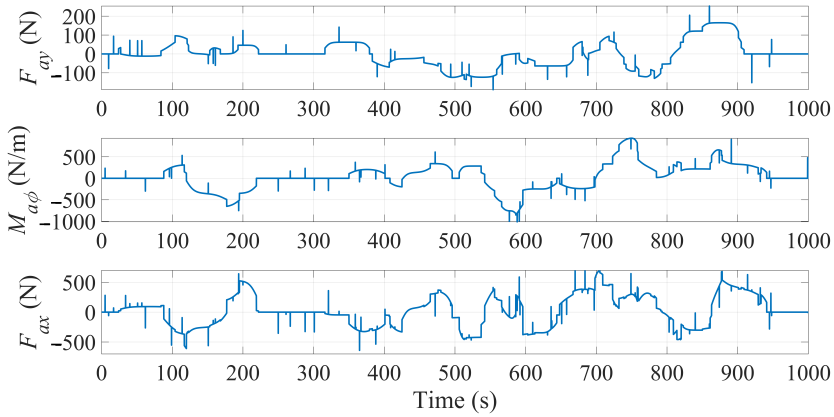


Figure 12. Function values of the auxiliary control variables obtained from GT-TD3.

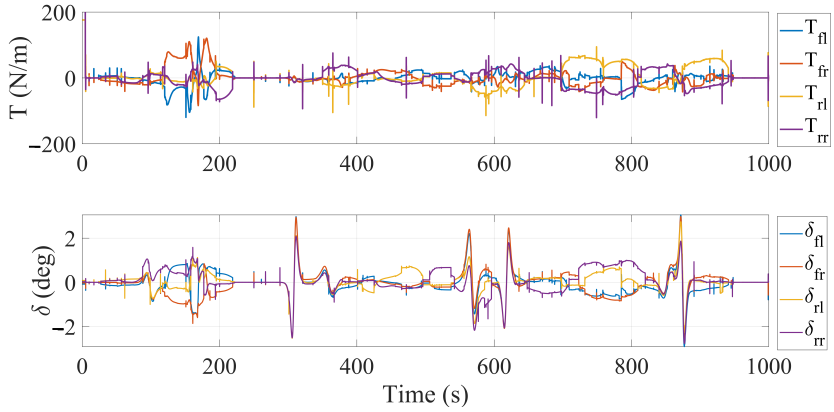


Figure 13. Function values of the 4WISD vehicle end-effector control variables.

The computation time of each controller is also an important factor for evaluating control performance. Figure 14 presents the computation time of the dual-layer controller, the compound control framework based on TD3, and the proposed compound control framework based on GT-TD3. The proposed GT-TD3 makes an improvement upon the original TD3 within the network architecture, rather than adding an independent structure. Therefore, this modification does not significantly increase the computation time of the controller. The experimental results demonstrate that the proposed compound control framework based on GT-TD3 achieves better control performance compared to conventional control methods, with little sacrifice of computation time.

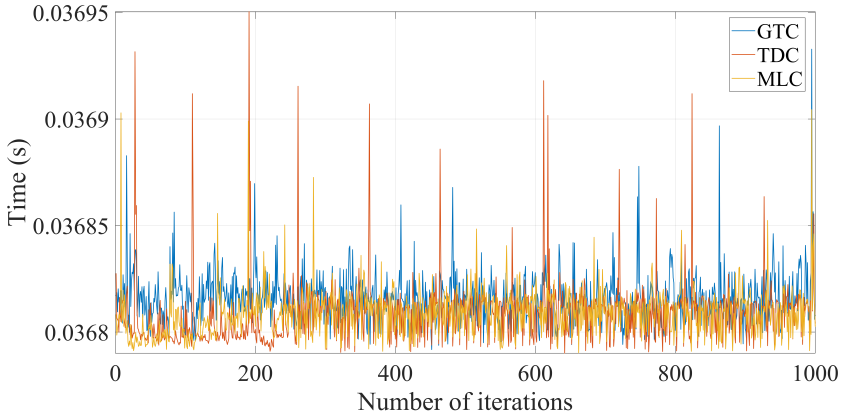


Figure 14. Comparisons of the computation time from the MLC, TDC, and GTC.

