Model-Based Policy Adaptation for Closed-Loop End-to-End Autonomous Driving

Haohong Lin[1], Yunzhi Zhang[2], Wenhao Ding[3], Jiajun Wu[2], Ding Zhao[1]
[1]CMU, [2]Stanford, [3]NVIDIA
haohongl@andrew.cmu.edu

Abstract

End-to-end (E2E) autonomous driving models have demonstrated strong performance in open-loop evaluations but often suffer from cascading errors and poor generalization in closed-loop settings. To address this gap, we propose Model-based Policy Adaptation (MPA), a general framework that enhances the robustness and safety of pretrained E2E driving agents during deployment. MPA first generates diverse counterfactual trajectories using a geometry-consistent simulation engine, exposing the agent to scenarios beyond the original dataset. Based on this generated data, MPA trains a diffusion-based policy adapter to refine the base policy's predictions and a multi-step Q value model to evaluate long-term outcomes. At inference time, the adapter proposes multiple trajectory candidates, and the Q value model selects the one with the highest expected utility. Experiments on the nuScenes benchmark using a photorealistic closed-loop simulator demonstrate that MPA significantly improves performance across in-domain, out-of-domain, and safety-critical scenarios. We further investigate how the scale of counterfactual data and inference-time guidance strategies affect overall effectiveness.

1. Introduction

Driving foundation models have made impressive strides by integrating perception, prediction, and planning into a unified end-to-end (E2E) learning framework [6, 17, 18, 22]. Leveraging large-scale offline driving datasets, E2E models perform well under open-loop evaluation protocols, where the agent passively predicts future behaviors from offline recorded observation sequences. However, these models degrade in closed-loop environments, where minor deviations accumulate over time, leading to compounding errors, distribution shifts, and poor generalization to long-horizon scenarios. This performance gap reveals a core challenge: offline training based on empirical risk minimization does not align with the online objective of maximizing cumulative reward, as is illustrated in Figure 1.

To bridge this gap, recent efforts have turned to evaluating the closed-loop performance of E2E agents. Some open-loop such as NavSim proposes the approximate closed-loop evaluation with a Predictive Driver Model Score in an open-loop evaluation fashion. Other works introduced sensor simulation for the closed-loop evaluation, generating camera views based on diffusion models [13, 50], Neural Radiance Field (NeRF) [31, 33, 42, 51] or 3D Gaussian Splatting (3DGS) [14, 26, 46, 48, 56] that enable photorealistic rendering of novel viewpoints. These tools provide fine-grained control over agent interventions and visual realism, making them promising testbeds for studying failure modes and recovery strategies. Existing works such as VAD [22], VAD-v2 [5], and Hydra-MDP [28] design different scoring mechanisms to select the predicted motions for closed-loop control, yet these works either lack closed-loop evaluation results or are evaluated in a non-photorealistic simulator like CARLA [9]. Most recently, RAD [11] incorporates reinforcement learning and uses 3DGS for online rollouts and evaluation, while the training of PPO agents can be costly, and the value critic is left unused at inference time. Among all the prior attempts, none of the works have curated counterfactual data into consideration during the training phase.

Our goal in this paper is to adapt the pretrained open-loop E2E driving agents from the real domain to safe and generalizable closed-loop agents, with a 3DGS-based driving simulation data engine. We identify that the performance drop between the closed-loop and open-loop evaluations stems from two fundamental sources: (1) Observation mismatch — the shift between training-time sensor inputs and deployment-time closed-loop observations under perturbed behaviors from different data engines; (2) Objective mismatch — the absence of meaningful reward feedback during offline imitation learning, which limits long-term reasoning.

We conducted preliminary experiments to demonstrate that the first mismatch is actually minor in the open-loop evaluation. Then we propose a unified solution called **Model-Based Policy Adaptation (MPA)**, a general framework that directly addresses both mismatches by separating and targeting their root causes. We first use the pretrained policy as a reference policy to generate a counterfactual dataset using a high-fidelity 3DGS simulation engine. To mitigate

