Commun Transp Res

Review

Recent Advances of Reinforcement Learning Algorithms for Autonomous Driving System

Bin Shuai[1,†], Min Hua[2,†], Letian Tao[1], Zhilong Zheng[1], Yujie Yang[1], Yang Guan[1], Lei He[1], Jingliang Duan[1], Jiaxin Gao[1], Shuo Feng[3], Xianyuan Zhan[4], Jincheng Yu[5], Yang Yu[6], Haifeng Zhang[7], Shengbo Eben Li[1,✉]

Cite this article: https://doi.org/10.26599/COMMTR.2026.9640053

ABSTRACT: Reinforcement learning (RL) is being studied for autonomous driving (AD), but its value depends on the role it plays in a task, the action interface, the evaluation protocol, and the evidence from deployment. This survey examines RL-based AD in modular and end-to-end pipelines and relates reported methods to task formulation and deployment evidence. It maps safe RL, offline RL, model-based RL, and PPO/GRPO-style fine-tuning to maneuver selection, continuous control, world modeling, and VLM/VLA-based driving. It also reviews simulators, datasets, RL platforms, and VLA benchmarks, with attention to reward design, observation space, traffic complexity, and open-loop versus closed-loop evaluation. The survey then examines deployment barriers, including safety, Sim2Real generalization, data efficiency, computation, embodied alignment, and evaluation readiness. RL and VLM/VLA-based methods have shown promise, but current evidence is insufficient to support reliable real-world deployment: many reported results come from restricted scenarios and depend on engineered rewards or simulator assumptions.

KEYWORDS: Autonomous driving, decision-making, vehicle control, reinforcement learning.

1 Introduction

Recent studies have examined autonomous driving as a way to improve traffic efficiency, road safety, and energy use (Yurtsever et al, 2020; Hua, et al, 2025). Traditional autonomous driving (AD) systems use modular architectures in which perception, planning, and control operate as separate components based on predefined rules and heuristics. This design increases system complexity, allows errors to propagate between modules, and can limit adaptation to changing traffic conditions.

Advances in artificial intelligence (AI) have increased the use of data-driven and end-to-end learning methods. End-to-end models map sensory inputs directly to driving actions through a unified model, reducing the number of separately designed interfaces. Imitation learning (IL) and reinforcement learning (RL) are two common approaches. IL learns driving behavior from expert demonstrations and can reproduce the styles represented in those data. Its performance depends on the coverage and quality of the demonstrations, and it may generalize poorly to out-of-distribution situations. It also does not provide formal safety guarantees on its own. RL learns a policy through interactions with an environment and can optimize behavior using task-specific rewards (Liu et al, 2023; Chen et al, 2025; Hua, et al, 2023).

While end-to-end AD frameworks have garnered significant attention for their potential to streamline complex system integration, they intensify fundamental challenges in safety guarantees, interpretable decision-making, and real-world generalization—particularly when deploying RL in safety-critical scenarios. Recent surveys have examined RL applications in AD. Wu et al, review RL-based methods for behavior decision-making, focusing on how RL can help in selecting optimal driving strategies (Wu et al, 2024). Zhang et al, provide a comprehensive overview of Multi-agent Reinforcement Learning (MARL) in AD, particularly in scenarios involving interactions with other vehicles and agents (Hua, et al, 2023). Zhao et al, examine the role of RL across different modules of modular AD systems, such as perception, planning, and control (Zhao et al, 2024). While these studies provide valuable insights, they primarily focus on RL within traditional modular architectures and overlook its increasing role in end-to-end AD systems.

Existing surveys still leave several gaps for understanding how RL should be used in autonomous driving. Many studies are compared mainly by algorithm name, while the task hierarchy, action-space design, reward formulation, simulator setting, and open-loop or closed-loop evaluation protocol are not discussed together. In addition, emerging directions such as safe RL, offline RL, imitation learning, model-based RL, and RL fine-tuning for foundation-model or vision

† Bin Shuai and Min Hua contributed equally to this work.

[1] State Key Laboratory of Intelligent Green Vehicle and Mobility, Tsinghua University, Beijing 100084, China. [2] School of Engineering, University of Birmingham, Birmingham B15 2TT, UK. [3] Department of Automation, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China. [4] Institute for AI Industry Research (AIR), Tsinghua University, Beijing 100084, China. [5] Department of Electronic Engineering, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China. [6] National Key Laboratory for Novel Software Technology Nanjing University, Nanjing 210023, China. [7] Complex System Cognition and Decision-Making Laboratory, University of Chinese Academy of Sciences, Beijing 101408, China.

✉ Corresponding author. E-mail: lishbo@tsinghua.edu.cn

Received: March 13, 2026; Revised: June 13, 2026; Accepted: September 9, 2026

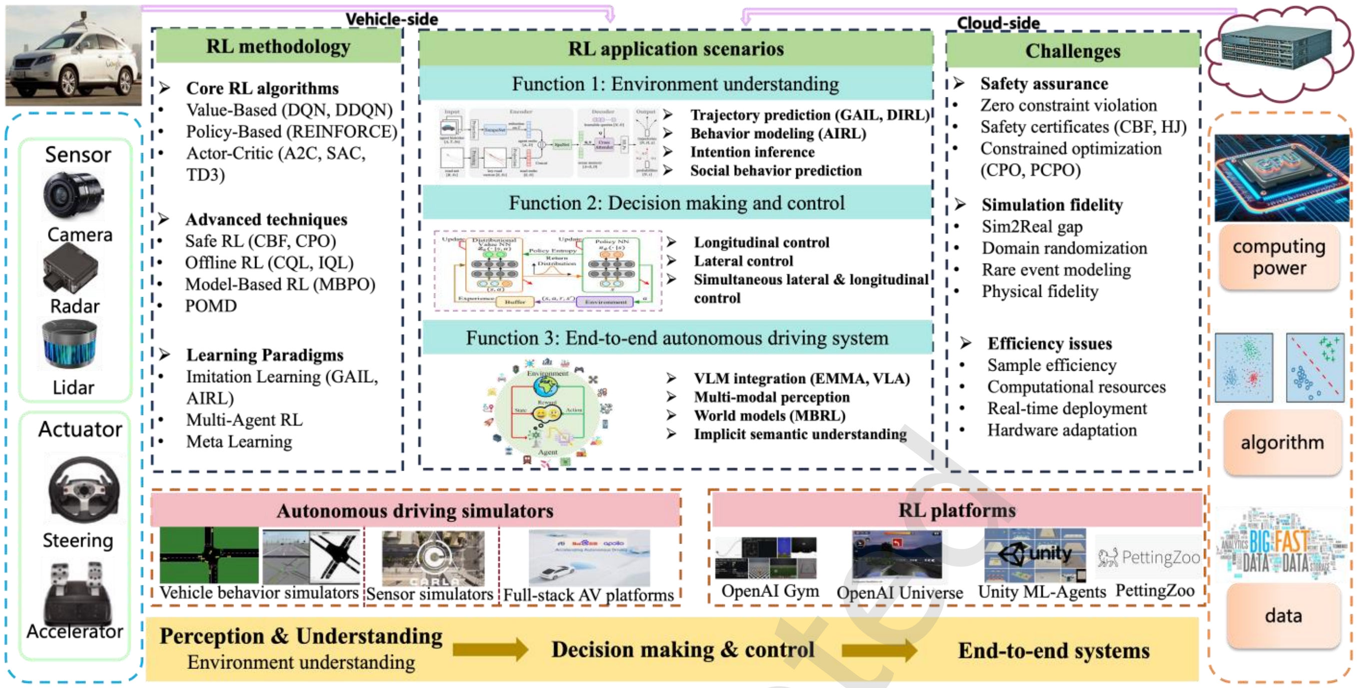

Fig.1 An overview of RL methodologies, applications, and challenges in autonomous driving

language action (VLA) driving systems are often mixed together, although they address different stages of data use, policy optimization, and deployment readiness. To address these gaps, this paper organizes the literature around the relationship between RL methodology and AD system requirements. The main contributions are summarized as follows:

1) It reviews safe RL, offline RL, imitation learning, model-based RL, and RL fine-tuning for foundation-model and VLA-based driving, with attention to their roles, limitations, and common uses.
2) It covers RL applications in environment understanding, decision-making and planning, vehicle control, and end-to-end driving. Behavior modeling and policy initialization are treated separately from closed-loop policy optimization.
3) It compares simulators, datasets, metrics, and evaluation protocols, and examines how scenario difficulty, reward design, observability, and action interfaces affect reported results.
4) It examines deployment constraints in safety assurance, Sim2Real transfer, sample efficiency, and computation, and considers which RL directions have evidence relevant to deployment.

The rest of the paper is organized into six sections. Section II introduces the foundations of reinforcement learning. Section III describes the simulation toolchains and training platforms. Section IV reviews RL applications in environment understanding, decision-making and control, and integrated end-to-end systems. Section V discusses deployment challenges, including safety assurance, simulation fidelity, and computational efficiency, and considers related research directions. Section VI concludes the paper and identifies directions for future work.

2 Reinforcement learning methodology

This section covers the RL background needed to read autonomous-driving studies. It focuses on how algorithm families relate to AD task structure, action interfaces, and safety requirements.

2.1. Principle of reinforcement learning

RL can be formulated as a Markov decision process (MDP). At each step, an agent observes the current state, selects an action, receives a reward, and moves to the next state according to the transition dynamics (Li, 2023). In autonomous driving, the main modeling choices are whether the state is fully or partially observed, whether the action space is discrete or continuous, and how the reward represents safety, comfort, efficiency, and route progress.

An MDP is denoted as (S, A, P, R, γ), w, where S is the state space, A is the action space, $P(s'|s,a)$ is the transition probability, $R(s,a)$ is the reward function, and $\gamma \in [0,1]$ is the discount factor. This compact notation is sufficient for the comparative discussions in the remainder of the survey.

The objective in RL is to identify an optimal policy $\pi: S \to A$ that maximizes the expected return, defined as the cumulative discounted sum of rewards:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{1}$$

where r_{t+k+1} denotes the reward received at time step $t+k+1$.

2.2. Classification of reinforcement learning algorithms

RL algorithms for autonomous driving can be distinguished by where policy improvement is performed, through action-value estimation, direct policy optimization, or a coupled actor–critic update.

2.2.1 Value-based algorithms

Value-based RL learns an action-value function and derives the policy by selecting the action with the highest estimated return. A representative update is Q-learning (Watkins 1989):

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left(r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right) \tag{2}$$

This family is most suitable for discrete action spaces. In autonomous driving, it is therefore more naturally used for high-level decisions such as lane selection, gap acceptance, and maneuver triggering. Deep Q-Network (DQN) and its variants extend this idea to larger state spaces through neural approximation.

2.2.2 Policy-based algorithms

Policy-based RL directly optimizes a parameterized policy $\pi_\theta(a \mid s)$, which is attractive when a stochastic or continuous policy is desired. Its core update follows the policy-gradient theorem (Williams 1992):

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[\nabla_\theta \log\pi_\theta(a_t \mid s_t) Q^\pi(s_t, a_t)] \tag{3}$$

and in practice the advantage form is more common for variance reduction.

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[\nabla_\theta \log\pi_\theta(a_t \mid s_t) A(s_t, a_t)] \tag{4}$$

REINFORCE is the classical example, but pure policy-gradient methods are less common in AD than actor–critic variants because of their higher variance.

2.2.3 Actor-critic algorithms

Actor–critic algorithms combine an explicit policy with a learned value estimator. The actor determines the action distribution, while the critic evaluates the current policy and supplies the learning signal used in the policy update. This separation supports both discrete and continuous action interfaces rather than defining actor–critic methods solely as continuous-control algorithms.

Representative methods include Proximal Policy Optimization (PPO), Trust Region Policy Optimization (TRPO), Deep Deterministic Policy Gradient (DDPG), Twin Delayed DDPG (TD3), Soft Actor-Critic (SAC), and Distributional Soft Actor-Critic (DSAC). PPO and TRPO provide on-policy updates for stochastic actors, whereas DDPG, TD3, SAC, and DSAC are commonly formulated as off-policy methods for continuous actions. In autonomous driving, the latter group maps naturally to steering, throttle, and braking commands, while PPO can be used with either maneuver-level or continuous-control policies. The practical distinction is therefore not only between discrete and continuous actions; it also concerns the policy representation, data reuse, and stability requirements of the driving task.

2.3. Advanced topics and foundation-model fine-tuning

This section is a broad catalogue of RL variants to autonomous driving: safe RL for constrained decisions, offline RL for logged driving data, imitation learning for behavioral priors, model-based RL for sample-efficient rollouts, and RL fine-tuning for foundation-model or VLA driving systems.

2.3.1 Safe reinforcement learning

Safe RL is essential for AD because reward maximization alone cannot guarantee collision avoidance, rule compliance, or zero constraint violation. Compared with rule-based safety checks, safe RL formulates safety as long-horizon constraints in policy optimization, for example through constrained MDPs, control barrier functions, feasible-region estimation, or external safety filters (Yang et al, 2024, 2023; Ma et al, 2021). These methods are useful for lane changing, following, intersection negotiation, and other safety-critical tasks where short-term distance checks may miss long-term risk.

However, safe RL remains difficult to deploy at scale. Feasible regions are hard to estimate under stochastic traffic behavior and high-dimensional sensor observations, while handcrafted barriers or filters can make the policy overly conservative. Recent methods such as feasible region iteration, dynamic confidence-aware RL, and approximate safe action filters improve constraint handling (Yang et al, 2024; Cao et al, 2023; Wang et al, 2023), but their guarantees still depend on model assumptions, scenario coverage, and filter design. Therefore, safe RL should be viewed as a key safety-enhancing component rather than a complete deployment guarantee for real-world AD.

2.3.2 Offline reinforcement learning

Offline RL learns policies from pre-collected driving datasets instead of online trial-and-error, making it attractive for AD where unsafe exploration is unacceptable (Levine et al, 2020). Representative methods such as Conservative Q-Learning (CQL), Implicit Q-Learning (IQL), and Conservative Offline Model-Based Policy Optimization (COMBO) use conservative value estimation or distribution-shift correction to avoid choosing actions that are poorly supported by logged data (Kumar et al, 2020; Kostrikov et al, 2022; Yu et al, 2021). In AD, this paradigm is useful for high-level planning, behavior policy learning, and data-replay benchmarks built from human driving or previous autonomous-system logs (Li et al, 2024; Lee, et al, 2024; Lin et al, 2024).

The main limitation is that offline datasets cannot cover all deployment states, especially long-tail and interactive scenarios. As a result, offline RL policies may be overly conservative or unreliable outside the data distribution (An et al, 2021). Benchmarks such as AD4RL help standardize evaluation (Lee, Eom, and Kwon 2024), but real deployment still requires uncertainty estimation, closed-loop validation, and careful dataset coverage analysis. Thus, offline RL reduces risky exploration but does not by itself solve generalization or safety assurance.