6. Conclusions

In this study, we investigate the path tracking control problem of 4WISD vehicles under complex external disturbances, which are characterized by high-dimensional nonlinearity, complex coupling, and high uncertainty. A path tracking control method based on deep reinforcement learning (DRL) is proposed. Firstly, a compound control framework consisting of an NMPC-SQP dual-layer controller and a DRL-based auxiliary controller is introduced to ensure stable and efficient path tracking control performance under complex external disturbances. A novel group intelligence collaborative experience replay (GER) mechanism is proposed to improve the sample efficiency and convergence of the DRL. Furthermore, an actor-critic architecture based on a two-stream information bottleneck (TIB) is presented to enhance DRL's ability in extracting high-dimensional nonlinear features and improving its generalization capability. Numerical simulations with extensive parameter settings are performed to validate the effectiveness of the proposed method. The experimental results demonstrate that the proposed GER improves convergence performance by 59% compared to AER and by 25% compared to PER. The proposed TIB can enhance the generalization efficiency and convergence performance of the original TD3 algorithm by 60% and 26%, respectively. Based on the proposed DRL algorithm, a well-trained DRL-based controller can essentially reduce the tracking error by 63%. Application of the proposed DRL can effectively address the issue of path tracking control of 4WISD vehicles under complex external disturbances. The proposed DRL-based compound control framework can significantly improve the stability and accuracy in path tracking control of 4WISD vehicles.

The compound control framework can be implemented in real-world environments to validate its robustness against uncertain disturbances and varying road conditions. Improvements can be made by incorporating advanced neural network designs or compound reinforcement learning strategies to reduce the complexity of the control framework and further improve its performance in complex 4WISD systems. Additionally, the scalability of the framework across different vehicle types and configurations can be further explored. Developing adaptive strategies that automatically adjust parameters in response to environmental changes is also recommended.

Author Contributions: Conceptualization, X.H. and X.C.; methodology, T.Z. and X.C.; software, T.Z.; validation, X.H., T.Z., X.C. and X.N.; formal analysis, T.Z. and X.C.; resources, X.H.; data curation, T.Z.; writing—original draft preparation, T.Z. and X.C.; writing—review and editing, X.H., T.Z., X.C. and X.N.; project administration, X.H.; funding acquisition, X.H. and X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China through Grant No. 52005443, and the Natural Science Foundation of Zhejiang Province through Grant No. LQ21E050016.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Stability Analysis of the Compound Control Framework

This section presents mathematical derivations and proofs to analyze the stability of the proposed composite control framework. The asymptotic stability conditions of the closed-loop system are derived, and the convergence of the states is estimated. These findings provide theoretical guidance for the design and parameter tuning of actual control systems.

The proposed compound control framework differs from the original dual-layer controller in two key aspects: the integration of a DRL auxiliary controller and the incorporation of external disturbances. Both modifications primarily impact the upper-layer controller, which is responsible for path tracking accuracy. Due to the last three force constraints in Equation (17), the impact of the lower-layer controller on the stability of the improved compound control framework is considered negligible. According to Equation (8), the proposed state-space equations for path following control can be transformed into

$$\dot{\mathbf{E}} = \dot{\mathbf{X}}^* - \dot{\mathbf{X}} = \dot{\mathbf{X}}^* - [\hat{\mathbf{A}}_n + \mathbf{B}(f_a(\mathbf{s}, \mathbf{a}) + f_{u_c}(\mathbf{e}, \mathbf{u_c}))] \tag{A1}$$

where $\mathbf{E} \in \mathbb{R}^{n\times 1}$ represents the error state vector between the reference state and the system state, $\mathbf{X}^* \in \mathbb{R}^{n\times 1}$ is the reference state vector, $\mathbf{X} \in \mathbb{R}^{n\times 1}$ is the system state vector, $\hat{\mathbf{A}}_n \in \mathbb{R}^{n\times 1}$ is the bounded external disturbance vector, $\mathbf{B} \in \mathbb{R}^{n\times n}$ is the inversion of vehicle mass matrix, $f_a(\mathbf{s}, \mathbf{a}) \in \mathbb{R}^{n\times 1}$ is the output of the DRL auxiliary controller, and $f_{u_c}(\mathbf{e}, \mathbf{u_c}) \in \mathbb{R}^{n\times 1}$ is the output of the upper-layer controller, respectively.