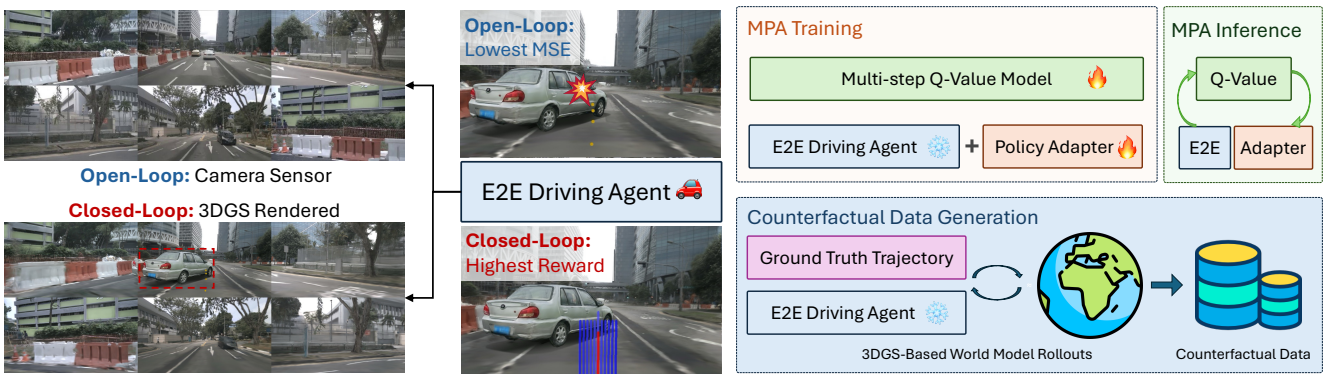

Figure 1. **Left:** Causes of closed-loop performance degradation in End-to-End driving, including observation and objective mismatches. **Right:** We propose counterfactual data generation to address the observation mismatch, and a model-based policy adaptation framework tackling the objective mismatch.

observation mismatch, we design a diffusion-based residual **policy adapter** that conditions on diverse, counterfactual trajectories. This exposes the policy to a broad range of behaviors and visual scenes beyond those seen in the offline dataset. To address objective mismatch, then learn a **Q-value model** from the same counterfactual data that captures long-horizon outcomes and enables value-based assessment beyond rule-based metrics. MPA uses both components at inference time: the policy adapter generates residual action proposals conditioned on the current observation, and the value model performs inference-time scaling to select the action with the highest expected utility.

Our Contribution. Our contributions are three-fold.

- We analyze the root causes of closed-loop performance degradation in E2E agents and assess the fidelity of 3DGS-based simulation for modeling observation and behavior shifts.
- We develop a counterfactual data curation pipeline using 3DGS rollouts and train the MPA with a diffusion-based policy adapter and reward model to address observation and reward mismatches, respectively.
- We demonstrate that inference-time scaling using the learned reward model significantly improves closed-loop performance on the nuScenes benchmark, particularly in safety-critical and out-of-domain scenarios.

2. Preliminary

2.1. Problem Formulation of E2E Driving

We formulate closed-loop end-to-end (E2E) driving as a Partially Observed Markov Decision Process (POMDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \mathcal{O}, \gamma, T)$, where $\mathcal{S}$ is the latent state space, $\mathcal{A}$ the action space, P the transition dynamics, R the reward function, $\mathcal{O}$ the observation space, γ the discount factor, and T the planning horizon. At each timestep, the agent receives an observation $o_t \in \mathcal{O}$ and outputs a trajectory action $a_t \in \mathcal{A}$. The environment evolves according to $P(s_{t+1}|s_t, a_t)$ and emits observations via $P_{\text{obs}}(o_t|s_t)$. In practice, the state s_t includes the ego vehicle's IMU status and surrounding road entities' past poses and motion intents. These are often only partially observable. The action a_t represents a sequence of future waypoints, which is translated into low-level throttle and steering control sequences via an LQR controller [27], following prior benchmarks [7, 25, 50, 56]. The observation o_t is captured by real sensors or rendered by a simulation engine during closed-loop evaluation. Notably, current open-loop E2E agents are trained using expert trajectories from a reference policy π_{ref}, inducing a state distribution $d^{\text{ref}}(s_t)$ and yielding a supervised model $\hat{\pi}_{\text{ref}}$. In contrast, closed-loop agents aim to maximize cumulative reward over time:

$$\text{Open-loop}: \ \hat{\pi}^* = \arg\min_{\pi} \sum_{t=1}^{T} \mathbb{E}_{(s_t,a_t)\sim\pi_{\text{ref}}} \|a_t - \pi(s_t)\|_2^2,$$

$$\text{Closed-loop}: \pi^* = \arg\max_{\pi} \sum_{t=1}^{T} \mathbb{E}_{s_t\sim P(s_{t-1},a_{t-1}),\ a_{t-1}\sim\pi(o_{t-1},s_{t-1}),\ o_{t-1}\sim P_{\text{obs}}(s_{t-1})} \left[r(s_t, a_t)\right], \tag{1}$$

This leads to an inherent objective mismatch: open-loop training minimizes imitation error under expert supervision, while closed-loop deployment optimizes long-horizon reward under evolving dynamics and partial observability, as is illustrated in Figure 1. Bridging this gap requires careful alignment of three components in equation (1): the transition model $P(s_{t+1}|s_t, a_t)$, which can be consistently approximated using vehicle dynamics; the observation model $P_{\text{obs}}(o_t|s_t)$, which may deviate from simulated sensors $\widehat{P}_{\text{obs}}$; and the reward function $r(s_t, a_t)$, which must be inferred from partial observations o_t using learned value models.