2.3.3 Imitation learning

IL learns driving behavior from expert demonstrations rather than from manually designed rewards (Hussein et al, 2017). In AD, it is useful when reward design is difficult, such as urban driving, interaction modeling, and VLA policy initialization. The two main forms are behavior cloning (BC), which directly maps observations to expert actions, and inverse RL (IRL), which infers a reward function that explains expert behavior (Pomerleau 1991).

BC is simple and scalable, but it suffers from compounding errors when the learned policy visits states that are rare in the demonstration data (Ross and Bagnell 2010). Methods such as Dataset Aggregation (DAgger) reduce this issue by collecting corrective expert labels during policy rollout (Ross et al., 2011). IRL and adversarial imitation methods such as Generative Adversarial Imitation Learning (GAIL) attempt to recover rewards or match expert state-action distributions (Ho and Ermon 2016), but they can be sensitive to dataset quality, simulator dynamics, and discriminator design. Therefore, in AD surveys, IL should be viewed mainly as a source of behavioral priors and action alignment, often combined with RL for closed-loop refinement rather than replacing policy optimization entirely.

2.3.4 Model-based reinforcement learning

Model-based RL (MBRL) improves sample efficiency by using a known or learned environment model to predict future states and rewards (Sutton and Barto 1998). This property is particularly relevant to autonomous driving, where collecting online trial-and-error data is expensive and potentially unsafe. In practice, the learned model can support planning methods such as model predictive control (MPC) or generate short simulated rollouts for policy and value-function improvement.

Representative MBRL methods include Probabilistic Inference for Learning Control (PILCO), which uses Gaussian processes for data-efficient model learning (Deisenroth and Rasmussen 2011), Dyna-style learning, which augments real experience with model-generated transitions (Sutton 1990), Model-Based Value Expansion (MVE), which improves value estimation with model rollouts(Feinberg et al, 2018), and Model-Based Policy Optimization (MBPO), which limits compounding model error through short branched rollouts(Janner et al, 2019). For AD tasks, MBRL is useful for trajectory planning, closed-loop simulation, and rare-scenario augmentation, but its effectiveness depends heavily on the accuracy and uncertainty calibration of the learned dynamics model. Poor model predictions can produce unsafe or overconfident policies, so MBRL is usually combined with conservative rollout horizons, safety filters, or real-data validation.

2.3.5 RL fine-tuning for foundation-models and VLA driving

Recent autonomous-driving research increasingly uses RL not only as a stand-alone controller, but also as a fine-tuning mechanism for foundation models. PPO is the most representative example: after supervised fine-tuning or imitation learning provides an initial policy, PPO can optimize driving-specific rewards while limiting destructive policy updates. This makes it suitable for reinforcement learning from human feedback (RLHF)-style preference alignment, vision-language model (VLM)/VLA policy refinement, and closed-loop tuning where safety, comfort, route progress, and rule compliance cannot be fully captured by static labels.

Group Relative Policy Optimization (GRPO) extends this direction for large reasoning models by comparing groups of sampled outputs instead of relying on a separately trained value critic. This property is useful when the driving policy must generate both reasoning traces and action tokens, because it can reward correct maneuver choice, concise reasoning, format compliance, and safer trajectory generation with lower critic-training overhead. In VLA-based driving, GRPO-style optimization therefore provides a practical route from language–vision reasoning to language–action alignment.

Several learning algorithms developed in the autonomous-driving community complement PPO and GRPO. Distributional Soft Actor-Critic (DSAC) improves robust continuous control under stochastic traffic dynamics; Reinforcement Learning with Augmented Data (RAD) strengthens visual representations through data augmentation; Integrated Decision and Control (IDC) connects actor–critic learning with integrated decision and control; and safe RL introduces constraints or safety filters for collision avoidance and rule compliance. Together, these methods show how RL serves semantic understanding, action generation, physical constraint satisfaction, and safety-aware decision optimization in modern AD systems.

3 Tool-chains for autonomous driving system

RL-based AV systems rely heavily on high-fidelity simulators and test platforms for algorithm development, validation, and benchmarking. For a review paper, however, listing tools is not sufficient: simulator choice, reward implementation, traffic density, observation interface, and whether testing is open-loop or closed-loop all strongly influence the conclusions drawn from RL experiments. This section introduces representative simulators and platforms that support RL-based research, organized into two categories: (i) AV simulators and (ii) RL platforms.

3.1. Autonomous driving simulators

Autonomous driving simulators typically integrate four capabilities: (i) vehicle behavior simulation, (ii) vehicle dynamics simulation, (iii) sensor simulation, and (iv) AV algorithm integration. Based on the dominant functionality, we categorize representative simulators into three groups: vehicle behavior simulators, sensor simulators (often paired with dynamics engines), and full-stack AV platforms. Table 1 summarizes and compares these simulators.

3.1.1 Vehicle behavior simulators

These simulators emphasize traffic flow modeling and multi-agent interactions, often using bird's-eye-view abstractions. Typical simulators in this category include Highway-env, **SUMO**, **SMARTS** and **Waymax**.

Highway-env (Leurent 2018): A lightweight Python-based environment designed for RL research in highway driving, supporting tasks like lane-changing and merging.

SUMO (Lopez et al, 2018): A widely used traffic simulation package capable of modeling multi-modal traffic and real-world network imports, with extensive API support.

SMARTS (Zhou et al, 2021): A MARL-focused simulator developed by Huawei, featuring a physics-based vehicle model and SUMO integration for scalable traffic simulation.

Waymax (Gulino et al, 2024): A data-driven, hardware-accelerated simulator from Waymo, leveraging real-world driving data to create diverse multi-agent scenarios optimized for large-scale testing on GPUs and TPUs.

3.1.2 Sensor simulators

This type of simulator builds on background vehicle behavior simulation by incorporating sensor data simulation. They typically leverage 3D engines to construct simulation environments, enabling more realistic lighting and physics simulations. Sensor simulators typically rely on underlying vehicle dynamics engines (e.g., **Carsim** (Simulation 2024), **Prescan** (Siemens 2024)) to generate physically realistic sensor outputs, such as LiDAR point clouds and camera feeds. Representative sensor simulators include **Metadrive** (Li et al, 2022), **CARLA** (Dosovitskiy et al, 2017), **Airsim** (Shah et al, 2018), LGSVL (Rong et al, 2020), and **LasVSim** (iDLab, Tsinghua University 2025).

3.1.3 Full-stack AV platforms

These platforms provide modular or end-to-end systems with perception, localization, planning, and control. Representative open-source simulators in this category are Autoware ("Autoware" 2024) and Apollo ("Baidu Apollo" 2024).

Table 1: Comparison of representative autonomous driving simulators by core features.

Simulators	Sensor simulation	High-fidelity visual rendering	Lighting and weather	Vehicle dynamics	Pedestrians	Multi agents	Distributed computation
Highway-env (Leurent, 2018)	-	-	-	-	-	✓	-
SUMO (Lopez et al., 2018)	-	-	-	-	✓	-	-
SMARTS (Zhou et al., 2021)	-	-	-	✓	-	✓	✓
Waymax (Gulino et al., 2024)	-	-	-	✓	✓	✓	✓
Autoware (2024)	✓	-	-	✓	✓	✓	-
Apollo (2024)	✓	-	-	✓	✓	✓	-
MetaDrive (Li et al., 2022)	✓	-	-	✓	-	✓	-
CARLA (Dosovitskiy et al., 2017)	✓	✓	✓	✓	✓	✓	-
AirSim (Shah et al., 2018)	✓	✓	-	✓	✓	✓	-
LGSVL (Rong et al., 2020)	✓	✓	✓	✓	✓	✓	-
LasVSim (iDLab, Tsinghua University, 2025)	✓	✓	✓	✓	✓	✓	✓

Table 2: Comparisons of Leading Platforms for Reinforcement Learning

Platform	Developer	Purpose	Application	Environment	Features and Tools	Characteristics
OpenAI Gym (Brockman, 2016)	OpenAI	Provide a variety of RL environments for testing	RL research and experiment	2D and 3D game environments	Supports multiple environments (classic controls, Atari games, robot simulations, etc.)	The interface is simple and easy to use and is widely used in RL research
OpenAI Universe (OpenAI, 2025)	OpenAI	Extended RL environment to support multiple applications	RL and AI research	A variety of real-world and simulated environments	It includes browsers, desktop apps, and games and supports multiple platforms and APIs	Focus on multi-task learning, able to handle more complex environments and multiple tasks
Unity ML-Agents (Technologies, 2025)	Unity Technologies	Experiment with RL using the Unity game engine	Games with robot control	3D virtual environment	Support Unity engine, suitable for a variety of environment modeling, support deep learning and RL	Powerful graphics engine for complex 3D simulation and RL research
PettingZoo (Terry et al., 2021)	Farama	Multi-agent RL platform	Multi-agent learning	A variety of 2D and 3D environments	Multi-agent support, providing multiple parallel environments, strong scalability	It focuses on multi-agent RL and supports multiple parallel tasks and environments
SMARTS (Zhou et al., 2021)	Huawei Noah's Ark Lab	Multi-agent RL platform	Multi-agent systems, autonomous driving	Complex, realistic multi-agent interactive environment	Support multi-agent interaction, highly scalable, support complex cooperation and competition tasks	It focuses on multi-agent learning and is suitable for cooperation and conflict handling between vehicles in autonomous driving tasks
Jidi (jidi.ai, 2025)	CASIA Collective Decision Intelligence Lab	On-line multi-agent RL open source gaming platform	Games, autonomous driving	Classic chess, automatic driving, supply chain optimization, economic finance, robot control, etc	Support Gym, Mujoco, MiniGrid, StarCraft 2, Google Football, SMARTS and other common simulation environment and game, economy-related self-developed simulation environment	It provides a fair and open third-party on-line evaluation method, providing regular automatic matchmaking and real-time rankings

Simulators enable safe, repeatable testing of RL algorithms, including rare safety-critical events. High-fidelity perception, realistic traffic behavior, and physics-aware control enhance sim2real performance and policy robustness.

3.2. Platforms for reinforcement learning

To support the development and evaluation of RL algorithms, various platforms have been developed across academia and industry. These platforms provide standardized APIs, benchmark environments, and task-specific testbeds for both single-agent and MARL research, as shown in Fig. 3. In the context of AV, platforms with support for complex sensor inputs, multi-agent interaction, and continuous control are particularly important. From an RL-oriented

Table 3: Benchmark-oriented evaluation dimensions for RL-based autonomous driving studies

Task family	Representative simulators / platforms	Typical datasets / benchmarks	Key metrics	Protocol details that affect fairness
Behavior planning and multi-agent interaction	Highway-env, SUMO, SMARTS, Waymax	NGSIM, highD, inD, simulator-defined scenario suites	Collision rate, success rate, TTC, comfort, travel efficiency	Report traffic density, reward design, observation abstraction, and whether evaluation is fully closed-loop and interactive.
Vehicle control	CARLA, MetaDrive, LGSVL, custom control testbeds	Lane-keeping, car-following, path-tracking, eco-driving scenarios	Tracking error, jerk, energy use, safety margin, task completion	Distinguish discrete vs. continuous action spaces, controller interfaces, and whether perception modules are bypassed.
End-to-end and VLM/VLA driving	CARLA, NAVSIM-style environments, closed-loop urban suites	Open-loop driving logs, instruction-tuning sets, closed-loop route benchmarks	Route completion, infraction score, comfort, planning accuracy, reasoning quality	Separate open-loop from closed-loop results and clarify whether RL optimizes actions, trajectories, or language / reasoning tokens.

Table 4: Comparisons of Leading Platforms for VLA

Platform	Developer	Purpose	Application	Environment	Features and Tools	Characteristics
LMDrive (Shao et al., 2024)	OpenDILab	Language-guided closed-loop driving	Instruction-following autonomous driving	CARLA-based LangAuto benchmark	Multi-view sensors, language instructions, and action outputs	Closed-loop benchmark for language-conditioned end-to-end driving
Bench2Drive-VL (Jia et al., 2026)	SJTU ThinkLab	Closed-loop VLM/VLA evaluation	Driving evaluation with language interfaces	CARLA and Bench2Drive	VLM interface, VQA generation, and closed-loop execution	Evaluation suite connecting visual–language reasoning with driving behavior
Bench2ADVLM (Zhang et al., 2025)	Beihang University	Interactive closed-loop testing	Safety diagnosis and failure discovery	Simulation and physical vehicle platforms	Language-to-action conversion and hierarchical testing	Supports closed-loop evaluation across simulated and physical settings
DriveAction (Hao et al., 2025)	Li Auto	Action-driven VLA benchmark	Human-like driving decision evaluation	Real-world driving scenarios	Action labels, VLA QA pairs, and tree-structured evaluation	Action-centered benchmark beyond conventional driving VQA
CoVLA (Arai et al., 2025)	Turing Motors	VLA dataset for training and evaluation	Language-conditioned trajectory generation	Real-world driving videos	Language descriptions, maneuver labels, and future trajectories	Reusable dataset for data-driven VLA policy learning
Impromptu VLA (Chi et al., 2025)	AIR, Tsinghua University	Open data and benchmark for VLA driving	Corner-case and unstructured-scenario driving	Curated public driving clips	Planning QA, action trajectories, open data, and open weights	Supports VLA training and evaluation in difficult scenarios

AD perspective, three additional capabilities deserve explicit attention: whether a platform supports closed-loop interactive training, how easily task-specific reward functions can be customized, and whether logged-data replay or world-model-generated rollouts can be integrated into the training loop. Table 2 summarizes key RL platforms commonly used for AV tasks. Note that SMARTS appears in both the simulator and platform lists; here we list it as an RL platform when its multi-agent learning interface and benchmarking role are the primary focus.

To make cross-paper comparison more evidence-centered, Table 3 summarizes the benchmark dimensions that should be reported together with algorithm names. These dimensions explain why two papers using the same RL method may still be incomparable when their simulator fidelity, reward function, or interaction protocol differs.

3.3. Platforms for VLA

With the rapid integration of large vision-language models (VLMs) into autonomous driving, a new category of tool-chain has emerged to complement conventional simulators and general RL platforms. Existing VLM-based driving resources mainly focus on scene-level understanding, such as object recognition, traffic-context reasoning, visual grounding, and driving-related question answering. Related vision-and-language navigation (VLN) studies further show how natural-language instructions can be grounded in embodied navigation behavior (Anderson et al, 2018; Krantz et al, 2020), while driving-command grounding datasets such as Talk2Car (Deruyttere et al, 2019) connect language instructions with real traffic scenes. However, these resources usually remain at the level of visual-language understanding, instruction grounding, or navigation-level reasoning, and are not designed to directly evaluate vehicle-level trajectories, control signals, or closed-loop driving behavior.