For the convenience of stability analysis, we make the following assumptions:

Assumption A1. *There exist positive constants γ_1, γ_2, and γ such that the reference state $\dot{\mathbf{X}}^*$ and the disturbance vector $\hat{\mathbf{A}}_n$ satisfy*

$$\|\dot{\mathbf{X}}^*\| \le \gamma_1, \quad \|\hat{\mathbf{A}}_n\| \le \gamma_2, \quad \gamma_1 + \gamma_2 = \gamma \tag{A2}$$

Assumption A2. *There exists a positive constant L such that the output of the DRL auxiliary controller $f_a(\mathbf{s}, \mathbf{a})$ satisfy*

$$\|f_a(\mathbf{s}, \mathbf{a})\| \le L\|\hat{\mathbf{A}}_n\|, \quad \forall \hat{\mathbf{A}}_n \in \mathbb{R}^{n\times 1} \tag{A3}$$

Assumption A3. *There exists a positive constant Q such that the output of the upper-layer controller $f_{u_c}(\mathbf{e}, \mathbf{u_c})$ satisfy*

$$\|f_{u_c}(\mathbf{e}, \mathbf{u_c})\| \le Q\|\mathbf{E}\|, \quad \forall \mathbf{E} \in \mathbb{R}^{n\times 1} \tag{A4}$$

These conditions for boundedness are typically met in practical systems and can be computed by estimating the disturbances, constraining the output of the DRL policy network, and imposing constraints on the NMPC solution.

Theorem A1. *Suppose Assumptions A1–A3 hold, then the closed-loop system described by Equation (A1) is globally asymptotically stable.*

Proof. We choose the following form for the Lyapunov function:

$$V(\mathbf{E}) = \frac{1}{2}\mathbf{E}^{\mathrm{T}}\mathbf{E} \tag{A5}$$

The Lyapunov function $V(\mathbf{E})$ satisfies the following conditions:

$$V(\mathbf{0}) = 0, \quad V(\mathbf{E}) > 0 \text{ for } \mathbf{E} \neq \mathbf{0}, \quad \lim_{\|\mathbf{E}\| \to \infty} V(\mathbf{E}) = \infty \tag{A6}$$

Taking the derivative of the Lyapunov function $V(\mathbf{E})$ with respect to time, we obtain

$$\begin{aligned} \dot{V}(\mathbf{E}) &= \frac{1}{2}\mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} + \frac{1}{2}\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{E} \\ &= \mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} \\ &= \mathbf{E}^{\mathrm{T}}\left[\dot{\mathbf{X}}^{*} - [\hat{\mathbf{A}}_n + \mathbf{B}(f_a(\mathbf{s},\mathbf{a}) + f_{u_c}(\mathbf{e},\mathbf{u_c}))]\right] \\ &= \mathbf{E}^{\mathrm{T}}(\dot{\mathbf{X}}^{*} - \hat{\mathbf{A}}_n) - \mathbf{E}^{\mathrm{T}}\mathbf{B}(f_a(\mathbf{s},\mathbf{a}) + f_{u_c}(\mathbf{e},\mathbf{u_c})) \end{aligned} \tag{A7}$$

From Assumption A1, we have

$$\mathbf{E}^{\mathrm{T}}(\dot{\mathbf{X}}^{*} - \hat{\mathbf{A}}_n) \leq \|\mathbf{E}\|(\|\dot{\mathbf{X}}^{*}\| + \|\hat{\mathbf{A}}_n\|) \leq \gamma\|\mathbf{E}\| \tag{A8}$$

From Assumptions A2 and A3, we have

$$\|f_a(\mathbf{s},\mathbf{a}) + f_{u_c}(\mathbf{e},\mathbf{u_c})\| \leq \|f_a(\mathbf{s},\mathbf{a})\| + \|f_{u_c}(\mathbf{e},\mathbf{u_c})\| \leq L\gamma_2 + Q\|\mathbf{E}\| \tag{A9}$$

Since $\mathbf{B}$ is a diagonal matrix that corresponds to the inverse of vehicle's mass and inertia parameters, it follows that $\mathbf{B}$ is positive definite. Let the minimum eigenvalue of $\mathbf{B}$ be $\lambda_{\mathbf{min}}(\mathbf{B}) > 0$, then

$$-\mathbf{E}^{\mathrm{T}}\mathbf{B}(f_a(\mathbf{s},\mathbf{a}) + f_{u_c}(\mathbf{e},\mathbf{u_c})) \leq -\lambda_{\mathbf{min}}(\mathbf{B})\|\mathbf{E}\|(L\gamma_2 + Q\|\mathbf{E}\|) \tag{A10}$$