To address these mismatches, we employ a counterfactual data generation supported by 3DGS-based observation model $\widehat{P}_{\text{obs}}$, then design a policy adapter that transforms

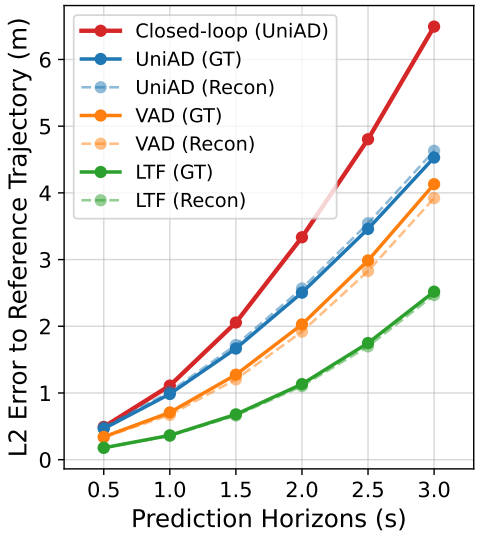

Figure 2. Comparison of average L2 error in the motion prediction under different prediction horizons.

the pretrained $\hat{\pi}_{\text{ref}}$ into a reward-aligned policy π^* under the guidance of a learned Q-value model, as outlined in Figure 1.

2.2. E2E Driving in the Closed-loop Simulation

A recent line of approach uses visual generative or reconstruction models for rendering photorealistic driving scenes from state parameterizations, utilizing the recent advancement in image diffusion modeling [50], neural field rendering [31], and 3D Gaussian Splatting [56]. This line of methods essentially learn to estimate $\hat{P}_{\text{obs}}(\cdot|s_t)$ in our POMDP formulation in Sec. 2.1. However, it is critical to verify that the visual quality of these model outputs remains close to real-world scenes for them to serve as valid proxies for our closed-loop evaluation.

We conducted the following preliminary experiments to study the fidelity of the closed-loop simulator and demonstrate the performance gap between open-loop and closed-loop evaluation. In Figure 2, we systematically study the difference in L2 error under open and closed-loop settings. We use 3DGS [56] to reconstruct the scenes from the nuScenes dataset and compare the performance difference using ground truth data in Figure 2. Among all three E2E policies, we see a very close open-loop performance in motion prediction. This confirms the fidelity of the 3DGS-based simulation in its reconstruction quality. Meanwhile, we also illustrate the L2 error based on UniAD's closed-loop rollout trajectory. As the prediction horizon grows, the prediction error becomes quite significant compared to the open-loop prediction. A non-ignorable L2 error in the short prediction horizon leads to out-of-distribution issues, resulting in compounding errors in the final prediction. Increasing the planning frequency cannot fundamentally close the gap between open-loop and closed-loop performance unless effective feedback guidance is provided to the agents.

3. Methodology

To bridge the observation and objective mismatches outlined in Sec. 2.1, we introduce Model-Based Policy Adaptation (MPA)—a unified framework for open-loop to closed-loop adaptation in end-to-end (E2E) autonomous driving. Figure 3 shows an overview. This section is organized from left to right along the pipeline: Section 3.1 describes our model-based counterfactual data synthesis to address distribution mismatch. Section 3.2 details the training of a diffusion-based policy adapter on the curated dataset. Section 3.3 presents the value model used to guide policy adaptation.

3.1. Counterfactual Data Generation

Algorithm 1: 3DGS-Based Counterfactual Generation

Input: Ref. dataset $\mathcal{D}^{\text{ref}}$, policy $\hat{\pi}_{\text{ref}}$, horizon T, thresholds δ, r_c
Output: Counterfactual dataset $\mathcal{D}^{\text{cf}}$
foreach *ref. traj* $(s_0^{ref}, \dots)$ *in* $\mathcal{D}^{ref}$ **do**
  $s_0 \leftarrow s_0^{\text{ref}}$
  for $t = 0$ **to** $T-1$ **do**
    $a_t \leftarrow \hat{\pi}_{\text{ref}}(s_{0:t})$
    if $dist(s_t, s_t^{ref}) > \delta$ ***or*** $r(s_t, a_t) < r_c$ **then**
      break
    $s_{t+1}, r_t \leftarrow \text{Sim.step}(s_t, a_t)$
    $o_t \leftarrow \text{3DGS.Render}(s_t)$
    Append (s_t, a_t, o_t, r_t) to $\mathcal{D}^{\text{cf}}$

We generate counterfactual trajectories using a geometry-consistent 3D Gaussian Splatting (3DGS) simulator [56], which renders photorealistic observations conditioned on poses of ego vehicle and surrounding agents, modeling the observation distribution $\widehat{P}_{\text{obs}}(\cdot|s_t)$. As shown in Section 2.2, this rendering remains high-fidelity as long as the rollout policy induces a state distribution close to the reference distribution.

To ensure reliable observations while introducing behavioral diversity, we simulate rollouts from a pretrained E2E policy $\hat{\pi}_{\text{ref}}$ under a teacher-forcing setup, starting from the original reference states. To prevent rendering artifacts, we discard trajectories that deviate beyond a distance threshold or fall below a minimum reward. The rollout horizon T determines how far into the future these counterfactual behaviors extend; as we show in our experiments, longer horizons expose richer supervision signals for downstream learning, but increase the risk of divergence from reference data. The full generation procedure is summarized in Algorithm 1.