As summarized in Table 4, existing resources can be roughly grouped into two categories. The first category provides simulation-based or closed-loop evaluation interfaces, represented by LMDrive (Shao et al, 2024), Bench2Drive-VL (Jia et al, 2026), and Bench2ADVLM (Zhang et al, 2025). The second category provides

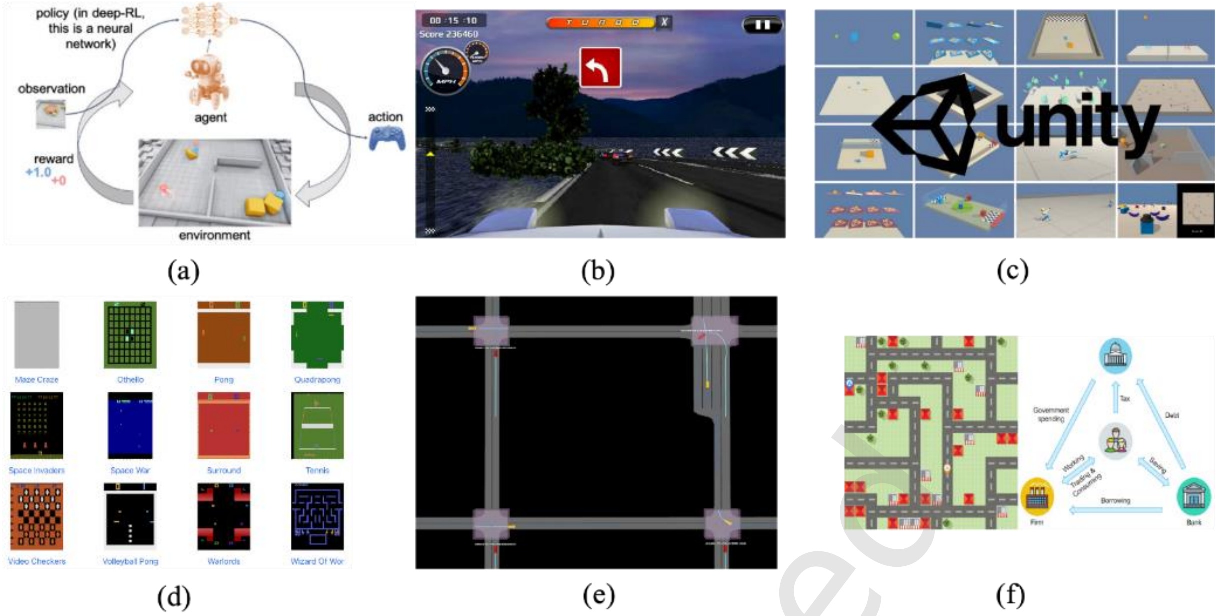

Fig. 3. Simulation platforms for reinforcement learning research. (a) OpenAI Gym (Brockman, 2016): a widely used toolkit for developing and benchmarking RL algorithms in diverse, lightweight environments; (b) OpenAI Universe (OpenAI, 2025): a platform that enables RL agents to interact with real-world applications and games through a general-purpose API; (c) Unity ML-Agents (Unity Technologies, 2025): a flexible, high-fidelity simulation toolkit based on the Unity engine for training intelligent agents in 2D and 3D environments; (d) PettingZoo (Terry et al., 2021): a library of MARL environments supporting a variety of discrete and continuous games; (e) SMARTS (Zhou et al., 2021): an autonomous driving simulation platform focusing on realistic traffic scenarios and multi-agent interactions; (f) Jidi (jidi.ai, 2025): a Chinese MARL competition platform offering diverse tasks, including traffic control and economic simulation.

Table 5: Representative RL-related approaches for environment understanding and world modeling

Category	**Representative work**	**Predicted representation**	**RL-related role**	**Main contribution or limitation**
Behavior modeling	(Bhattacharyya et al., 2022)	Multi-agent driver behavior	GAIL-based imitation from demonstrations	Captures interaction-aware human driving, but depends strongly on expert trajectory coverage.
Behavior modeling	(Choi et al., 2021)	Urban vehicle trajectories	Adversarial imitation in a POMDP setting	Generates socially plausible trajectories for open-loop forecasting rather than direct vehicle control.
Behavior modeling	(Azadani and Boukerche, 2024, 2023)	Context-aware future motion	DIRL for reward and intent inference	Uses scene context and transformers to improve multimodal trajectory prediction, but remains demonstration-driven.
Behavior modeling	(Huang et al., 2023, 2021)	Personalized reward priors and behavior plans	Deep MEIRL for intent-aware planning	Learns interpretable reward priors for behavior prediction, with limited evidence on closed-loop robustness.
Behavior modeling	(Alsaleh and Sayed, 2021; Sackmann et al., 2022)	Interaction-aware reward functions	AIRL for multi-agent reward reconstruction	Improves interpretability and transfer, but adversarial training is computationally heavier.
World modeling	(Y. Wang et al., 2024)	Multiview future scene forecasts	Predictive world model for planning	Connects scene forecasting and planning, but evidence is still mainly benchmark-oriented.
World modeling	(Hu et al., 2023;X. Wang et al., 2024)	Generative future driving scenes	Data-driven world model / simulator	Promising for synthetic data and scenario generation, yet closed-loop policy validation remains limited.
World modeling	(Chen et al., 2021; Gao et al., 2024; Q. Li et al., 2024)	BEV states and latent dynamics	Model-based RL with latent imagination or rollout	Improves sample efficiency for end-to-end or policy learning, but model error can accumulate under long-horizon interaction.
Structured scene representation	(Liu et al., 2025; Tang et al., 2024; Yu et al., 2023)	3D occupancy / sparse occupancy	Compact state abstraction for downstream planning or RL	Provides richer geometry and frees RL from raw-sensor burden, although most occupancy papers are not RL methods themselves.

Table 6: Summary of RL-Based Approaches for Vehicle Control

Vehicle control	References	Algorithm	Application scenario	Focus/advantages
Longitudinal control	(Shan et al., 2020)	PPO	Path tracking	Balances tracking accuracy and ride quality in real-time.
	(Chen and Chan, 2021)	PPO	Path tracking	Improves robustness and accuracy, especially at low speeds.
	(Shi et al., 2021)	DPPO	Mixed traffic	Enhances string stability, car-following, and energy efficiency.
	(Yue et al., 2024)	Linear Feedback + DPPO	Hybrid car-following	Dampens traffic oscillations and ensures stability.
	(Du et al., 2022)	DDPG	Speed control	Optimizes vertical ride comfort, energy efficiency, and real-time computation.
	(Liu et al., 2022)	SAC + IL	Urban driving	Improves training efficiency and tracking performance.
Lateral control	(Ye et al., 2020)	PPO	Lane-changing in dense traffic	Enables safe, smooth, and efficient maneuvers with improved learning efficiency and stability.
	(G. Li et al., 2022)	DQN	Risk-aware lane-change	Incorporates probabilistic risk assessment to improve safety and performance in static and dynamic obstacle scenarios.
	(Marvi and Kiumarsi, 2021)	Off-policy RL + CBFs	Lane-keeping control	Designs safe optimal controllers, ensuring robust safety and performance.
	(X. He et al., 2022)	Observation Adversarial RL	Lane-changing	Enhances robustness and performance against observation perturbations in mixed traffic.
	(Chu et al., 2022)	DDQN	Path planning	Enables AUVs to navigate unknown environments.
	(Toromanoff et al., 2020)	Rainbow-IQN	Complex urban driving	Handles diverse urban driving tasks such as lane keeping, obstacle avoidance, and traffic light detection in the CARLA.
Simultaneous control	(Guo et al., 2021)	HDQPG	Multi-lane eco-driving	Achieves up to 46% fuel reduction with minimal impact on travel time by integrating longitudinal and lateral control.
	(Fu et al., 2020)	DDPG	Emergencies braking	Optimizes efficiency, accuracy, and passenger comfort through a multi-objective reward function.

DriveAction (Hao et al, 2025), CoVLA (Arai et al, 2025), and Impromptu VLA (Chi et al, 2025). Together, these resources expand the autonomous-driving tool-chain from visual-language understanding to language-conditioned action generation and evaluation, making it possible to study VLA agents at the level of driving decisions, trajectories, and executable behaviors.

Overall, platforms for VLA in autonomous driving are still emerging, with current resources mainly focusing on closed-loop evaluation and action-centered data construction. Although native RL fine-tuning is not yet widely supported, these platforms provide the visual inputs, language interfaces, action targets, and evaluation protocols needed for developing RL-based VLA driving systems.

4 Reinforcement learning application for AD system in different functions

A conclusion is not restatement of the abstract, but to stress the importance of the work, to give the paper a sense of completeness, and leave a final impression on the readers. The conclusion section is the last section of the paper to be numbered.

4.1. Function 1: environment understanding and world modeling

Environment understanding in autonomous driving (AD) covers nearby-agent detection, interaction-intent inference, scene-geometry representation, and prediction of how the scene may change under candidate ego actions. RL-related studies in this area use two different approaches. Imitation learning and inverse RL extract behavior priors from demonstrations. World-model methods predict future observations, BEV states, occupancy, or latent dynamics for planning and policy optimization, either as structured state representations or as synthetic rollouts. We organize this section around these roles instead of grouping the literature solely as variants of GAIL, DIRL, or AIRL.

Most studies in the first group do not require to train an RL controller through online closed-loop interaction. Instead, they use offline demonstrations to estimate rewards, imitate human intent, or generate trajectories. In this review, we classify them as behavior-modeling and reward-inference methods within environment understanding.

4.1.1 Behavior modeling and reward inference

Early RL-related work in environment understanding used offline demonstrations to predict the behavior of surrounding agents. GAIL-based methods such as PSGAIL (Bhattacharyya et al., 2022) and TrajGAIL (Choi et al., 2021) formulate human driving and trajectory generation as imitation. This avoids specifying a reward by hand, but the resulting models are usually evaluated by their similarity to expert data. Similarity scores alone do not show whether these methods improve downstream closed-loop safety.

DIRL and related maximum-entropy methods infer latent rewards

Table 7: RL in end-to-end AD (Two-stage)

Method	System	RL usage	Task	State	Action	Reward	Network	Simulator	Algorithm
N. Ashraf et al. (2021) (Ashraf et al., 2021)	Two-Stage	Pure RL	Racing-style lane-following	Vehicle speed and distance	Throttle/brake + steering	Progress along track, penalty for leaving track	MLP	TORCS	DDPG
X. Fang et al. (2022) (Fang et al., 2022)	Two-Stage	Pure RL	Highway scenario	Shape, speed, position of ego and 5 social vehicles	Desired longitudinal velocity	Efficiency, penalty for collision	MLP	Real-world data replay	Offline RL
Y. Du et al. (2022) (Du et al., 2022)	Two-Stage	Pure RL	Rough pavements speed control	Previous acceleration, current speed, future max comfortable speed	Acceleration	Driving and energy efficiency, longitudinal and vertical comfort	MLP	Dataset replay	DDPG
C. Hoel et al. (2023) (Hoel et al., 2023)	Two-Stage	Pure RL	Truck control at the intersection	Position, speed, heading of ego and vehicles	High-level decisions	Task completion, penalty for collision	MLP	SUMO	EQN
Y. Guan et al. (2022) (Guan et al., 2022)	Two-Stage	Pure RL	General driving scenario	States of ego and nearby vehicles	Acceleration and steering	Safety constraint, efficiency and comfort objective	MLP	SUMO	GEP
Zhong Cao et al. (2021)(Cao et al., 2021)	Two-Stage	RL + Rule-Based	Two-lane roundabout	Ego and the surrounding vehicles' state	Acceleration and degree of lane change	Penalty for collision	MLP	CARLA	DQN
Y. Fu et al. (2022) (Fu et al., 2022)	Two-Stage	RL + Federated Learning	Two-lane collision avoidance with CAVs	Speed, position of object and other vehicles	Acceleration of object vehicle	Penalty for collision, jerk and being conservative	MLP	Custom simulator	DDPG
B. Gangopadhyay et al. (2023) (Gangopadhyay et al., 2022)	Two-Stage	RL + CBF	Four driving tasks	Position, speed and relative pose to lane and key vehicle	Acceleration and steering angle	Penalty for collision, and rewards differ among tasks	MLP	Custom environment	PPO
S. Hwang et al. (2022) (Hwang et al., 2022)	Two-Stage	RL + FSM	Lane-merging scenario	Ego and four target vehicles' state	Acceleration and steering angle	Risk and ride-comfort penalty, approach reward	MLP	Custom simulator	SAC
R. Bautista-Montesano et al. (2022) (Bautista-Montesano et al., 2022)	Two-Stage	RL + MPC	Unsignalized T-intersection navigation	Position, yaw and speed of ego and surrounding vehicles	Vehicle power and path selection	Reward for expected speed, yaw rate, small cross-track error and angle to path.	MLP	Custom simulator	TD3
Y. Lu et al. (2023) (Lu et al., 2023)	Two-Stage	RL + IL	Urban driving	Vehicle states, road-graph points, traffic lights signals, and route goals	Acceleration and steering	Penalty for collision and off-road	Transformer	Log replay	BC-SAC
J. Nan et al. (2024) (Nan et al., 2023)	Two-Stage	Inverse RL	Merge scenario	Ego speed, acceleration, position, relative distance and velocity to 5 surrounding vehicles	Trajectory	Learned from demonstration inversely	CNN + MLP	Log-sim testing on NGSIM	Inverse RL
J. Duan et al. (2024) (Duan et al., 2024)	Two-Stage	RL + Safety Shield	On-ramp merging scenario	States of Ego and 8 vehicles	Acceleration and front wheel angle	Safety, task completion, efficiency and comfort	MLP	SUMO	DSAC
Q. Li et al. (2024) (Qifeng Li et al., 2024)	Two-Stage	RL + World Model	Urban driving	BEV features	Throttle/Brake and steering discretization	Speed and travel reward, penalty for deviation and steering	CNN + GRU	CARLA V2	Model Based RL

instead of matching actions directly. Representative studies use IRL to estimate context-dependent reward priors for motion forecasting, personalized behavior modeling, and pedestrian or micromobility intent prediction (Azadani and Boukerche, 2024, 2023; Huang et al., 2023, 2021; Wang et al., 2023). AIRL further combines reward recovery with adversarial policy learning and has been used to model cyclist-pedestrian interaction, reconstruct driver preferences, and generate interaction-aware highway behavior (Alsaleh and Sayed, 2021; Radtke et al, 2023; Sackmann et al., 2022; Wen et al., 2023). These methods remain valuable because they provide interpretable human-behavior priors for downstream planners. Their main limitation, however, is that most evidence is still based on open-loop likelihood, trajectory similarity, or reward reconstruction quality rather than on interactive closed-loop deployment.