Substituting Equations (A8) and (A10) into Equation (A7) leads to

$$\begin{aligned} \dot{V}(\mathbf{E}) &= \mathbf{E}^{\mathrm{T}}(\dot{\mathbf{X}}^{*} - \hat{\mathbf{A}}_n) - \mathbf{E}^{\mathrm{T}}\mathbf{B}(f_a(\mathbf{s},\mathbf{a}) + f_{u_c}(\mathbf{e},\mathbf{u_c})) \\ &\leq \gamma\|\mathbf{E}\| - \lambda_{\mathbf{min}}(\mathbf{B})\|\mathbf{E}\|(L\gamma_2 + Q\|\mathbf{E}\|) \\ &\leq [\gamma - \lambda_{\mathbf{min}}(\mathbf{B})L\gamma_2]\|\mathbf{E}\| - \lambda_{\mathbf{min}}(\mathbf{B})Q\|\mathbf{E}\|^2 \\ &\leq -\alpha_1\|\mathbf{E}\| - \alpha_2\|\mathbf{E}\|^2 \end{aligned} \tag{A11}$$

where $\alpha_1 = -[\gamma - \lambda_{\mathbf{min}}(\mathbf{B})L\gamma_2]$, $\alpha_2 = \lambda_{\mathbf{min}}(\mathbf{B})Q > 0$. According to Assumptions A1 and A2, there exists a constant $L > \gamma/[\lambda_{\mathbf{min}}(\mathbf{B})\gamma_2]$, such that $\alpha_1 > 0$.

According to Lyapunov stability theory, if there exists a continuously differentiable scalar function $V(\mathbf{E})$ satisfying the condition in Equation (A6), and $\dot{V}(\mathbf{E})$ is negative definite, that is, there exists positive constants α_1 and α_2 such that $\dot{V}(\mathbf{E}) \leq -\alpha\|\mathbf{E}\|^2$ for all $\mathbf{E} \neq \mathbf{0}$, then the equilibrium point of the system is globally asymptotically stable. □

References

1. Zhao, M.; Wang, L.; Ma, L.; Liu, C.; An, N. Development of a four wheel independent-driving and four wheel steering electric testing car. *China Mech. Eng.* **2009**, *20*, 319–322.
2. Dong, Z.R.; Ren, S.; He, P. Kinematics modeling and inverse kinematics simulation of a 4WID/4WIS electric vehicle based on multi-body dynamics. *Automot. Eng.* **2015**, *3*, 253–259.
3. Kumar, P.; Sandhan, T. Path-tracking control of the 4WIS4WID electric vehicle by direct inverse control using artificial neural network. In Proceedings of the 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 6–8 July 2023; pp. 1–8.
4. Zhang, Z.; Zhang, X.; Pan, H.; Salman, W.; Rasim, Y.; Liu, X.; Wang, C.; Yang, Y.; Li, X. A novel steering system for a space-saving 4WS4WD electric vehicle: Design, modeling, and road tests. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 114–127. [CrossRef]
5. Maoqi, L.; Ishak, M.I.; Heerwan, P.M. The effect of parallel steering of a four-wheel drive and four-wheel steer electric vehicle during spinning condition: A numerical simulation. *IOP Mater. Sci. Eng.* **2019**, *469*, 012084. [CrossRef]
6. Li, Y.; Cai, Y.; Sun, X.; Wang, H.; Jia, Y.; He, Y.; Chen, L.; Chao, Y. Trajectory tracking of four-wheel driving and steering autonomous vehicle under extreme obstacle avoidance condition. *Veh. Syst. Dyn.* **2024**, *62*, 601–622. [CrossRef]