3.2. Diffusion-Based Policy Adaptation

We propose a diffusion-based policy adapter that refines the output of a frozen end-to-end (E2E) driving model by predicting residual trajectories $\Delta a = a^* - a^{\text{base}}$, where

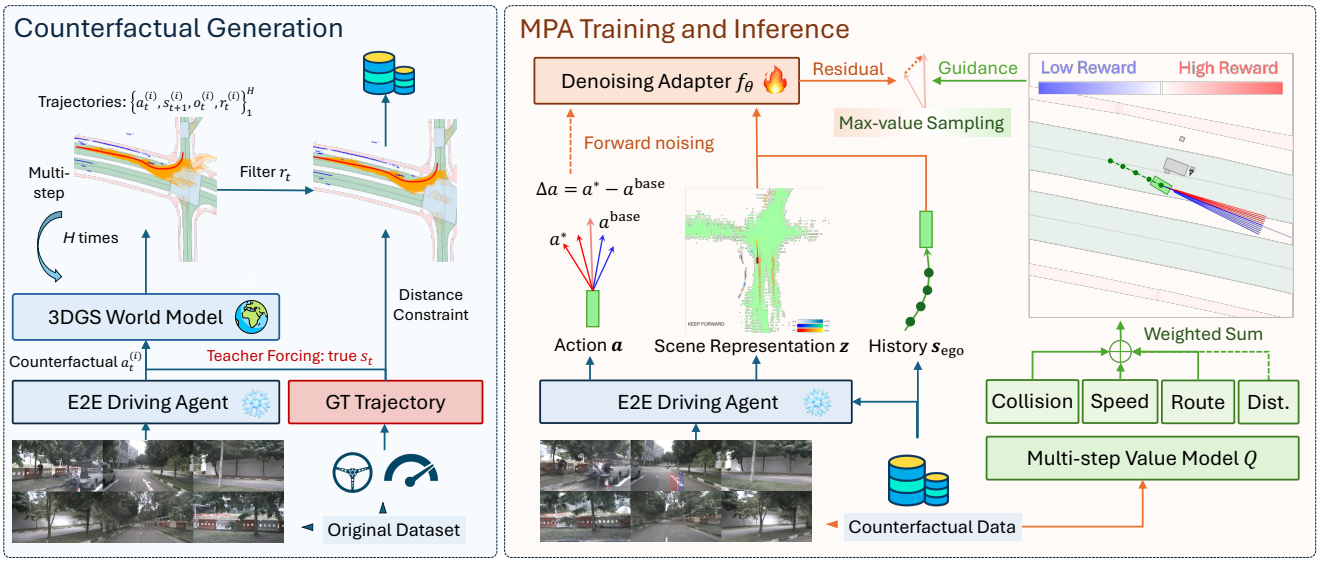

Figure 3. **Overview of Model-Based Policy Adaptation (MPA).** Left: We propose a counterfactual data generation pipeline, where we first generate initial data of T-step trajectories rolled out with pretrained E2E policy and 3DGS-based world model, and then filter the generated data with reward and distance constraints to improve data realism, resulting in counterfactual *(action, state, observation, reward)* sequences. Right: We utilize the generated data to train two MPA modules: (i) a diffusion policy adapter predicting residual actions on top of a base E2E agent, and (ii) a value model Q estimating multi-step cumulative rewards under different principles, such as collision and speed.

$a^{\text{base}} \in \mathbb{R}^{H\times 2}$ is the trajectory from a pretrained policy (e.g., UniAD) and a^* is the high-reward trajectory from counterfactual data.

Training. To model the distribution over residuals, we apply a latent diffusion process. The forward step adds Gaussian noise over K steps:

$$q(\Delta a^{(k)} \mid \Delta a^{(0)}) = \mathcal{N}\left(\Delta a^{(k)}; \sqrt{\bar{\alpha}_k}\Delta a^{(0)}, (1-\bar{\alpha}_k)\mathbf{I}\right),$$

where $\Delta a^{(0)} = \Delta a$ and $\bar{\alpha}_k$ is the cumulative noise schedule. The denoising network f_θ is a 1D U-Net that outputs multi-mode $\Delta a^{(0)}$ from $\Delta a^{(k)}$, conditioned on the scene encoding $z = \phi_{\text{enc}}(o, \boldsymbol{s}_{\text{ego}})$, ego history $\boldsymbol{s}_{\text{ego}}$, and base predicted trajectory a^{base}. The output of f_θ contains N modes, we denote the i-th mode as $f_\theta(...)[i]$. It is trained with the loss:

$$\mathbb{E}_{\Delta a^{(0)},k,\epsilon} \min_i \left\| f_\theta(\Delta a^{(k)}, k, z, \boldsymbol{s}_{\text{ego}}, a^{\text{base}})[i] - \Delta a^{(0)} \right\|_2^2,$$

where $\Delta a^{(k)} = \sqrt{\bar{\alpha}_k}\Delta a^{(0)} + \sqrt{1-\bar{\alpha}_k}\epsilon$, with $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Inference. For inference, we use DDIM [37] to sample N modes of $\Delta a^{(0)}$ and recover the adapted trajectory:

$$a^{\text{adapt}}[i] = a^{\text{base}} + \Delta a^{(0)}[i], \ \forall i \in [N].$$

This design adapts the knowledge from the pretrained base policy by conditioning on their encoded scene context and base actions, enabling generalization beyond the counterfactual data domain.