4.1.2 World models and predictive scene representations

Recent work has expanded Function 1 from behavior forecasting toward world modeling. In this setting, a world model learns how the surrounding scene evolves over time, often conditioned on ego history and candidate actions. The learned state can take several forms. BEV world models represent future road layout, agents, and

Table 8: RL in end-to-end AD (One-Stage and VLM-Based)

Method	System	RL usage	Task	State	Action	Reward	Network	Simulator	Algorithm
J. Chen et al. (2022) (Chen et al., 2021)	One-Stage	RL + World Model	Urban driving	Front view RGB image + 2D lidar image	Throttle/brake + steering	Efficiency, penalty for collision and out of lane	CNN + MLP	CARLA	SAC
K. Stachowicz et al. (2023) (Stachowicz et al., 2023)	One-Stage	Pure RL	Small-scale RC car high speed driving	Front view RGB image + Position estimation	Throttle/brake + steering	Speed facing checkpoint, penalty for stuck and collision	CNN + MLP	CARLA	SAC
H. Gao et al. (2025) (Gao et al., 2025)	One-Stage	RL + IL	Urban driving	Multi-view image sequences	Linear and angular velocities	Penalty for collision and deviation from expert trajectory	Transformer	3DGS	PPO
Z. Zhang et al. (2021) (Zhang et al., 2021)	One-Stage	RL + Teacher-Student	Urban driving	Camera images for student model, BEV image for teacher model	Throttle/brake + steering	Penalty large steering changes and high ego speed	CNN	CARLA	PPO
X. Jia et al. (2023) (Jia et al., 2023)	One-Stage	RL + Teacher-Student	Urban driving	Camera images for student model, BEV image for teacher model	Throttle/brake + steering	Penalty large steering changes and high ego speed	CNN	CARLA	PPO
P. Wu et al. (2022) (P. Wu et al., 2022)	One-Stage	RL + IL	Urban driving	Front camera image + speed and navigation	Throttle/brake + steering	Speed prediction accuracy	CNN + MLP	CARLA	Auxiliary loss
P. Cai et al. (2022) (Cai et al., 2021)	One-Stage	RL + Contrastive Learning	T-shaped junction	Lidar point clouds	Throttle/brake + steering	Efficiency, penalty for collision	CNN + MLP	CARLA	D3QN
Y. Li et al. (2025) (Yongkang Li et al., 2025)	VLM-based	RL + SFT	Urban driving	Ego status, camera input and navigation information	Future trajectory	Predictive driver model score	InternVL3-8B + Diffusion Model	NAVSIM	GRPO
B. Jiang et al. (2025) (Jiang et al., 2025)	VLM-based	RL + SFT	Urban driving	Front-view image, planning prompt of navigation and speed	Path and speed text description	Planning accuracy, action-weighted, diversity and format compliance	Qwen2VL-2B	Distillation dataset	GRPO
Y. Li et al. (2025) (Yue Li et al., 2025)	VLM-based	RL + SFT	Urban driving	Multi-view images and text prompt	Reasoning chains and trajectory outcomes	Trajectory accuracy, meta-action correctness, repetition penalty, format compliance	Intern2VL-4B	Reasoning-Planning COT dataset	GRPO
Z. Zhou et al. (2025) (Zhou et al., 2025)	VLM-based	RL + SFT	Urban driving	Multi-view camera streams, instruction, ego status	Reasoning and action token	Safety and comfort, penalty for slow thinking	Qwen2.5VL-3B	nuPlan	GRPO
Z. Huang et al. (2025) (Huang et al., 2024)	VLM-based	Pure RL	Urban driving	RGB images	Throttle/brake + steering	Contrasting language goal, speed/position alignment	CLIP + MLP	CARLA	SAC

traffic context in a planner-friendly spatial frame. For example, Chen et al, (Chen et al, 2021) learn latent scene dynamics and semantic BEV predictions to improve end-to-end urban driving, while Think2Drive (Qifeng Li et al, 2024) learns a latent world model that predicts multi-step BEV observations and then performs model-based RL in the latent space. These works are especially relevant because they create a tighter bridge between environment understanding and downstream policy optimization than pure behavior cloning or reward inference.

A second trend is future scene prediction and generative world modeling. Wang et al, (Yuqi Wang et al, 2024) forecast multiview future observations to support planning, while DriveDreamer (Xiaofeng Wang et al, 2024) and Gaia-1 (Hu et al, 2023) treat autonomous driving scenes as a generative modeling problem, enabling data-driven scenario synthesis and controllable future rollout. From the perspective of RL, such models are attractive because they can serve as learned simulators for imagination, counterfactual evaluation, or offline data augmentation. However, many of these results are still strongest in open-loop forecasting or generation benchmarks, so their benefit for safety-critical interactive driving should be interpreted cautiously.

A third trend is to use structured scene representations, especially occupancy prediction, as the state substrate for downstream planning and RL. Occupancy models predict which 3D regions are free, occupied, or semantically typed, thereby offering a richer geometry-aware interface than raw images alone. Efficient occupancy predictors such as FlashOcc (Yu et al, 2023), SparseOcc (Tang et al, 2024), and fully sparse occupancy models (Liu et al., 2025) are not RL algorithms by themselves, but they are highly relevant to RL-oriented AD because they provide compact state abstractions, improve physical consistency, and can support collision-aware rollout evaluation. Closely related latent-dynamics models, such as the semantic masked world model in (Gao et al, 2024), further improve sample efficiency by masking irrelevant details and focusing policy learning on behaviorally important scene factors.

4.2. Function 2: Decision making and control

RL studies in autonomous-vehicle decision-making and control cover high-level tasks such as lane changing and eco-driving, as well as low-level steering and speed control. AD control is commonly divided into lateral and longitudinal components (Hao Liu et al., 2023). Some studies address these components separately, while others coordinate them within a single control framework (Kuutti et al., 2020). This section reviews RL methods for lateral, longitudinal, and combined control, together with their application settings and limitations.

The action interface provides a practical basis for comparing these methods. DQN-based methods generally operate on discrete maneuver choices, such as lane changes and behavior-planning actions. DDPG, TD3, SAC, and DSAC are used when steering and acceleration are represented as continuous actions. PPO is used for on-policy closed-loop training and for policy updates with safety constraints. Algorithm comparisons should therefore be considered in relation to the control formulation.

4.2.1 Longitudinal control

Longitudinal control determines a vehicle's acceleration and deceleration. The controller must maintain the target speed and following distance while reducing the risk of rear-end collisions. RADAR and LiDAR provide measurements of relative speed and distance. In RL-based studies, the reward may include speed, stability, energy use, or a combination of these objectives.

Several studies apply RL to longitudinal control. Shan et al. combine a Pure Pursuit (PP) controller and a PID controller with PPO. PPO adjusts the weighting between PP and PID to balance path-tracking accuracy and ride quality in real time (Shan et al., 2020). Chen et al. also combine PP with PPO to correct baseline path-tracking errors and improve robustness and accuracy at low speeds (Chen and Chan, 2021). For mixed traffic, Shi et al. use Distributed PPO (DPPO) for cooperative longitudinal control of Connected and Automated Vehicles (CAVs). Their method uses subsystem-level learning of human-driven Vehicle (HDV) characteristics to address string stability, car-following efficiency, and energy efficiency (Zhao et al., 2019). Yue et al. combine a linear feedback controller with DPPO to reduce traffic oscillations, maintain stability, and improve distance keeping (Yue et al., 2024). Du et al. use crowdsourced data with DDPG for speed control on rough pavements, considering vertical ride comfort, energy efficiency, and real-time operation. The study reports improvements over traditional optimization methods (Hua et al., 2023; Hua et al., 2023; Du et al., 2022). Liu et al. combine SAC with imitation learning (IL) to improve RL training efficiency and generalization in urban driving (Liu et al., 2022).

4.2.2 Lateral control

Lateral control includes maneuver decisions, such as lane changes and risk-aware planning, as well as low-level lateral actuation. Many studies use vision-based observations with RL (G. Chen et al., 2024; Hua et al., 2019).

For lane-changing tasks, Ye et al. use PPO to plan maneuvers in dense traffic. The method targets safety, smoothness, and efficiency, and the authors report improvements in learning efficiency and stability (Ye et al., 2020). Li et al. combine DQN with probabilistic risk assessment for lane-change decisions in scenarios with static and dynamic obstacles (G. Li et al., 2022). For lane keeping, Marvi et al. combine control barrier functions (CBFs) with off-policy RL to incorporate safety constraints into the controller (Marvi and Kiumarsi, 2021). He et al. formulate observation perturbations as a constrained observation-robust Markov decision process and use Bayesian optimization for lane-change decisions in stochastic mixed traffic (X. He et al., 2022). In a related underwater-navigation study, Chu et al. use DDQN with a dynamic composite reward for underactuated autonomous underwater vehicles operating in unknown environments with ocean-current disturbances (Chu et al., 2022). Toromanoff et al. use Rainbow-IQN with implicit affordances for urban tasks including lane keeping, obstacle avoidance, and traffic-light detection, with evaluation in the CARLA challenge.

4.2.3 Simultaneous lateral & longitudinal control

Joint lateral and longitudinal control combine steering and speed-related decisions for scenarios including intersection navigation, autonomous lane changes, and coordinated driving (G. Chen, Zheng Gao, et al., 2024; Hua et al., 2024).

RL algorithms optimize steering, throttle, and braking simultaneously to ensure smooth and coordinated vehicle control (Chen et al, 2023). For instance, RL policies guide vehicles safely through intersections while accounting for dynamic traffic conditions, and enable autonomous lane-changing systems by balancing speed and position adjustments for safety and efficiency. Guo et al, propose a hybrid RL-based eco-driving algorithm (HDQPG) that combines DDPG for continuous longitudinal control with DQN for discrete lane-changing decisions, achieving up to 46% fuel reduction with minimal travel time impact in multi-lane urban signalized corridors (Guo et al, 2021). Similarly, Fu et al, present a DDPG-based autonomous braking strategy with a multi-objective reward function, optimizing efficiency, accuracy, and passenger comfort during emergency situations, demonstrating improved safety and decision-making performance (Fu et al, 2020). Tang et al, propose a hierarchical control structure for vision-based AD of hybrid electric vehicles, integrating YOLO-based object detection with a DQN for car-following and energy management, achieving a fuel economy of 5.76 L/100 km and real-time operation with a 0.26 s control loop on embedded devices (Tang et al, 2022).

4.2.4 Summary of current algorithms and applications

Table 6 summarizes RL algorithms, their application scenarios, and specific focuses in recent studies.

RL has demonstrated strong potential in addressing challenges across the AD control stack, including both high-level decision-making (e.g., trajectory planning, lane selection) and low-level control (e.g., speed regulation, steering optimization) (Aradi, 2020). Studies have applied PPO, DQN, and DDPG to lateral-control tasks such as lane keeping and lane changing, and to longitudinal tasks such as path tracking and speed optimization. Joint-control methods address scenarios including eco-driving and emergency braking, where efficiency, safety, and comfort are considered together. Open issues include sample efficiency, perception uncertainty, and real-time execution. Future studies can examine multi-agent RL, reward design, and combinations of RL with other learning methods, and evaluate their effects on the adaptation and scalability of autonomous-vehicle control.

4.3 Function 3: End-to-End AD systems to VLM/VLA-

based driving

End-to-end (E2E) driving aims to map sensor inputs directly to driving actions, reducing modular complexity and error propagation. RL contributes to this paradigm in two primary ways: (i) fine-tuning policies pre-trained via supervised or imitation learning to improve robustness under distribution shift, and (ii) optimizing high-level decision-making components in unified perception–action pipelines. Recent work explores the integration of large vision or vision-language models to enhance scene understanding and reasoning, while RL provides task-specific grounding through interaction-based optimization. A critical distinction, however, is whether evidence comes from open-loop prediction benchmarks or from interactive closed-loop driving, because gains in planning accuracy or reasoning quality do not automatically transfer to safer online behavior. For this reason, we discuss E2E RL not only by model family, but also by evaluation maturity and by the extent to which explicit planning or safety modules are retained.

Reinforcement learning has been widely applied in the design of end-to-end AD systems to reduce the reliance on human driving data, enhance safety in long-tailed driving scenarios, and improve alignment with human preferences. In this section, we categorize reinforcement learning applications in end-to-end AD systems into three types: two-stage end-to-end, one-stage end-to-end, and VLM-based end-to-end. Additionally, we summarize the methods based on different RL usage types, and the specific AD tasks they target. Detailed information about all methods is listed in Table 7 and Table 8.

4.3.1 RL for two-stage end-to-end systems

Two-stage end-to-end AD systems emerged after 2021, where the input is structured information from the perception module, and the output is vehicle control commands or desired trajectories. Typically, reinforcement learning in these systems utilizes vectorized inputs and outputs, targets specific scenario tasks, and trains policy using traffic flow simulator. According to the RL application approach, the main methods can be divided into pure RL and RL combined with other methods. In 2021, Ashraf et al, (Ashraf et al, 2021) optimized the hyperparameters of the DDPG algorithm using the Whale Optimization Algorithm, training in the TORCS environment. In the racing-style lane-following task, they achieved the maximum cumulative reward. In 2022, Fang et al, (Fang et al, 2022) trained an offline RL model for highway car-following tasks using the states of ego and 5 social vehicles as inputs, controlling longitudinal speed to ensure safe and efficient passage. The policy was trained and tested using a simulator constructed with real-world data replay. Du et al, (Du et al, 2022) used DDPG to address the rough pavement speed control task, where the policy took previous acceleration and current states as inputs, outputting acceleration to improve vertical comfort. The simulation environment for this task was similarly built using dataset replay. In 2023, Hoel et al, (Hoel et al, 2023) proposed the Ensemble Quantile Networks (EQN) method for truck control at intersections, combining distributional RL with an ensemble approach to provide a complete uncertainty estimate, balancing risk and time efficiency in various occluded intersection scenarios. Unlike the above methods, Guan et al, (Guan et al, 2022) proposed an integrated decision and control method for general driving tasks, which abstracted AD decision and control tasks into a constrained optimal control problem, seamlessly matching the actor-critic framework in RL. The resulting critic network was used for path selection, while the actor network facilitated collision-free tracking, achieving safe, efficient, and real-time end-to-end decision-making and control. This method also supported China's first public road verification of an end-to-end AD system (Guan et al, 2026).