7. Hang, P.; Han, Y.; Chen, X.; Zhang, B. Design of an active collision avoidance system for a 4WIS-4WID electric vehicle. *IFAC-PapersOnLine* **2018**, *51*, 771–777. [CrossRef]
8. Hang, P.; Chen, X. Towards autonomous driving: Review and perspectives on configuration and control of four-wheel independent drive/steering electric vehicles. *Actuators* **2021**, *10*, 184. [CrossRef]
9. Wang, Z.; Ding, X.; Zhang, L. Chassis coordinated control for full x-by-wire four-wheel-independent-drive electric vehicles. *IEEE Trans. Veh. Technol.* **2022**, *72*, 4394–4410. [CrossRef]
10. Long, W.; Zhang, Z. Research and design of steering control for four wheel driving mobile robots. *Control Eng. China* **2017**, *24*, 2387–2393.
11. Jin, L.; Gao, L.; Jiang, Y.; Chen, M.; Zheng, Y.; Li, K. Research on the control and coordination of four-wheel independent driving/steering electric vehicle. *Adv. Mech. Eng.* **2017**, *9*, 1687814017698877. [CrossRef]
12. Wang, C.; Heng, B.; Zhao, W. Yaw and lateral stability control for four-wheel-independent steering and four-wheel-independent driving electric vehicle. *Proc. Inst. Mech. Eng. Part J. Automob. Eng.* **2020**, *234*, 409–422. [CrossRef]
13. Yutao, L.; Tianyang, Z.; Xiaotong, X. Time-varying LQR control of four-wheel steer/drive vehicle based on genetic algorithm. *J. South China Univ. Technol. (Natural Sci. Ed.)* **2021**, *49*, 9.
14. Potluri, R.; Singh, A.K. Path-tracking control of an autonomous 4WS4WD electric vehicle using its natural feedback loops. *IEEE Trans. Control. Syst. Technol.* **2015**, *23*, 2053–2062. [CrossRef]
15. Lai, X.; Chen, X.B.; Wu, X.J.; Liang, D. A study on control system for four-wheels independent driving and steering electric vehicle. *Appl. Mech. Mater.* **2015**, *701*, 807–811. [CrossRef]
16. Tan, Q.; Dai, P.; Zhang, Z.; Katupitiya, J. MPC and PSO based control methodology for path tracking of 4WS4WD vehicles. *Appl. Sci.* **2018**, *8*, 1000. [CrossRef]
17. Zhang, X.; Zhu, X. Autonomous path tracking control of intelligent electric vehicles based on lane detection and optimal preview method. *Expert Syst. Appl.* **2019**, *121*, 38–48. [CrossRef]
18. Akermi, K.; Chouraqui, S.; Boudaa, B. Novel SMC control design for path following of autonomous vehicles with uncertainties and mismatched disturbances. *Int. J. Dyn. Control.* **2020**, *8*, 254–268. [CrossRef]
19. Jeong, Y.; Yim, S. Model predictive control-based integrated path tracking and velocity control for autonomous vehicle with four-wheel independent steering and driving. *Electronics* **2021**, *10*, 2812. [CrossRef]
20. Barari, A.; Afshari, S.S.; Liang, X. Coordinated control for path-following of an autonomous four in-wheel motor drive electric vehicle. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2022**, *236*, 6335–6346. [CrossRef]
21. Rui, L.; Duan, J. A path tracking algorithm of intelligent vehicle by preview strategy. In Proceedings of the 32nd Chinese Control Conference, Xi'an, China, 26–28 July 2022; pp. 26–28.