3.3. Principled Q-Value Guided Sampling

Training. While step-wise rewards can be computed given full access to s_t, estimating long-horizon returns is challenging under partial observability. Using the generated counterfactual dataset, we train a multi-step action value model $Q(o_t, s_t, a_t; T) = \sum_{t=1}^{T} \gamma^t r(s, a_t)$ based on four interpretable principles: route following Q_{route}, lane distance Q_{dist}, collision avoidance $Q_{\text{collision}}$, and speed compliance Q_{speed}. Each Q-function is trained independently to predict cumulative returns using $(o_t, \boldsymbol{s}_{\text{ego}}, a_t)$. $Q = \sum_{i\in\{\text{collision, dist, ...}\}} w_i \times Q_i$ We provide thorough ablations of the Q-function components in Sec. 4.3.

Inference. At inference, residual actions Δa are sampled following the policy adapter inference procedure from Sec. 3.2, and the best proposal is selected via

$$\hat{a}^* = \arg\max_{a \in a^{\text{adapt}}} Q(o_t, \boldsymbol{s}_{\text{ego}}, a; T)$$

Compared to classifier-based reward guidance [49, 54], our Q-value guidance offers feedback from longer planning horizons and avoids the gradient instability of the reward model.

4. Experiment Results

In our experiments, we aim to answer the following research questions. **RQ1**. Compared to the baselines, can MPA bring benefits to the E2E driving agents in a closed-loop evaluation in a generalizable way? **RQ2**. How does MPA benefit the

safety-critical performance in the closed-loop evaluation? **RQ3**. How do different adapters and value guidance modules contribute to the performance of MPA? **RQ4**. How does MPA scale with the number of counterfactual planning steps in the data generation phase?

4.1. Experiment Settings

Dataset and Simulation Engine. We utilize the nuScenes dataset [3] that consists of 5.5 hours of driving data in Boston and Singapore. Every scene has a reference trajectory of 20 seconds. We use HUGSIM [56] as the simulation engine and evaluation benchmark. We train on a split of 290 scenes in the nuScenes train-val split, and evaluate on three settings. (1) **In-domain evaluation**: the model will be tested on a sub-split of 70 scenes, the surrounding dynamic entities (vehicles, pedestrians) will be replayed by a fixed ratio of their reference trajectory in the offline dataset. (2) **Unseen nominal scene evaluation**: the model will be tested on a sub-split of 70 scenes that are unseen yet during training, the surrounding dynamic entities (vehicles, pedestrians) are nominal and will be replayed by a fixed ratio of their reference trajectory in the offline dataset. (3) **Safety-critical evaluation**: the model will be tested on 10 scenes, where there exists one (or few) non-native agents to challenge the ego agents in an adversarial way. The simulation frequency is 4 Hz. In all the scenes, the termination occurs under one of the following five conditions: (i) full route completion, (ii) off-road events, (iii) collision events, (iv) too far from the reference trajectory, (v) maximum rollout time limits (50 seconds, 2.5× of the reference trajectory) reached.

Baselines. We compare the MPA with diverse baselines in E2E driving algorithms that fall in the two following categories. (1) Pretrained base policy with open-loop training manner: we compare with the performance of **UniAD** [18], **VAD** [22] and **LTF** [6] on the HUGSIM dataset. We further build on our MPA with these policies. (2) E2E agents trained with curated counterfactual dataset: We further train several baseline policies with the curated dataset. **AD-MLP** [53] utilizes the ego's velocity, acceleration, past trajectories, and high-level command as the input, which is recognized as a naive baseline for the closed-loop Driving tasks. **BC-Safe** [35] uses the safe segments in the counterfactual datasets to train and End-to-End policies. **Diffusion** adopts the implementation of [29] in the scene encoding and utilizes a DDIM-based sampler [37] instead of truncated denoising during the inference time. To ensure a fair comparison between the MPA and the second category of the baselines, all the approach uses pretrained ResNet [15] as the perception backbone to encode the RGB inputs from 6 perspective cameras.

Metrics. We follow the evaluation protocol in HUGSIM [56], which is inspired by the NAVSIM-based metrics [7]. The metrics include Route Completion (RC), Non-Collision (NC), Driveable Area Compliance (DAC), Time-To-Collision (TTC), Comfort (COM), HUGSIM Driving Score (HDScore). Specifically, HDScore is computed with the above metrics along with Route Completion (RC), instead of the Ego Progress (EP) in PDMS [7]. HDScore is the weighted sum as follows:

$$\text{HDScore} = \text{RC} \times \frac{1}{T} \sum_{t=0}^{T} \Big\{ \prod_{m \in \{\text{NC, DAC}\}} \text{score}_m \times \frac{\sum_{m \in \{\text{TTC, COM}\}} \text{weight}_m \times \text{score}_m}{\sum_{m \in \{\text{TTC,COM}\}} \text{weight}_m} \Big\}_t .$$

We list all the metrics with ($\times 100$) in the tables. All the metrics fall in $[0.0, 100]$.