In two-stage end-to-end systems, another category of reinforcement learning methods often works in conjunction with other approaches. Cao et al, (Cao et al, 2021) introduced a confidence-aware RL method combining RL with driving rules, implementing a strategy-switching mechanism that takes the strengths of both approaches, outperforming pure RL and rule-based methods in the CARLA two-lane roundabout scenario. Fu et al, (Fu et al, 2022) combined RL with federated learning to train local driving models for connected and automated vehicles (CAVs) to address collision avoidance tasks, improving model accuracy and learning efficiency through online knowledge transfer and aggregation. Gangopadhyay et al, (Gangopadhyay et al, 2022) combined RL with control barrier functions to learn policies for real vehicles with enhanced dynamics, ensuring safety and stability across multiple driving tasks. In 2022, Hwang et al, (Hwang et al, 2022) used a finite state machine to provide a framework for four cut-in phases in lane-merging scenarios, employing the SAC algorithm in the lane-change phase to achieve safe and efficient cut-in behavior, surpassing the performance of rule-based and end-to-end models. Bautista-Montesano et al, (Bautista-Montesano et al, 2022) proposed an RL-MPC coupled method to address the unsignalized T-intersection navigation problem, where RL and MPC navigation algorithms were used to track feasible trajectories, running in parallel and selected to enhance ego vehicle safety and driving efficiency. Lu et al, (Lu et al, 2023) introduced the BC-SAC algorithm, combining imitation learning with RL, and used data replay on large amounts of real-world human driving data to fine-tune pre-trained models with RL, significantly reducing failure rates in out-of-distribution and challenging scenarios. Duan et al, (Duan et al, 2024) trained a driving strategy offline using the DSAC algorithm in an on-ramp merging scenario, considering driving safety, efficiency, and task completion. Online inference further corrected dangerous behaviors using a safety shield, leading to improved performance compared to baseline methods. Li et al, (Qifeng Li et al, 2024) used Carla V2 to learn a world model that predicts multi-step BEV images, and based on this, applied model-based RL to optimize throttle, brake, and steering, significantly enhancing RL training efficiency through state latent space representation and tensor parallel computing (Wang et al, 2026).

4.3.2 RL for one-stage end-to-end systems

One-stage end-to-end AD systems, which became prominent after 2023, take raw sensor data as input and output vehicle control commands or desired trajectories. Due to low training efficiency and poor accuracy with raw sensor inputs, these systems often combine RL with other algorithms to improve trainability and performance. Additionally, the inherent difficulty in modeling raw sensor data in real-world vehicles leads to reinforcement learning in one-stage end-to-end systems primarily relying on image-based simulations, scene reconstruction, and knowledge distillation. In 2021, Chen et al, (Chen et al, 2021) trained a one-stage end-to-end model in the CARLA simulator using front view RGB images and 2D LiDAR images as inputs, with throttle/brake and steering as outputs. They also trained a latent state world model for predicting future sensor states and semantic bird's-eye view maps, greatly enhancing RL sample efficiency, surpassing multiple baseline RL methods in urban scenarios. Stachowicz et al, (Stachowicz et al, 2023) trained a small-scale RC car for high-speed driving in a real environment using only SAC. The model took front view RGB images and position estimation as inputs, controlling throttle and steering in real time, demonstrating emergent aggressive driving skills such as timing braking and avoiding obstacles. Gao et al, (Gao et al, 2025) established a 3DGS-based closed-loop RL training paradigm, using real-world multi-view image sequences as inputs and 3D visual reconstruction to help the PPO policy explore trial-and-error, handling out-of-distribution problems and significantly outperforming imitation learning methods in safety and other closed-loop metrics. Unlike directly using camera images for RL training, Zhang et al, (Zhang et al, 2021) combined RL with a Teacher-Student paradigm, where the teacher network used ground-truth BEV images as inputs for efficient RL closed-loop training. The student network, using camera images, employed knowledge distillation to transfer the teacher network's RL policy, greatly improving training efficiency. In 2023, Jia et al, (Jia et al, 2023) modified the student network output to predict BEV images rather than low-level actions, feeding them into the teacher network to obtain optimal actions. To address the distribution gap between predicted and ground-truth images, they used adapters with a feature alignment objective function between the student and teacher modules.

4.3.3 Reinforcement learning for VLM-based end-to-end systems

Since the rise of Vision-Language Models (VLMs) in 2024, VLM-based end-to-end AD systems have become popular. These systems differ from one-stage end-to-end systems in the following ways: 1) They integrate a language modality on top of visual and action modalities, allowing input of language instructions and output of scene descriptions and reasoning processes, enabling human interaction and interpretability. 2) Models embedding VLMs have many more parameters, making training more challenging. In these systems, RL is typically used to fine-tune pre-trained model parameters, with open-loop simulations used for training. Li et al, (Yongkang Li et al, 2025) used InternVL3-8B as the base model and adopted a three-stage paradigm for training, first using large-scale driving question-answering datasets to train the VLM, then employing a diffusion-based planner to generate multimodal trajectories, and finally fine-tuning the diffusion planner with RL using the NAVSIM non-reactive simulator. This method achieved superior closed-loop performance on the NAVSIM benchmark. Jiang et al, (Jiang et al, 2025) used Qwen2VL-2B as the base model, combining supervised fine-tuning (SFT) and GRPO to enhance semantic reasoning for path and speed actions. RL was used with rewards such as reasoning accuracy, diversity, and format compliance, achieving better performance compared to SFT and non-reasoning models. Li et al, (Yue Li et al, 2025) used Intern2VL-4B as the base model, first fine-tuning it with supervised training on a dataset containing long/short Chain-of-Though (COT) reasoning, and then exploring diverse reasoning COTs and predicting trajectories using GRPO. This method demonstrated superior performance on the nuScenes benchmark compared to existing VLMs. Zhou et al, (Zhou et al, 2025) proposed a novel VLA model that unifies reasoning and action generation within a single autoregressive generation model. This model performs semantic reasoning and trajectory planning directly from raw visual inputs and language instructions, with RL used to further enhance safety and suppress unnecessary reasoning. Huang et al, (Huang et al, 2024) used the CLIP to construct a contrasting language reward function involving both speed and position preferences. They trained a one-stage end-to-end model in the CARLA simulator with SAC, and achieved better safety and robustness in edge scenarios. Beyond using large models as the backbone of an end-to-end policy, an emerging direction is to use foundation models as semantic priors for RL. The foundation models can summarize scene context, provide high-level goals, or guide exploration with language-conditioned preferences. Thus, the downstream control can be done with smaller policy heads. This perspective is particularly relevant for modular AD, where large models can improve decision quality without replacing the existing planner or controller. It is also important to note that VLA-style systems are usually not trained with RL alone. Supervised learning and imitation learning typically provide the initial behavioral prior, action alignment, and reasoning-format supervision, while RL mainly refines safety, efficiency, and reasoning economy. A promising frontier therefore lies in tighter integration among foundation models, RL, and explicit planning/safety modules such as MPC or CBF shields, which can preserve semantic reasoning while retaining stronger modular guarantees.

5 Current challenges

5.1 Safety assurance

RL policies for autonomous driving must optimize driving performance subject to explicit safety constraints. A low-probability unsafe action can still cause serious harm in real traffic. Evaluation should therefore include average return, collision avoidance, rule compliance, and behavior in rare interactive events.

5.1.1 Existing solutions and limitations

Safe RL studies commonly use two formulations, iterative unconstrained optimization and constrained policy optimization. The iterative formulation converts a constrained RL problem into a sequence of unconstrained problems, often with Lagrangian relaxation. This introduces an outer optimization loop and may lead to slow convergence (Ganai et al., 2024). Constrained policy optimization imposes safety constraints during each policy update. For example, Constrained Policy Optimization (CPO) uses a trust-region approximation for this constrained update (Achiam et al., 2017).

These formulations have limitations in real-world autonomous driving. Iterative methods often suffer from prohibitive sample complexity, while strict constraints may be infeasible or overly conservative early in training. Furthermore, existing safety guarantees are frequently derived from simulations that rely on simplified vehicle dynamics, hand-crafted rewards, restricted scenario distributions, or external safety shields. While such empirical results indicate promising progress, they are insufficient to establish readiness for production-level deployment.

5.1.2 Future directions

Future studies can integrate formal safety filters, uncertainty-aware policy learning, interpretable reward design, and standardized safety-critical scenario suites. Specifically, techniques such as Control Barrier Functions (CBFs), reachability analysis, and safety shields can be employed to rigorously restrict policy actions during execution. Concurrently, uncertainty estimation can be leveraged to identify out-of-distribution states that necessitate conservative behavior or fallback control mechanisms. To establish robust benchmarks, safety evaluations must move beyond average rewards and collision rates, incorporating interpretable safety margins, rule violation metrics, intervention frequencies, and systematic performance across long-tail scenarios.

5.2 Sim2Real and world model generalization

Sim2Real is a critical challenge in deploying RL for autonomous driving. Although many methods achieve strong results in simulation, their performance often drops in real-world driving because the learned policy depends on the fidelity of the training environment and the rollout model used during learning. This issue has become more important with the recent interest in world-model-based autonomous driving, where planning and policy improvement rely on learned latent dynamics over long horizons (Wang et al., 2025; Xiaofeng Wang et al, 2024; Yuqi Wang et al, 2024). The key difficulty lies not only in the domain gap between simulation and reality, but also in the error accumulation that appears when a learned world model is used in closed loop.

5.2.1 Existing solutions and limitations

Existing studies have explored system identification, domain randomization, adversarial Sim2Real transfer, DAGGER-style data aggregation, and data augmentation to improve transfer robustness (X. He et al, 2022; Ljung, 1998; Ross et al, 2011; Tobin et al, 2017). These methods can reduce the reality gap to some extent, but their performance still depends heavily on how well the training distribution covers real deployment conditions. For this reason, Sim2Real should be discussed together with world model generalization, rare-event coverage, and evaluation protocols rather than treated as an isolated simulator design problem.

Simulators inherently simplify vehicle dynamics, sensor responses, environmental conditions (e.g., weather and road-surface variations), and traffic-agent behavior (Chen et al., 2025). Even with careful calibration, it is practically impossible to fully eliminate discrepancies caused by unmodeled physical factors such as temperature fluctuations, tire wear, payload variations, and sensor aging. Furthermore, perception modules are highly susceptible to differences in illumination, atmospheric conditions, and sensor-noise profiles.

Enhancing training diversity via domain randomization, adversarial perturbations, or DAgger-style online aggregation, is widely used to bridge the sim-to-real transfer gap. Nevertheless, these methods depend on predefined perturbation spaces and limited real-world feedback. Crucially, they fail to capture complex surrounding-agent dynamics in dense traffic, where minor deviations in background-vehicle behavior can amplify into major planning failures. Interaction modeling is thus an inseparable dimension of the transfer problem.

A learned world model may still falter despite the apparent realism of the simulator. Although world models are advantageous for improving sample efficiency and facilitating planning within latent spaces, the quality of their generated rollouts inherently relies on generalization beyond the training distribution (Gao et al., 2024; Xiaofeng Wang et al., 2024; Yuqi Wang et al., 2024). Furthermore, during recursive querying, minor single-step errors rapidly accumulate, leading to severe distortions in future scene states. Ultimately, these compounded errors can inject bias into policy updates or result in overly optimistic planning.

To mitigate these issues, prior works have employed uncertainty-aware ensembles, latent-dynamics regularization, synthetic-rollout filtering, and hybrid training pipelines blending real and simulated experiences (Diehl et al., 2023; Wang et al., 2025; J. Wu et al., 2022). While these methods can identify unreliable rollouts and reduce extrapolation errors, closed-loop decision remains highly vulnerable to rare interactions. Furthermore, long-horizon predictions often degrade under safety-critical conditions. Consequently, high open-loop predictive accuracy in world models is insufficient to guarantee reliable closed-loop policy improvements.

Moreover, the ultimate deployment risk is often governed by safety-critical events—such as unusual cut-ins, multi-agent deadlocks, extreme weather, and sensor degradation (Liu and Feng, 2024). While simulation and data augmentation can expand scenario coverage, they fail to capture the full spectrum of the real-world long tail. Thus, a policy demonstrating excellent average-case performance may still exhibit critical vulnerabilities during rare, out-of-distribution events.

Future research should seek to combine mixed simulation and real-world training with targeted scenario generation and counterfactual robustness analysis, thereby testing policies against plausible yet underrepresented disturbances (Foerster et al., 2018). Furthermore, within world-model architectures, it is imperative that uncertainty estimates actively guide online planning and policy optimization, rather than serving only as offline diagnostics (Diehl et al., 2023; J. Wu et al., 2022).

Concurrently, existing Sim2Real literature often use different evaluation protocols. While some studies report open-loop reconstruction or prediction metrics, others rely on non-reactive planning scores or fully interactive closed-loop driving benchmarks (Dosovitskiy et al., 2017; D. Lee et al., 2024; Yuqi Wang et al., 2024). Because these metrics expose fundamentally distinct failure modes, they must be interpreted independently when assessing readiness for real-world deployment. Crucially, a framework may exhibit superior open-loop accuracy yet fail catastrophically in closed-loop execution if compounding errors, multi-agent interactions, and out-of-distribution recovery behaviors are not systematically evaluated.

5.2.2 Future directions

Therefore, comparing Sim2Real claims without disclosing evaluation protocols is fundamentally uninformative. Reliable future evaluations must unify open-loop realism checks, interactive closed-loop tests, and deployment-grade safety criteria. Only through such comprehensive standards can simulator fidelity, world-model quality, and policy transfer be rigorously and meaningfully benchmarked.

5.3 Data and sample efficiency

The third challenge is data and sample efficiency. Real driving data are expensive, privacy-sensitive, geographically biased, and sparse in long-tail safety-critical scenarios, while RL usually requires far more samples than human learning. This data bottleneck directly limits both end-to-end RL and RL fine-tuning for autonomous driving, because policies must experience enough diverse interactions to learn robust recovery behavior without relying on unsafe real-world exploration.

5.3.1Existing solutions and limitations

Existing methods improve sample efficiency through four main routes: offline RL, imitation learning, data augmentation, and synthetic data generation. Offline RL methods such as Implicit Q-Learning (IQL) and Twin Delayed Deep Deterministic Policy Gradient with Behavior Cloning (TD3+BC) learn from logged driving data without online exploration, making them attractive for safety-critical AD. Imitation learning uses expert demonstrations to warm-start policies before RL fine-tuning. Data augmentation improves robustness by perturbing visual, sensor, or scene features. Synthetic data generation and world-model rollouts can create additional scenarios for rare-event training.

However, out-of-distribution actions must be constrained while adapting Offline RL, otherwise the policy will tend to be overly conservative because of underestimated values, thus affect efficiency. Dataset bias will be introduced to imitation learning during training process, and the learned policy may fail to generalize when encountering long-tail scenarios that are underrepresented in expert data. Synthetic data generation is dependent on the quality, distribution, and physical fidelity. Low-quality simulation will introduce simulator-specific artifacts. Data flywheels can further improve data coverage by mining failure cases and incorporating them into model training, but they may still rely on data curation, scenario annotation, human review, and safety validation, limiting their ability to support a fully automated and scalable training process.

5.3.2 Future directions

The combination of active learning, generative simulation, and automated data flywheels is a compromising direction for future studies. Active learning selects typical and safety-critical scenarios for annotation and simulation, therefore increase data efficiency. Generative simulation is capable of producing rare and multi-modal traffic interactions, guided by real logs and long-tail scenario templates. Last but not least, automated data flywheels bridge failure mining, scenario generation, retrained, closed-loop validation, and deployment monitoring, avoiding the burden of manual annotation. As for RL, feasible directions include uncertainty-aware offline RL, policy pre-training from large-scale driving logs, and selective online fine-tuning in validated simulators before real-road shadow testing.