22. Li, Y.; Jiang, Y.; Wang, L.; Cao, J.; Zhang, G. Intelligent PID guidance control for AUV path tracking. *J. Cent. South Univ.* **2015**, *22*, 3440–3449. [CrossRef]
23. Zhang, P.; Zhang, J.; Kan, J. A research on manipulator-path tracking based ondeep reinforcement learning. *Appl. Sci.* **2023**, *13*, 7867. [CrossRef]
24. Li, Z.; Yuan, S.; Yin, X.; Li, X.; Tang, S. Research into autonomous vehicles following and obstacle avoidance based on deep reinforcement learning method under map constraints. *Sensors* **2023**, *23*, 844. [CrossRef] [PubMed]
25. Lu, Y.; Wu, C.; Yao, W.; Sun, G.; Liu, J.; Wu, L. Deep reinforcement learning control of fully-constrained cable-driven parallel robots. *IEEE Trans. Ind. Electron.* **2022**, *70*, 7194–7204. [CrossRef]
26. Chen, H.; Zhang, Y.; Bhatti, U.A.; Huang, M. Safe decision controller for autonomous driving based on deep reinforcement learning in nondeterministic environment. *Sensors* **2023**, *23*, 1198. [CrossRef]
27. Mirmozaffari, M.; Yazdani, M.; Boskabadi, A.; Ahady Dolatsara H.; Kabirifar K.; Amiri Golilarz N. A novel machine learning approach combined with optimization models for eco-efficiency evaluation. *Appl. Sci.* **2020**, *10*, 5210. [CrossRef]
28. Osedo, A.; Wada, D.; Hisada, S. Uniaxial attitude control of uncrewed aerial vehicle with thrust vectoring under model variations by deep reinforcement learning and domain randomization. *ROBOMECH J.* **2023**, *10*, 20. [CrossRef]
29. Huang, S.; Wang, T.; Tang, Y.; Hu, Y.; Xin, G.; Zhou, D. Distributed and scalable cooperative formation of unmanned ground vehicles using deep reinforcement learning. *Aerospace* **2023**, *10*, 96. [CrossRef]
30. Abbas, A.N.; Chasparis, G.C.; Kelleher, J.D. Specialized deep residual policy safe reinforcement learning-based controller for complex and continuous state-action spaces. *arXiv* **2023**, arXiv:2310.14788.
31. Liu, T.; Yang, Y.; Xiao, W.; Tang, X.; Yin, M. A Comparative Analysis of Deep Reinforcement Learning-Enabled Freeway Decision-Making for Automated Vehicles. *IEEE Access* **2024**, *12*, 24090–24103 [CrossRef]
32. Lin, Y.; McPhee, J.; Azad, N.L. Comparison of deep reinforcement learning and model predictive control for adaptive cruise control. *IEEE Trans. Intell. Veh.* **2020**, *6*, 221–231. [CrossRef]
33. Chen, T.C.; Sung, Y.C.; Hsu, C.W.; Liu, D.R.; Chen, S.J. Path following and obstacle avoidance of tracked vehicle via deep reinforcement learning with model predictive control as reference. In Proceedings of the Multimodal Sensing and Artificial Intelligence: Technologies and Applications III, Munich, Germany, 26–30 June 2023; pp. 91–96.
34. Selvaraj, D.C.; Hegde, S.; Amati, N.; Deflorio, F.; Chiasserini, C.F. An ML-aided reinforcement learning approach for challenging vehicle maneuvers. *IEEE Trans. Intell. Veh.* **2022**, *8*, 1686–1698. [CrossRef]
35. Li, D.; Zhao, D.; Zhang, Q.; Chen, Y. Reinforcement learning and deep learning based lateral control for autonomous driving [application notes]. *IEEE Comput. Intell. Mag.* **2019**, *14*, 83–98. [CrossRef]