4.2. Main Results and Analysis (RQ1, RQ2)

To answer RQ1, we first evaluate the closed-loop performance for in- and out-of-domain scenes, as shown in Figure 4. All the reported MPA approaches are evaluated with 20 action samples at inference time. In-domain scenes refer to the scenes that are used to generate counterfactual training data in Singapore. We evaluate the quantitative closed-loop results in these training scenes in Table 1. MPA-based E2E driving agents achieved better results compared to their pretrained counterparts, as well as three baseline methods trained on the counterfactual curated dataset, especially in the most important metrics, RC and HDScore. The baseline AD-MLP moves very conservatively, so the NC and TTC are low, yet the RC is also quite low as it barely completes the assigned routes for the challenging E2E scenes. Besides, the NC score in HUGSIM is a bit underestimated compared to NAVSIM, as HUGSIM erodes the vehicle boxes compared to the actual size by the point clouds. This leads to a few false 'collision' signals during the evaluation. Yet, most of them will not cause a collision that terminates the entire episode in the closed-loop simulation. This is why we observe some high RC with mediocre NC metrics. This still means the ego agents are capable of navigating around different collisions and off-road maneuvers to reach the goal in a reasonable way, and will result in a good HDScore.

We further evaluate the closed-loop performance under the unseen scenes. We select 70 scenes in Boston that are not accessible in the curated counterfactual dataset. The qualitative results are shown in Table 2 (left). Compared to the in-domain results in Table 1, we can see a significantly degraded performance in AD-MLP and Diffusion, while the pretrained E2E policies still perform similarly as they do in the in-domain scenes. We observe that the MPA agents built upon the pretrained E2E policies are still optimal and quite robust under the unseen scenes. All three variants have

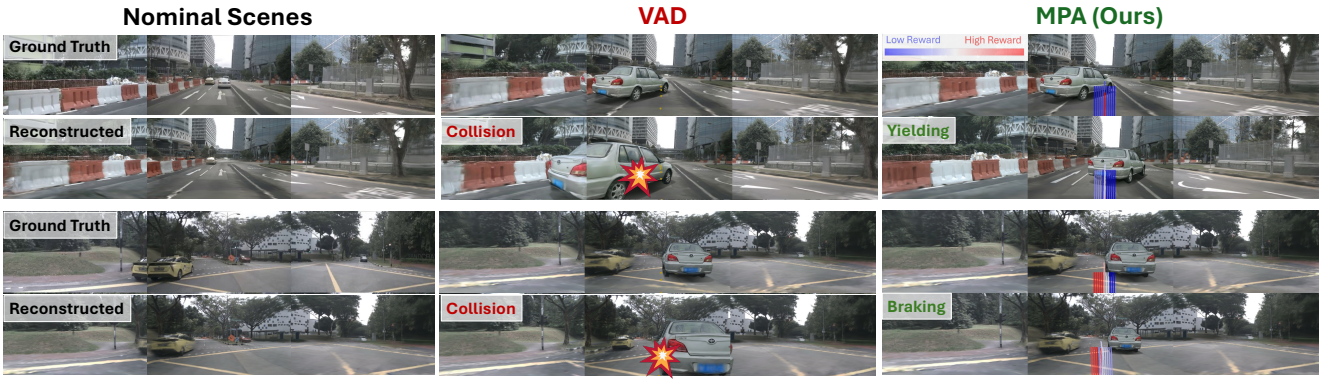

Figure 4. **Qualitative Results** in the in-domain and safety-critical scene. The silver car cuts in from the right side, forcing the ego vehicle to yield. Compared to the pretraind VAD, MPA-adapted policy can successfully brake and yield to the cut-in vehicles under the guidance of the Q-value model.

Table 1. **In-Domain Closed-Loop Evaluation Results.** All the evaluation metrics are higher the better. **Bold** means the best, and underlined is the best runner-up for each metrics.

Model	Ego Status	Camera	Curation	RC	NC	DAC	TTC	COM	HDScore
UniAD	✓	✓	✗	39.4	56.9	75.1	52.1	**98.7**	19.4
VAD	✓	✓	✗	50.1	68.4	87.2	66.1	90.2	31.9
LTF	✓	✓	✗	65.2	71.3	92.1	67.6	98.4	46.7
AD-MLP	✓	✗	✓	13.4	**80.2**	86.2	**79.4**	90.1	6.5
BC-Safe	✓	✓	✓	57.0	59.8	87.9	55.2	89.4	33.6
Diffusion	✓	✓	✓	71.8	67.4	88.1	64.5	91.5	45.1
MPA (UniAD)	✓	✓	✓	93.6	76.4	92.8	72.8	91.8	66.4
MPA (VAD)	✓	✓	✓	**94.9**	75.4	**93.6**	72.5	92.8	**67.0**
MPA (LTF)	✓	✓	✓	93.1	70.8	90.9	67.9	94.9	60.0

comparable HDScore with their in-domain evaluation. This demonstrates the generalizability of the proposed adapter and value model under unseen scene contexts.