5.4 Computational efficiency of neural network deployment

Overall model size, power consumption, heat dissipation, memory capacity, and computing latency are the main bottlenecks of deploying neural-network and Deep-RL-based autonomous driving algorithms on vehicle-side hardware. These constraints become even more challenging for end-to-end policies, occupancy networks, 3D reconstruction modules, world models, and VLA systems, which require multi-modal models to process continuous sensor streams and generate control actions within fixed real-time deadlines. The software-hardware co-design pipeline is demonstrated in Figure 1. Deployment performance depends on model design, workload reduction, hardware mapping, memory scheduling, compiler support, and system validation.

Existing solutions include reducing computational workload through quantization, pruning, low-rank factorization, efficient convolution, and lightweight architecture design as shown in Figure 1. Quantization limits the bit width of weights and activations, hence lowering memory bandwidth and computational cost. Pruning eliminates redundant parameters or channels. Depth-wise separable convolutions further reduce arithmetic cost.

These approaches correspond to the algorithm-side optimization stage in the figure and can make DRL policies and perception backbones more deployable on embedded automotive hardware.

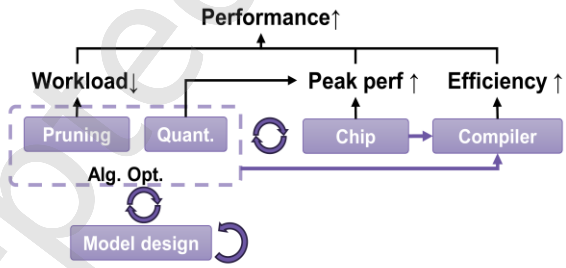

Fig. 4: Software–hardware co-design flow for efficient neural network deployment in autonomous driving

A second route improves peak computing performance and deployment efficiency through specialized accelerators, parallel hardware, memory hierarchy optimization, and model-specific scheduling. This corresponds to the hardware and compiler stages in Figure 1, where the objective is to match neural-network workloads to the available computing units and memory hierarchy. Efficient perception is especially important because LiDAR point clouds, camera images, BEV representations, and occupancy features must be processed with low latency before policy inference. Recent perception-efficiency methods include BDC-Occ (Zhang et al, 2024), FastOcc (Hou et al, 2024), and SparseOcc (Tang et al, 2024) for occupancy estimation; BiPointNet (Qin et al, 2021) and BSC-Net (Xu et al, 2023) for efficient point-cloud processing; and RT-NeRF (Chaojian Li et al, 2022), Fusion-3D (Sixu Li et al, 2024), and GSCore (Junseo Lee et al, 2024) for efficient 3D reconstruction or neural rendering. Compiler-level optimizations such as operator fusion, computation-graph rewriting, and memory reuse can further boost runtime efficiency. For instance, DNNVM (Xing et al, 2019) adapts software-stack optimization.

The trade-off between efficiency and accuracy, robustness, or model flexibility is another potential limitation. Lightweight policies obtained by compressing models may be vulnerable to distribution shift and lose robustness while encountering long-tail scenarios. VLA-based models introduce another deployment bottleneck: language reasoning and action generation require large parameter counts, long context windows, and sequential token decoding, all of which conflict with real-time closed-loop control. Developing hardware-aware RL training algorithms, latency-aware reward objectives, compact world models, efficient VLA architectures, and dual-system designs, in which large models provide semantic guidance while smaller verified controllers handle high-frequency control, remain to be addressed in future work.

5.5 Embodied intelligence and VLA Alignment

In autonomous driving tasks, language understanding can be combined with vehicle dynamics, temporal interaction and real-time control which must remain consistent during actioin generation. Foundation models of autonomous driving mainly relies on internet-scale text and image datasets for pretraining. Therefore, pretrained models may possess broad semantic knowledge, they still exhibit notable limitations in vehicle control and traffic interaction. The key is that errors tolerable in text generation may turn hazardous once translated into autonomous driving control commands such as steering and braking.

5.5.1 Existing solutions and limitations

Currently, there are four main VLA alignment methods including action tokenization, physical-constraint injection, RLHF for driving and chain-of-thought (CoT) reasoning for planning. Action tokenization, commonly realized via Vector Quantized-Variational Autoencoders (VQ-VAE), encodes continuous driving actions into discrete tokens for prediction by the language model. Physical-constraint injection introduces vehicle kinematics, traffic rules, and safety constraints via prompts, auxiliary losses, or post-processing. RLHF-style optimization leverages human preferences to build driving-specific reward models. CoT reasoning prompts the model to output intermediate decisions before generating an action..

However, these methods still have some limitations in closed-loop driving. For example, sequential CoT reasoning tokens increase inference latency and may conflict with vehicle control update frequencies. Soft prompts cannot enforce hard constraints such as friction limits, collision-avoidance envelopes, or actuator saturation. Discretizing continuous trajectories can reduce action precision. Moreover, standardized embodied-alignment benchmarks are still required to jointly evaluate semantic correctness and physical execution safety; text accuracy and open-loop planning metrics are insufficient on their own.

5.5.2 Future directions

Focusing on driving-specific foundation-model pretraining, physics-aware reward models, and hierarchical alignment in the future may be helpful. Pretraining data should integrate camera, LiDAR, maps, ego-motion, traffic-agent behavior, language annotations, and action trajectories. Reward models should penalize some rule violations like kinematically infeasible trajectories, insufficient TTC margins and comfort violations during RL fine-tuning. Hierarchical alignment should separate low-speed semantic reasoning from fast reactive control. A VLA model should infer intent, risk, and maneuver-level goals, while a lightweight planner or controller should enforce real-time safety constraints.

5.6 Evaluation and deployment readiness

Many benchmarks adopt open-loop data, whereas autonomous driving is widely considered as an interactive closed-loop process. A well-trained policy can perform well on logged data yet fail in real urban driving because of severe time delay. This gap could influence research conclusions, technology roadmap decisions, and regulatory assessment, since paper-reported scores may not reflect vehicle reliability.

5.2.1 Existing evaluation paradigms and limitations

Evaluation is performed via open-loop datasets, closed-loop simulation, and real-road testing. Datasets including nuScenes and the Waymo Open Dataset provide naturalistic driving logs for perception, prediction, and planning. Simulators such as CARLA and nuPlan enable the ego policy to interact with the environment, supporting sequential decision-making and safety evaluation. Real-road testing and accumulated driving mileage expose systems to sensor noise, traffic behavior, and operational constraints that simulations may omit.

Each evaluation paradigm mentioned above has limitations. Open-loop evaluation cannot capture feedback, recovery behavior, or reactions of other agents to the ego vehicle. Closed-loop simulation suffers from Sim2Real gaps, limited scenario diversity, and imperfect sensor or agent models. Real-road testing is costly, hard to reproduce, and inefficient for rare hazardous events. Moreover, the field still has no agreed benchmark for safety-critical driving scenarios. Such a benchmark should clarify which failure modes need examination, manage the hardness of test cases, and set rules to simultaneously report metrics covering safety, ride comfort, traffic-rule compliance and operational efficiency.

5.2.2 Future directions

Future evaluation can use adversarial scenario generation, world-model-based closed-loop testing, interpretable safety metrics, and real-road shadow testing. Adversarial generation can search for rare interactions that expose policy weaknesses. World-model evaluation can turn logged data into counterfactual closed-loop rollouts when model uncertainty is reported. In addition to route completion and average displacement error, safety reports should include collision risk, time-to-collision, rule violations, intervention rate, comfort, reasoning-action consistency, and performance across safety-critical scenario families. Deployment readiness should be based on consistent evidence from open-loop datasets, closed-loop simulation, reconstructed scenarios, and controlled real-road validation.

6. Conclusion

This survey organizes the literature by task scope, action-space structure, benchmark protocol, and deployment maturity. RL has produced strong results in several narrow or simulator-based settings, but evidence for reliable production deployment remains limited. The main obstacles are safety constraints, Sim2Real fidelity, inconsistent evaluation, and the computational cost of edge deployment. Future systems may combine RL with imitation learning, world models, foundation-model reasoning, and explicit planning or safety modules. The appropriate combination depends on the task and on the evidence available for the target deployment setting.

Replication and Data Sharing

This article is a review paper and does not report any newly generated datasets, source code, or experimental benchmarks. All analyses, discussions, and conclusions are derived from previously published studies cited in the References. Therefore, no additional data or code are associated with this work.

Reference

Achiam, J., Held, D., Tamar, A., Abbeel, P., 2017. Constrained policy optimization. In: International Conference on Machine Learning. PMLR, 22–31.

Alsaleh, R., Sayed, T., 2021. Markov-game modeling of cyclist-pedestrian interactions in shared spaces: A multi-agent adversarial inverse reinforcement learning approach. Transportation research part C: emerging technologies, 128, 103191.

An, G., Moon, S., Kim, J., Song, H.O., 2021. Uncertainty-based offline reinforcement learning with diversified q-ensemble. In: NeurIPS.

Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., et al., 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 3674–3683.

Aradi, S., 2020. Survey of deep reinforcement learning for motion planning of autonomous vehicles. IEEE Transactions on Intelligent Transportation Systems, 23(2), 740–759.

Arai, H., Miwa, K., Sasaki, K., Watanabe, K., Yamaguchi, Y., Aoki, S., et al., 2025. Covla: Comprehensive vision-language-action dataset for autonomous driving. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). IEEE, 1933–1943.

Ashraf, N.M., Mostafa, R.R., Sakr, R.H., Rashad, M., 2021. Optimizing hyperparameters of deep reinforcement learning for autonomous driving based on whale optimization algorithm. Plos one, 16(6), e0252754.

Autoware, 2024. https://autoware.org/ (accessed 2024-12-22).

Azadani, M.N., Boukerche, A., 2023. A context-aware path forecasting method for connected autonomous vehicles. In: ICC 2023-IEEE International Conference on Communications. IEEE, 247–252.

Azadani, M.N., Boukerche, A., 2024. Hierarchical transformers for motion forecasting based on inverse reinforcement learning. IEEE Transactions on Vehicular Technology.

Baidu apollo, 2024. https://apollo.baidu.com/ (accessed 2024-12-22).

Bautista-Montesano, R., Galluzzi, R., Ruan, K., Fu, Y., Di, X., 2022. Autonomous navigation at unsignalized intersections: A coupled reinforcement learning and model predictive control approach. Transportation research part C: emerging technologies, 139, 103662.

Bhattacharyya, R., Wulfe, B., Phillips, D.J., Kuefler, A., Morton, J., Senanayake, R., et al., 2022. Modeling human driving behavior through generative adversarial imitation learning. IEEE Transactions on Intelligent Transportation Systems, 24(3), 2874–2887.

Brockman, G., 2016. Openai gym. arXiv preprint arXiv:1606.01540.

Cai, P., Wang, S., Wang, H., Liu, M., 2021. Carl-lead: Lidar-based end-to-end autonomous driving with contrastive deep reinforcement learning. arXiv preprint arXiv:2109.08473.

Cao, Z., Xu, S., Peng, H., Yang, D., Zidek, R., 2021. Confidence-aware reinforcement learning for self-driving cars. IEEE Transactions on Intelligent Transportation Systems, 23(7), 7419–7430.

Cao, Z., Jiang, K., Zhou, W., Xu, S., Peng, H., Yang, D., 2023. Continuous improvement of self-driving cars using dynamic confidence-aware reinforcement learning. Nature Machine Intelligence, 5(2), 145–158.

Chen, G., Zhao, X., Gao, Z., Hua, M., 2023. Dynamic drifting control for general path tracking of autonomous vehicles. IEEE Transactions on Intelligent Vehicles, 8(3), 2527–2537.

Chen, G., Gao, Z., Hua, M., Shuai, B., Gao, Z., 2024. Lane change trajectory prediction considering driving style uncertainty for autonomous vehicles. Mechanical Systems and Signal Processing, 206, 110854.

Chen, G., Hua, M., Liu, W., Wang, J., Song, S., Liu, C., et al., 2024. Planning and tracking control of full drive-by-wire electric vehicles in unstructured scenario. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 238(13), 3941–3956.

Chen, G., Yao, J., Gao, Z., Gao, Z., Hua, M., Zhao, Y., 2025. High precision data-mechanism-driven lateral velocity estimation using transfer learning in distributed vehicles. IEEE Transactions on Instrumentation and Measurement.

Chen, I.-M., Chan, C.-Y., 2021. Deep reinforcement learning based path tracking controller for autonomous vehicle. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 235(2-3), 541–551.

Chen, J., Li, S.E., Tomizuka, M., 2021. Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning. IEEE Transactions on Intelligent Transportation Systems, 23(6), 5068–5078.

Chi, H., Gao, H.-a., Liu, Z., Liu, J., Liu, C., Li, J., et al., 2025. Impromptu vla: Open weights and open data for driving vision-language-action models. Advances in Neural Information Processing Systems, 38.

Choi, S., Kim, J., Yeo, H., 2021. Trajgail: Generating urban vehicle trajectories using generative adversarial imitation learning. Transportation Research Part C: Emerging Technologies, 128, 103091.

Chu, Z., Wang, F., Lei, T., Luo, C., 2022. Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance. IEEE Transactions on Intelligent Vehicles, 8(1), 108–120.

Deisenroth, M., Rasmussen, C.E., 2011. Pilco: A model-based and data-efficient approach to policy search. In: ICML.

Deruyttere, T., Vandenhende, S., Grujicic, D., Van Gool, L., Moens, M.F., 2019. Talk2car: Taking control of your self-driving car. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), 2088–2098.

Diehl, C., Sievernich, T.S., Krüger, M., Hoffmann, F., Bertram, T., 2023. Uncertainty-aware model-based offline reinforcement learning for automated driving. IEEE Robotics and Automation Letters, 8(2), 1167–1174.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., 2017. Carla: An open urban driving simulator. In: Conference on robot learning. PMLR, 1–16.

Du, Y., Chen, J., Zhao, C., Liu, C., Liao, F., Chan, C.-Y., 2022. Comfortable and energy-efficient speed control of autonomous vehicles on rough pavements using deep reinforcement learning. Transportation Research Part C: Emerging Technologies, 134, 103489.

Duan, J., Kong, Y., Jiao, C., Guan, Y., Li, S.E., Chen, C., et al., 2024. Distributional soft actor-critic for decision-making in on-ramp merge scenarios. Automotive Innovation, 7(3), 403–417.

Fang, X., Zhang, Q., Gao, Y., Zhao, D., 2022. Offline reinforcement learning for autonomous driving with real world driving data. In: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC). IEEE, 3417–3422.

Feinberg, V., Wan, A., Stoica, I., Jordan, M.I., Gonzalez, J.E., Levine, S., 2018. Model-based value estimation for efficient model-free reinforcement learning. arXiv preprint arXiv:1803.00101.

Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., Whiteson, S., 2018. Counterfactual multi-agent policy gradients. In: AAAI.

Fu, X., Liu, R., Wang, X., Zhao, L., 2020. Decision-making for autonomous vehicles in emergency scenarios using deep reinforcement learning. IEEE Access, 8, 123456–123467.