36. Peng, Y.; Tan, G.; Si, H.; Li, J. DRL-GAT-SA: Deep reinforcement learning for autonomous driving planning based on graph attention networks and simplex architecture. *J. Syst. Archit.* **2022**, *126*, 102505. [CrossRef]
37. Li, J.; Wu, X.; Fan, J.; Liu, Y.; Xu, M. Overcoming driving challenges in complex urban traffic: A multi-objective eco-driving strategy via safety model based reinforcement learning. *Energy* **2023**, *284*, 128517. [CrossRef]
38. EL Sallab, A.; Abdou, M.; Perot, E.; Yogamani, S. Deep reinforcement learning framework for autonomous driving. *arXiv* **2017**, arXiv:1704.02532. [CrossRef]
39. Wei, Q.; Ma, H.; Chen, C.; Dong, D. Deep reinforcement learning with quantum-inspired experience replay. *IEEE Trans. Cybern.* **2021**, *52*, 9326–9338. [CrossRef]
40. Li, Y.; Aghvami, A.H.; Dong, D. Path planning for cellular-connected UAV: A DRL solution with quantum-inspired experience replay. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 7897–7912. [CrossRef]
41. Zhu, P.; Dai, W.; Yao, W.; Ma, J.; Zeng, Z.; Lu, H. Multi-robot flocking control based on deep reinforcement learning. *IEEE Access* **2020**, *8*, 150397–150406. [CrossRef]
42. Na, S.; Niu, H.; Lennox, B.; Arvin, F. Bio-inspired collision avoidance in swarm systems via deep reinforcement learning. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2511–2526. [CrossRef]
43. Ye, X.; Yu, Y.; Fu, L. Deep reinforcement learning based link adaptation technique for LTE/NR systems. *IEEE Trans. Veh. Technol.* **2023**, *72*, 7364–7379. [CrossRef]
44. Ma, J.; Ning, D.; Zhang, C.; Liu, S. Fresher experience plays a more important role in prioritized experience replay. *Appl. Sci.* **2022**, *12*, 12489. [CrossRef]
45. Wang, X.; Luo, Y.; Qin, B.; Guo, L. Power allocation strategy for urban rail HESS based on deep reinforcement learning sequential decision optimization. *IEEE Trans. Transp. Electrif.* **2022**, *9*, 2693–2710. [CrossRef]
46. Osei, R.S.; Lopez, D. Experience replay optimization via ESMM for stable deep reinforcement learning. *Int. J. Adv. Comput. Sci. Appl.* **2024**, *15*, 1. [CrossRef]
47. Liu, Y.; Wang, H.; Wu, T.; Lun, Y.; Fan, J.; Wu, J. Attitude control for hypersonic reentry vehicles: An efficient deep reinforcement learning method. *Appl. Soft Comput.* **2022**, *123*, 108865. [CrossRef]
48. Xiang, G.; Dian, S.; Du, S.; Lv, Z. Variational information bottleneck regularized deep reinforcement learning for efficient robotic skill adaptation. *Sensors* **2023**, *23*, 762. [CrossRef]
49. Zou, Q.; Suzuki, E. Compact goal representation learning via information bottleneck in goal-conditioned reinforcement learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, 1–14. [CrossRef]
50. Schwarzer, M.; Anand, A.; Goel, R.; Hjelm, R.D.; Courville, A.; Bachman, P. Data-efficient reinforcement learning with self-predictive representations. *arXiv* **2020**, arXiv:2007.05929.
51. Zhang, A.; McAllister, R.; Calandra, R.; Gal, Y.; Levine, S. Learning invariant representations for reinforcement learning without reconstruction. *arXiv* **2020**, arXiv:2006.10742.
52. Laskin, M.; Srinivas, A.; Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. In Proceedings of the 37th International Conference on Machine Learning, Virtual, 13–18 July 2020; pp. 5639–5650.
53. Stooke, A.; Lee, K.; Abbeel, P.; Laskin, M. Decoupling representation learning from reinforcement learning. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 9870–9879.
54. Wei, W.; Fu, L.; Gu, H.; Zhang, Y.; Zou, T.; Wang, C.; Wang, N. GRL-PS: Graph embedding-based DRL approach for adaptive path selection. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 2639–2651. [CrossRef]
55. Qian, Z.; You, M.; Zhou, H.; He, B. Weakly supervised disentangled representation for goal-conditioned reinforcement learning. *IEEE Robot. Autom. Lett.* **2022**, *7*, 2202–2209. [CrossRef]
56. Yarats, D.; Zhang, A.; Kostrikov, I.; Amos, B.; Pineau, J.; Fergus, R. Improving sample efficiency in model-free reinforcement learning from images. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; pp. 10674–10681.
57. Dai, P.; Katupitiya, J. Force control for path following of a 4WS4WD vehicle by the integration of PSO and SMC. *Veh. Syst. Dyn.* **2018**, *56*, 1682–1716. [CrossRef]
58. Besselink, I.J.M.; Schmeitz, A.J.C.; Pacejka, H.B. An improved Magic Formula/Swift tyre model that can handle inflation pressure changes. *Veh. Syst. Dyn.* **2010**, *48*, 337–352. [CrossRef]
59. Wang, X.; Zhang, J.; Hou, D.; Cheng, Y. Autonomous driving based on approximate safe action. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 14320–14328. [CrossRef]
60. Yan, S.; Liu, W.; Li, X.; Yang, P.; Wu, F.; Yan, Z. Comparative study and improvement analysis of sparrow search algorithm. *Wirel. Commun. Mob. Comput.* **2022**, *1*, 4882521. [CrossRef]
61. Tian, Y.; Wang, H.; Zhang, X.; Jin, Y. Effectiveness and efficiency of non-dominated sorting for evolutionary multi-and many-objective optimization. *Complex Intell. Syst.* **2017**, *3*, 247–263. [CrossRef]
62. Wu, A.; Deng, C. TIB: Detecting unknown objects via two-stream information bottleneck. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *46*, 611–625. [CrossRef]
63. Feng, X. Consistent experience replay in high-dimensional continuous control with decayed hindsights. *Machines* **2022**, *10*, 856. [CrossRef]
64. Dankwa, S.; Zheng, W. Twin-delayed ddpg: A deep reinforcement learning technique to model a continuous movement of an intelligent robot agent. In Proceedings of the 3rd International Conference on Vision, Image and Signal Processing, Vancouver, BC, Canada, 26–28 August 2019; pp. 1–5.