4.3. Ablation Studies (RQ3, RQ4)

We conduct ablation studies to analyze the contribution of the three main modules of MPA: (i) counterfactual dataset generation, (ii) policy adapter, and (iii) Q-value guidance.

Counterfactual Dataset. We further analyze the impact of the curated counterfactual dataset by ablating the step size during the rollout of counterfactual data. Then we train the MPA over the curated dataset and evaluate its performance in the unseen scenes in Boston, similar to the setting in Table 2 (left). We illustrate the trends of evaluation metrics with respect to the step sizes in the counterfactual dataset in Figure 5. MPA benefits from longer counterfactual steps, as there would be more informative supervision for the value function training in the future steps.

Policy Adapter. In Table 3, we evaluate a few variants of MPA (UniAD). We evaluate the unseen scene's results across different variants. The comparison between ID-5 and ID-6 (ours) demonstrates the effectiveness of the adapter in a better route completion. Under the safety-critical scenes, the adapter brings ∼20% increase to the out route completion, leading to a significantly higher HDScore compared to the policies without the adapter.

Principles of Q-Value Guidance. In Table 3, we remove different principles in the state-action value function Q. Compared to the ID-5 variants, ID-1 removes the route information used in all the baselines, which leads to drastically degraded performance. ID-2 removes the distance function to the reference route, significantly degrading driveable area compliance and non-at-fault collision metrics. ID-3 removes the collision values, and ID-4 removes the speeding value function. Both of them have an impact on HDScore, especially in safety-critical situations. The reason that NC still seemed to be high for ID-3 in safety-critical scenes is that the available frame length before collision is short, which makes

Table 2. **Out-of-Domain Closed-Loop Evaluation Results** in unseen nominal and safety-critical scenes. All the evaluation metrics are higher the better. **Bold** means the best, and underlined is the best runner-up for each metrics.

	Unseen Nominal Scenes						Safety-Critical Scenes					
Model	**RC**	**NC**	**DAC**	**TTC**	**COM**	**HDScore**	**RC**	**NC**	**DAC**	**TTC**	**COM**	**HDScore**
UniAD	39.3	56.6	74.0	52.6	**98.2**	22.2	11.4	76.2	82.1	57.8	95.9	4.5
VAD	45.4	64.8	86.2	62.0	95.9	29.3	25.4	77.0	88.3	73.2	88.4	16.0
LTF	63.3	64.8	86.5	62.8	**98.2**	41.9	35.1	80.9	96.8	78.1	**100.0**	24.2
AD-MLP	7.6	**71.6**	82.2	**69.8**	92.3	3.3	4.9	**93.5**	96.2	**93.4**	85.9	4.3
BC-Safe	59.2	59.8	81.2	56.3	95.9	34.6	20.2	80.1	91.7	67.3	86.7	13.5
Diffusion	57.9	62.1	83.5	58.3	96.2	35.1	20.9	84.3	92.3	72.4	86.3	13.1
MPA (UniAD)	**93.7**	69.5	92.9	66.6	97.6	60.9	95.1	76.8	98.9	74.2	97.7	70.4
MPA (VAD)	90.9	71.0	**94.4**	68.8	97.7	**61.2**	**96.6**	79.8	**99.0**	77.3	97.7	**74.7**
MPA (LTF)	91.8	68.3	91.0	66.5	96.9	57.0	87.3	72.0	94.0	66.9	97.8	56.3

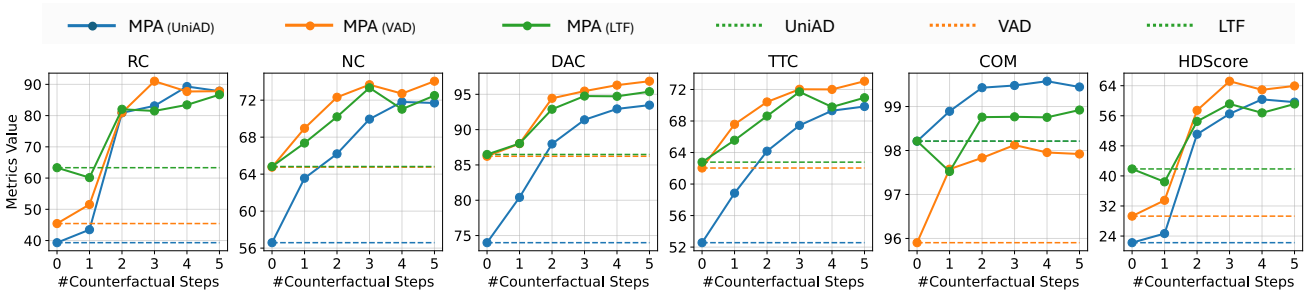

Figure 5. **Impact of Rollout Steps** (T from Algorithm 1) during the counterfactual data generation. We fix the sample size to six during the closed-loop evaluation for all the MPA variants. MPA trained with more counterfactual steps results in better test-time performance, as the Q value model takes more future steps as the supervision signals.

the denominator in NC smaller compared to the other group. However, when we look at the RC metrics, the drop when removing $Q_{\text{collision}}$ is significant, as the agents will encounter a collision and end to episode earlier than the nominal cases.