Fu, Y., Li, C., Yu, F.R., Luan, T.H., Zhang, Y., 2022. A selective federated reinforcement learning strategy for autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 24(2), 1655–1668.

Ganai, M., Gong, Z., Yu, C., Herbert, S., Gao, S., 2024. Iterative reachability estimation for safe reinforcement learning. Advances in Neural Information Processing Systems, 36.

Gangopadhyay, B., Dasgupta, P., Dey, S., 2022. Safe and stable rl (s 2 rl) driving policies using control barrier and control lyapunov functions. IEEE Transactions on Intelligent Vehicles, 8(2), 1889–1899.

Gao, H., Chen, S., Jiang, B., Liao, B., Shi, Y., Guo, X., et al., 2025. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. arXiv preprint arXiv:2502.13144.

Gao, Z., Mu, Y., Chen, C., Duan, J., Luo, P., Lu, Y., et al., 2024. Enhance sample efficiency and robustness of end-to-end urban autonomous driving via semantic masked world model. IEEE Transactions on Intelligent Transportation Systems.

Guan, Y., Tang, L., Lyu, Y., Li, S.E., Yang, Y., Sheng, K., Zhang, B., Li, K., 2026. Enhanced integrated decision and control for high-level automated vehicles and its experiment verification. *IEEE Transactions on Automation Science and Engineering*.

Guan, Y., Ren, Y., Sun, Q., Li, S.E., Ma, H., Duan, J., et al., 2022. Integrated decision and control: Toward interpretable and computationally efficient driving intelligence. IEEE transactions on cybernetics, 53(2), 859–873.

Gulino, C., Fu, J., Luo, W., Tucker, G., Bronstein, E., Lu, Y., et al., 2024. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. Advances in Neural Information Processing Systems, 36.

Guo, M., Li, Z., Wang, Y., Chen, H., Zhang, W., 2021. Hybrid reinforcement learning-based eco-driving strategy for autonomous vehicles. arXiv preprint arXiv:2201.07833.

Hao, Y., Li, Z., Sun, L., Wang, W., Yi, N., Song, S., et al., 2025. Driveaction: A benchmark for exploring human-like driving decisions in vla models. arXiv preprint arXiv:2506.05667.

He, X., Yang, H., Hu, Z., Lv, C., 2022. Robust lane change decision making for autonomous vehicles: An observation adversarial reinforcement learning approach. IEEE Transactions on Intelligent Vehicles, 8(1), 184–193.

Ho, J., Ermon, S., 2016. Generative adversarial imitation learning. NeurIPS, 29.

Hoel, C.-J., Wolff, K., Laine, L., 2023. Ensemble quantile networks: Uncertainty-aware reinforcement learning with applications in autonomous driving. IEEE Transactions on intelligent transportation systems, 24(6), 6030–6041.

Hou, J., Li, X., Guan, W., Zhang, G., Feng, D., Du, Y., et al., 2024. Fastocc: Accelerating 3d occupancy prediction by fusing the 2d bird's-eye view and perspective view. arXiv preprint arXiv:2403.02710.

Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., et al., 2023. Gaia-1: A generative world model for autonomous driving. arXiv preprint arXiv:2309.17080.

Hua, M., Chen, G., Zhang, B., Huang, Y., 2019. A hierarchical energy efficiency optimization control strategy for distributed drive electric vehicles. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 233(3), 605–621.

Hua, M., Zhang, C., Zhang, F., Li, Z., Yu, X., Xu, H., et al., 2023. Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning. Applied Energy, 348, 121526.

Hua, M., Chen, D., Qi, X., Jiang, K., Liu, Z.E., Zhou, Q., et al., 2023. Multi-agent reinforcement learning for connected and automated vehicles control: Recent advancements and future prospects. arXiv preprint arXiv:2312.11084.

Hua, M., Shuai, B., Zhou, Q., Wang, J., He, Y., Xu, H., 2023. Recent progress in energy management of connected hybrid electric vehicles using reinforcement learning. arXiv preprint arXiv:2308.14602.

Hua, M., Chen, D., Jiang, K., Zhang, F., Wang, J., Wang, B., et al., 2024. Communication-efficient marl for platoon stability and energy-efficiency co-optimization in cooperative adaptive cruise control of cavs. IEEE Transactions on Vehicular Technology.

Huang, Z., Wu, J., Lv, C., 2021. Driving behavior modeling using naturalistic human driving data with inverse reinforcement learning. IEEE transactions on intelligent transportation systems, 23(8), 10239–10251.

Huang, Z., Liu, H., Wu, J., Lv, C., 2023. Conditional predictive behavior planning with inverse reinforcement learning for human-like autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 24(7), 7244–7258.

Huang, Z., Sheng, Z., Qu, Y., You, J., Chen, S., 2024. Vlm-rl: A unified vision language models and reinforcement learning framework for safe autonomous driving. arXiv preprint arXiv:2412.15544.

Hussein, A., Gaber, M.M., Elyan, E., Jayne, C., 2017. Imitation learning: A survey of learning methods. ACM Computing Surveys (CSUR), 50(2), 1–35.

Hwang, S., Lee, K., Jeon, H., Kum, D., 2022. Autonomous vehicle cut-in algorithm for lane-merging scenarios via policy-based reinforcement learning nested within finite-state machine. IEEE Transactions on Intelligent Transportation Systems, 23(10), 17594–17606.

iDLab, Tsinghua University, 2025. LasVSim: A Self-Developed Software for Large-Scale Autonomous Vehicles Simulation. https://github.com/ rl-lasvsim/openapi-sdk-python, compatible with C++/Python, CarSim, and TensorFlow. (accessed 2025-05-16).

Janner, M., Fu, J., Zhang, M., Levine, S., 2019. When to trust your model: Model-based policy optimization. In: NeurIPS.

Jia, X., Gao, Y., Chen, L., Yan, J., Liu, P.L., Li, H., 2023. Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 7953–7963.

Jia, X., Shao, Y., Yang, Z., Li, Q., Zhang, Z., Yan, J., 2026. Bench2drive-vl: Benchmarks for closed-loop autonomous driving with vision-language models. arXiv preprint arXiv:2604.01259.

Jiang, B., Chen, S., Zhang, Q., Liu, W., Wang, X., 2025. Alphadrive: Unleashing the power of vlms in autonomous driving via reinforcement learning and reasoning. arXiv preprint arXiv:2503.07608.

jidi.ai, 2025. Jidi. http://www.jidiai.cn/ (accessed 2025-01-16).

Kostrikov, I., Nair, A., Levine, S., 2022. Offline reinforcement learning with implicit q-learning. In: ICLR.

Krantz, J., Wijmans, E., Majumdar, A., Batra, D., Lee, S., 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In: European Conference on Computer Vision. Springer, 104–120.

Kumar, A., Zhou, A., Tucker, G., Levine, S., 2020. Conservative q-learning for offline reinforcement learning. In: NeurIPS.

Kuutti, S., Bowden, R., Jin, Y., Barber, P., Fallah, S., 2020. A survey of deep learning applications to autonomous vehicle control. IEEE Transactions on Intelligent Transportation Systems, 22(2), 712–733.

Li, S.E., 2023. *Reinforcement Learning for Sequential Decision and Optimal Control*. Springer.

Lee, D., Eom, C., Kwon, M., 2024. AD4RL: autonomous driving benchmarks for offline reinforcement learning with value-based dataset. In: ICRA.

Lee, J., Lee, S., Lee, J., Park, J., Sim, J., 2024. Gscore: Efficient radiance field rendering via architectural support for 3d gaussian splatting. In: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3, 497–511.

Leurent, E., 2018. Highwayenv. https://github.com/eleurent/highway-env (accessed 2024-12-31).

Levine, S., Kumar, A., Tucker, G., Fu, J., 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. arXiv preprint arXiv:2005.01643.

Li, C., Li, S., Zhao, Y., Zhu, W., Lin, Y., 2022. Rt-nerf: Real-time on-device neural radiance fields towards immersive ar/vr rendering. In: Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design, 1–9.

Li, G., Yang, Y., Li, S., Qu, X., Lyu, N., Li, S.E., 2022. Decision making of autonomous vehicles in lane change scenarios: Deep reinforcement learning approaches with risk awareness. Transportation research part C: emerging technologies, 134, 103452.

Li, Q., Peng, Z., Feng, L., Zhang, Q., Xue, Z., Zhou, B., 2022. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. IEEE transactions on pattern analysis and machine intelligence, 45(3), 3461–3475.

Li, Q., Jia, X., Wang, S., Yan, J., 2024. Think2drive: Efficient reinforcement learning by thinking with latent world model for autonomous driving (in carla-v2). In: European Conference on Computer Vision. Springer, 142–158.

Li, S., Zhao, Y., Li, C., Guo, B., Zhang, J., Zhu, W., et al., 2024. Fusion-3d: Integrated acceleration for instant 3d reconstruction and real-time rendering. In: 2024 57th IEEE/ACM International Symposium on Microarchitecture (MICRO). IEEE, 78–91.

Li, Y., Tian, M., Zhu, D., Zhu, J., Lin, Z., Xiong, Z., et al., 2025. Drive-r1: Bridging reasoning and planning in vlms for autonomous driving with reinforcement learning. arXiv preprint arXiv:2506.18234.

Li, Y., Xiong, K., Guo, X., Li, F., Yan, S., Xu, G., et al., 2025. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. arXiv preprint arXiv:2506.08052.

Li, Z., Nie, F., Sun, Q., Da, F., Zhao, H., 2024. Boosting offline reinforcement learning for autonomous driving with hierarchical latent skills. In: ICRA.

Lin, H., Ding, W., Liu, Z., Niu, Y., Zhu, J., Niu, Y., et al., 2024. Safety-aware causal representation for trustworthy offline reinforcement learning in autonomous driving. IEEE Robotics and Automation Letter, 9(5), 4639–4646.

Liu, H., Huang, Z., Wu, J., Lv, C., 2022. Improved deep reinforcement learning with expert demonstrations for urban autonomous driving. In: 2022 IEEE intelligent vehicles symposium (IV). IEEE, 921–928.

Liu, H., Kiumarsi, B., Kartal, Y., Koru, A.T., Modares, H., Lewis, F.L., 2023. Reinforcement learning applications in unmanned vehicle control: A comprehensive overview. Unmanned Systems, 11(01), 17–26.

Liu, H., Chen, Y., Wang, H., Yang, Z., Li, T., Zeng, J., et al., 2025. Fully sparse 3d occupancy prediction. In: European Conference on Computer Vision. Springer, 54–71.

Liu, H.X., Feng, S., 2024. Curse of rarity for autonomous vehicles. nature communications, 15(1), 4808.

Liu, W., Hua, M., Deng, Z., Meng, Z., Huang, Y., Hu, C., et al., 2023. A systematic survey of control techniques and applications in connected and automated vehicles. IEEE Internet of Things Journal, 10(24), 21892–21916.

Ljung, L., 1998. System identification. In: Signal analysis and prediction. Springer, 163–173.

Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., et al., 2018. Microscopic traffic simulation using sumo. In: 2018 21st international conference on intelligent transportation systems (ITSC). IEEE, 2575–2582.

Lu, Y., Fu, J., Tucker, G., Pan, X., Bronstein, E., Roelofs, R., et al., 2023. Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 7553–7560.

Ma, H., Chen, J., Eben, S., Lin, Z., Guan, Y., Ren, Y., et al., 2021. Model-based constrained reinforcement learning using generalized control barrier function. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 4552–4559.

Marvi, Z., Kiumarsi, B., 2021. Safe reinforcement learning: A control barrier function optimization approach. International Journal of Robust and Nonlinear Control, 31(6), 1923–1940.

Nan, J., Deng, W., Zhang, R., Wang, Y., Zhao, R., Ding, J., 2023. Interaction-aware planning with deep inverse reinforcement learning for human-like autonomous driving in merge scenarios. IEEE Transactions on Intelligent Vehicles, 9(1), 2714–2726.

OpenAI, 2025. Openai universe. https://openai.com/index/universe/ (accessed 2025-01-16).

Pomerleau, D.A., 1991. Efficient training of artificial neural networks for autonomous navigation. Neural computation, 3(1), 88–97.

Qin, H., Cai, Z., Zhang, M., Ding, Y., Zhao, H., Yi, S., et al., 2021. Bipointnet: Binary neural network for point clouds. https://arxiv.org/abs/2010.05501.

Radtke, H., Bey, H., Sackmann, M., Schön, T., 2023. Predicting driver behavior on the highway with multi-agent adversarial inverse reinforcement learning. In: 2023 IEEE Intelligent Vehicles Symposium (IV). IEEE, 1–8.

Rong, G., Shin, B.H., Tabatabaee, H., Lu, Q., Lemke, S., Možeiko, M., et al., 2020. Lgsvl simulator: A high fidelity simulator for autonomous driving. In: 2020 IEEE 23rd International conference on intelligent transportation systems (ITSC). IEEE, 1–6.

Ross, S., Bagnell, D., 2010. Efficient reductions for imitation learning. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 661–668.

Ross, S., Gordon, G., Bagnell, D., 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 627–635.

Sackmann, M., Bey, H., Hofmann, U., Thielecke, J., 2022. Modeling driver behavior using adversarial inverse reinforcement learning. In: 2022 IEEE Intelligent Vehicles Symposium (IV). IEEE, 1683–1690.

Shah, S., Dey, D., Lovett, C., Kapoor, A., 2018. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In: Field and Service Robotics: Results of the 11th International Conference. Springer, 621–635.

Shan, Y., Zheng, B., Chen, L., Chen, L., Chen, D., 2020. A reinforcement learning-based adaptive path tracking approach for autonomous driving. IEEE Transactions on Vehicular Technology, 69(10), 10581–10595.

Shao, H., Hu, Y., Wang, L., Waslander, S.L., Liu, Y., Li, H., 2024. Lmdrive: Closed-loop end-to-end driving with large language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

Shi, H., Zhou, Y., Wu, K., Wang, X., Lin, Y., Ran, B., 2021. Connected automated vehicle cooperative control with a deep reinforcement learning approach in a mixed traffic environment. Transportation Research Part C: Emerging Technologies, 133, 103421.

Siemens, 2024. Simcenter prescan software. https://plm.sw.siemens. com/en-US/simcenter/autonomous-vehicle-solutions/prescan (accessed 2025-01-01).

Simulation, M., 2024. Carsim. https://www.carsim.com (accessed 2025-01-01).

Stachowicz, K., Shah, D., Bhorkar, A., Kostrikov, I., Levine, S., 2023. Fastrlap: A system for learning high-speed driving via deep rl and autonomous practicing. In: Conference on Robot Learning. PMLR, 3100–3111.

Sutton, R.S., 1990. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In: Machine learning proceedings 1990, 216–224.

Sutton, R.S., Barto, A.G., 1998. Reinforcement learning: An introduction. MIT Press.

Tang, P., Wang, Z., Wang, G., Zheng, J., Ren, X., Feng, B., et al., 2024. Sparseocc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 15035–15044.

Tang, X., Chen, J., Yang, K., Toyoda, M., Liu, T., Hu, X., 2022. Visual detection and deep reinforcement learning-based car following and energy management for hybrid electric vehicles. IEEE Transactions on Transportation Electrification, 8(2), 2501–2515.

Technologies, U., 2025. Unity ml-agents toolkit. https://github.com/ Unity-Technologies/ml-agents?tab=readme-ov-file (accessed 2025-01-16).

Terry, J., Black, B., Grammel, N., Jayakumar, M., Hari, A., Sullivan, R., et al., 2021. Pettingzoo: Gym for multi-agent reinforcement learning. Advances in Neural Information Processing Systems, 34, 15032–15043.

Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P., 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, 23–30.

Toromanoff, M., Wirbel, E., Moutarde, F., 2020. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 7153–7162.

Wang, B., Tao, L., Shuai, B., Gao, J., Zhao, W., Xiong, W., Sheng, K., Zhang, B., Guan, Y., Li, S.E., 2026. FAST: A framework for aligned sampling and training in parallel reinforcement learning for autonomous driving. *arXiv preprint arXiv:2606.21587*

Wang, H., Ye, X., Tao, F., Pan, C., Mallik, A., Yaman, B., et al., 2025. Adaptive world model based planning for autonomous driving. In: International Conference on Learning Representations (ICLR).

Wang, X., Zhang, J., Hou, D., Cheng, Y., 2023. Autonomous driving based on approximate safe action. IEEE Transactions on Intelligent Transportation Systems, 24(12), 14320–14328.

Wang, X., Zhu, Z., Huang, G., Chen, X., Zhu, J., Lu, J., 2024. Drivedreamer: Towards real-world-driven world models for autonomous driving. In: Computer Vision–ECCV 2024. Springer Nature Switzerland.

Wang, Y., Wan, S., Li, Q., Niu, Y., Ma, F., 2023. Modeling crossing behaviors of e-bikes at intersection with deep maximum entropy inverse reinforcement learning using drone-based video data. IEEE Transactions on Intelligent Transportation Systems, 24(6), 6350–6361.

Wang, Y., Niu, Y., Zhu, W., Chen, W., Li, Q., Wang, T., 2023. Predicting pedestrian crossing behavior at unsignalized mid-block crosswalks using maximum entropy deep inverse reinforcement learning. IEEE Transactions on Intelligent Transportation Systems.

Wang, Y., He, J., Fan, L., Li, H., Chen, Y., Zhang, Z., 2024. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14749–14759.

Watkins, C.J.C.H., 1989. Learning from delayed rewards.

Wen, X., Jian, S., He, D., 2023. Modeling the effects of autonomous vehicles on human driver car-following behaviors using inverse reinforcement learning. IEEE Transactions on Intelligent Transportation Systems.

Williams, R.J., 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning, 8, 229–256.

Wu, J., Huang, Z., Lv, C., 2022. Uncertainty-aware model-based reinforcement learning: Methodology and application in autonomous driving. IEEE Transactions on Intelligent Vehicles, 8(1), 194–203.

Wu, J., Huang, C., Huang, H., Lv, C., Wang, Y., Wang, F.-Y., 2024. Recent advances in reinforcement learning-based autonomous driving behavior planning: A survey. Transportation Research Part C: Emerging Technologies, 164, 104654.

Wu, P., Jia, X., Chen, L., Yan, J., Li, H., Qiao, Y., 2022. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. Advances in Neural Information Processing Systems, 35, 6119–6132.

Xing, Y., Liang, S., Sui, L., Jia, X., Qiu, J., Liu, X., et al., 2019. Dnnvm: End-to-end compiler leveraging heterogeneous optimizations on fpga-based cnn accelerators. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 39(10), 2668–2681.

Xu, X., Wang, Z., Zhou, J., Lu, J., 2023. Binarizing sparse convolutional networks for efficient point cloud analysis. https://arxiv.org/abs/2303.15493.

Yang, Y., Jiang, Y., Liu, Y., Chen, J., Li, S.E., 2023. Model-free safe reinforcement learning through neural barrier certificate. IEEE Robotics and Automation Letters, 8(3), 1295–1302.

Yang, Y., Zhang, Y., Zou, W., Chen, J., Yin, Y., Li, S.E., 2024. Synthesizing control barrier functions with feasible region iteration for safe reinforcement learning. IEEE Transactions on Automatic Control, 69(4), 2713–2720.

Ye, J., Wang, L., Zhang, H., Liu, X., 2020. Automated lane change strategy using proximal policy optimization-based deep reinforcement learning. arXiv preprint arXiv:2002.02667.

Yu, T., Kumar, A., Rafailov, R., Rajeswaran, A., Levine, S., Finn, C., 2021. COMBO: conservative offline model-based policy optimization. In: NeurIPS.

Yu, Z., Shu, C., Deng, J., Lu, K., Liu, Z., Yu, J., et al., 2023. Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin. arXiv preprint arXiv:2311.12058.

Yue, X., Shi, H., Zhou, Y., Li, Z., 2024. Hybrid car following control for cavs: Integrating linear feedback and deep reinforcement learning to stabilize mixed traffic. Transportation Research Part C: Emerging Technologies, 167, 104773.

Yurtsever, E., Lambert, J., Carballo, A., Takeda, K., 2020. A survey of autonomous driving: Common practices and emerging technologies. IEEE access, 8, 58443–58469.

Zhang, T., Jin, T., Wang, L., Liu, J., Liang, S., Zhang, M., et al., 2025. Bench2advlm: a closed-loop benchmark for vision-

language models in autonomous driving. arXiv preprint arXiv:2508.02028.
Zhang, Z., Liniger, A., Dai, D., Yu, F., Van Gool, L., 2021. End-to-end urban driving by imitating a reinforcement learning coach. In: Proceedings of the IEEE/CVF international conference on computer vision, 15222–15232.
Zhang, Z., Xu, Z., Yang, W., Liao, Q., Xue, J.-H., 2024. Bdc-occ: Binarized deep convolution unit for binarized occupancy network. arXiv preprint arXiv:2405.17037.
Zhao, R., Li, Y., Fan, Y., Gao, F., Tsukada, M., Gao, Z., 2024. A survey on recent advancements in autonomous driving using deep reinforcement learning: Applications, challenges, and solutions. IEEE Transactions on Intelligent Transportation Systems.
Zhao, S., Chen, G., Hua, M., Zong, C., 2019. An identification algorithm of driver steering characteristics based on backpropagation neural network. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 233(9), 2333–2342.
Zhou, M., Luo, J., Villella, J., Yang, Y., Rusu, D., Miao, J., et al., 2021. Smarts: An open-source scalable multi-agent rl training school for autonomous driving. In: Conference on robot learning. PMLR, 264–285.
Zhou, Z., Cai, T., Zhao, S.Z., Zhang, Y., Huang, Z., Zhou, B., et al., 2025. Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. arXiv preprint arXiv:2506.13757.

Author biography

Bin Shuai received his Ph.D. in mechanical engineering from the University of Birmingham, Birmingham, UK, in 2021. He is now working as a postdoctoral researcher with Tsinghua-Shuimu Fellowship in the School of Vehicle and Mobility, Tsinghua University, Beijing, China. His research interests include intelligent vehicle control, reinforcement learning, system modeling, and energy management.

Min Hua received the Ph.D. degree in Mechanical Engineering from the University of Birmingham, Birmingham, U.K., in 2025. Her research interests include autonomous driving, connected and automated vehicles, vehicle dynamics and control, reinforcement learning, and intelligent transportation systems.

Letian Tao received the B.S. degree in Vehicle Engineering from the School of Vehicle and Mobility, Tsinghua University, Beijing, in 2022. He is currently pursuing the Ph.D. degree in Mechanical Engineering at the School of Vehicle and Mobility, Tsinghua University, Beijing. His research interests focus on reinforcement learning and optimal control, particularly their applications in autonomous driving.

Zhilong Zheng received his B.S. degrees from the School of Vehicle and Mobility, Tsinghua University, Beijing, China, in 2022. He is currently pursuing his Ph.D. degree in the School of Vehicle and Mobility, Tsinghua University, Beijing, China. His research interests include forgetting-free fine-tuning, decision and control of autonomous vehicle, and safe reinforcement learning.

Yujie Yang received his Ph.D. in Mechanical Engineering from the School of Vehicle and Mobility at Tsinghua University, Beijing, China in 2026, where he also earned his B.S. degree in Automotive Engineering in 2021. He is an incoming Research Assistant Professor in the Department of Electrical and Computer Engineering at the University of Hong Kong. His research interests include safe reinforcement learning and decision-making and control of autonomous vehicles.

Yang Guan received his Ph.D. degree from the School of Vehicle and Mobility, Tsinghua University in 2022. He is currently an assistant researcher in the School of Vehicle and Mobility, Tsinghua University, China. Before joining Tsinghua University, he worked as a research scientist at Meituan Autonomous Driving Department. His active research interests include data-driven planning, end-to-end autonomous driving, reinforcement learning and neural networks.

Lei He received his B.S. in Beijing University of Aeronautics and Astronautics, China, in 2013, and the Ph.D. in the National Laboratory of Pattern Recognition, Chinese Academy of Sciences, in 2018. From then to 2021, Dr. He served as a postdoctoral fellow in the Department of Automation, Tsinghua University, Beijing, China. He worked as the research leader of the Autonomous Driving algorithm at Baidu and NIO from 2018 to 2023. He is a Research Scientist in automotive engineering with Tsinghua University. His research interests include Perception, SLAM, Planning, and Control.

Jingliang Duan received his Ph.D. degree in the School of Vehicle and Mobility, Tsinghua University, China, in 2021. He worked as a research fellow in the Department of Electrical and Computer Engineering, National University of Singapore, from 2021 to 2022. He is currently an associate professor in the School of Mechanical Engineering, University of Science and Technology Beijing, China. His research interests include reinforcement learning, optimal control, and self-driving decision-making.

Jiaxin Gao received his B.S. and Ph.D. degrees from the University of Science & Technology Beijing in 2017 and 2023, respectively. He conducted his postdoctoral research at the School of Vehicle and Mobility, Tsinghua University from July 2023 to July 2025. He is currently an Assistant Researcher at the State Key Laboratory of Intelligent Green Vehicle and Mobility, Tsinghua University. His research interests focus on autonomous driving, reinforcement learning, decision and control for autonomous vehicles, perceptual data generation, and simulation data for autonomous driving.

Shuo Feng received the bachelor's and Ph.D. degrees from the Department of Automation, Tsinghua University, China, in 2014 and 2019, respectively. He was a Post-Doctoral Research Fellow with the Department of Civil and Environmental Engineering and also an Assistant Research Scientist with the University of Michigan Transportation Research Institute (UMTRI), University of Michigan, Ann Arbor. He is currently an Associate Professor with the Department of Automation, Tsinghua University. His research interests include the development and validation of safety-critical machine learning, particularly for connected and automated vehicles.

Xianyuan Zhan is an ajssociate professor at the Institute for AI Industry Research (AIR), Tsinghua University. He received a dual Master's degree in Computer Science and Transportation Engineering, and a PhD degree in Transportation Engineering from Purdue University. Before joining AIR, Dr. Zhan was a data scientist at JD Technology and also a researcher at Microsoft Research Asia (MSRA). His research directions include data-driven decision-making, embodied AI, and autonomous driving.

Jincheng Yu received his B.S. and Ph.D. degrees from Tsinghua University, Beijing, in 2016 and 2021. He is a Research Associate Professor of EE department of Tsinghua University. His research interests include DSA accelerator and AI robot ap plication. He has published over 30 papers including ICRA, RAL and IROS. He is the reviewer of ICRA, RAL and IROS, and also serves as Assistant Editor for IROS and RAL

Haifeng Zhang is an Associate Professor at the Institute of Automation, Chinese Academy of Sciences, leading the Collective Decision-Intelligence research group. He received his B.S. and Ph.D. from Peking University, and completed his postdoctoral research at University College London. His research focuses on multi-agent reinforcement learning, game intelligence, agent evaluation and large-model agents.

Yang Yu received his Ph.D. degree in Computer Science from Nanjing University in 2011 (supervisor Prof. Zhi-Hua Zhou), and then joined the LAMDA Group (LAMDA Publications), in the Department of Computer Science and Technology of Nanjing University as an Assistant Researcher from 2011, and as an Associate Professor from 2014. He joined the School of Artificial Intelligence of Nanjing University as a Professor from 2019. His research interest is in machine learning, a subfield of artificial intelligence. Currently, he is working on reinforcement learning in various aspects, including optimization, representation, transfer, etc.

Shengbo Eben Li received his M.S. and Ph.D. degrees from Tsinghua University in 2006 and 2009. Before joining Tsinghua University, he has worked at Stanford University, University of Michigan, and UC Berkeley. His active research interests include intelligent vehicles and driver assistance, deep reinforcement learning, optimal control and estimation, etc. He is the author of over 200 peer-reviewed journal/conference papers, and co-inventor of over 50 patents. Dr. Li has received over 20 prestigious awards, including Youth Sci. & Tech Award of Ministry of Education (annually 10 receivers in China), Natural Science Award of Chinese Association of Automation (First level), National Award for Progress in Sci & Tech of China, and best (student) paper awards of IET ITS, IEEE ITS, IEEE ICUS, CVCI, etc.

Author contributions

Bin Shuai: Conceptualization, Investigation, Writing – original draft (Abstract, Section 1.2, and Section 6), Writing – review & editing, Project administration. **Min Hua**: Investigation, Writing – original draft, Writing – review & editing. **Letian Tao**: Investigation, Writing – original draft, Writing – review & editing. **Zhilong Zheng**: Investigation, Writing – original draft (Section 1.1, Background), Writing – review & editing. **Yujie Yang**: Investigation, Writing – original draft, Writing–review & editing. **Yang Guan**: Investigation, Writing–review & editing. **Lei He**: Investigation, Writing–original draft, Writing–review & editing. **Jingliang Duan**: Investigation, Writing–review & editing. **Jiaxin Gao**: Investigation, Writing–review & editing. **Shuo Feng**: Investigation, Writing–original draft, Writing – review & editing. **Xianyuan Zhan**: Investigation, Writing – original draft, Writing – review & editing. **Jincheng Yu**: Investigation, Writing – original draft, Writing–review & editing. **Yang Yu**: Investigation, Writing–original draft, Writing – review & editing. **Haifeng Zhang**: Investigation, Writing – original draft, Writing – review & editing. **Shengbo Eben Li**: Conceptualization, Supervision, Project administration, Writing – review & editing. All authors contributed to the review and revision of the manuscript and approved the final version for submission

Declaration of competing interest (mandatory)

The authors have no competing interests to declare that are relevant to the content of this article.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT (OpenAI) for language polishing, grammar correction, and assistance with manuscript formatting. After utilizing this tool, the authors thoroughly reviewed and edited the content as necessary and take full responsibility for the final content of the published article.