5. Related Works

End-to-End Autonomous Driving. End-to-End (E2E) autonomous driving has achieved significant progress by jointly training the detection, tracking, prediction, and planning modules to avoid information loss throughout the cascading system. ST-P3 [17] and UniAD [18] propose unified E2E frameworks that achieve state-of-the-art open-loop performance on the nuScenes dataset [3]. VAD [22] encodes the driving scene with a vectorized representation and incorporates query-based planning modules, and VAD-v2 [22] further designs a probabilistic planning approach and improves the closed-loop performance over the CARLA [9] benchmark. Hydra-MDP follows VAD-v2's query-based framework and conducts multi-target hydra distillation with a set of scoring rules. With the prosperity of foundation models, a series of works [2, 32, 34, 40, 41] incorporate Large Language Models (LLMs) and Vision-Language Models (VLMs) into the E2E planning pipeline. Despite the benefits of commonsense reasoning with foundation models, most of the existing models still focus on the open-loop evaluation or the approximate closed-loop metrics from the open-loop evaluation [7], which lack counterfactual reasoning for the safety-critical scenarios. A recent work RAD [11] pays attention to the photorealistic closed-loop evaluation and utilizes imitation learning and online reinforcement learning to fine-tune the E2E driving agents. Yet, the value functions trained for the Proximal Policy Optimization (PPO) agents are not effectively integrated during the inference. MPA aims to use the value model as an effective inference-time guidance to make the E2E agents more robust in closed-loop evaluation.

Counterfactual Data Generation. Counterfactual data generation has been explored within the context of offline reinforcement learning. Wang et al. [43] utilize a learned model to autonomously generate additional offline data, thereby enhancing the training of sequence models. OASIS [52] introduces a method to produce counterfactual data by modulating guidance signals during diffusion model inference. In high-stakes decision-making domains such as autonomous driving, generating counterfactual data is essential due to the limited presence of safety-critical scenarios in existing datasets. Previous research has addressed the trade-off between *realism* and *controllability*

Table 3. **Ablation Study** on MPA's variants on the UniAD base policy. **Top**: Unseen scenes that are nominal but not appearing in the training dataset. **Bottom**: Safety-critical scenes with adversarial surrounding agents. **Bold** means the best, and underlined is the best runner-up for each metrics.

ID	Q_{route}	Q_{dist}	$Q_{\text{collision}}$	Q_{speed}	Adapter	RC	NC	DAC	TTC	COM	HDScore
1		✓	✓	✓		6.9	**81.2**	95.1	81.0	**100**	5.1
2	✓		✓	✓		83.9	57.0	81.0	53.6	99.4	43.2
3	✓	✓		✓		89.2	70.8	**95.6**	68.6	99.4	60.8
4	✓	✓	✓			90.4	68.9	91.8	65.4	99.4	56.6
5	✓	✓	✓	✓		91.1	71.5	94.1	69.4	99.4	**60.9**
6	✓	✓	✓	✓	✓	**93.7**	69.5	92.9	66.6	97.6	**60.9**
ID	Q_{route}	Q_{dist}	$Q_{\text{collision}}$	Q_{speed}	**Adapter**	**RC**	**NC**	**DAC**	**TTC**	**COM**	**HDScore**
1		✓	✓	✓		4.6	**86.0**	98.3	**79.3**	90.1	3.6
2	✓		✓	✓		65.1	65.6	85.7	53.8	86.5	39.5
3	✓	✓		✓		57.7	82.4	**99.0**	69.6	84.6	39.2
4	✓	✓	✓			79.3	82.9	98.5	68.0	93.9	50.1
5	✓	✓	✓	✓		75.6	81.2	98.8	78.6	**99.7**	55.3
6	✓	✓	✓	✓	✓	**95.1**	76.8	98.9	74.2	97.7	**70.4**

in safety-critical scenario generation by integrating various constraints. These include inference-time sampling techniques [39], retrieval-augmented generation [8], low-rank fine-tuning approaches [10], and language-conditioned generation methods [38]. However, these efforts primarily focus on behavioral scenario generation without incorporating visual information. With advancements in Neural Radiance Fields (NeRF) and 3D Gaussian Splatting (3DGS), recent studies such as DriveArena [50] and MagicDrive [12] have begun developing E2E simulators for closed-loop evaluation. Similarly, RAD [11] employs a 3DGS-based simulator for RL fine-tuning. Notably, to date, no existing work has focused on generating E2E counterfactual data within E2E simulators.

Reward Model for Inference-Time Scaling. Recent LLM research has shown the power of the reward model in LLM's inference-time scaling [30, 36]. In sequential decision-making problems, the reward model was explicitly used for inference-time supervision signals, such as the guidance for the diffusion-based policy models [1, 16, 21]. For the closed-loop autonomous driving and decision-making tasks, several prior works incorporate reward models as the classifier-based guidance to steer the diffusion model's sampling process [23, 24, 54]. Other format of the guidance signals include signal temporal logic (STL) guidance, language-based guidance [55], adversarial guidance [4, 45, 47], and game-theoretic guidance [20]. In the autonomous driving domain, prior works like DiffusionDrive [29] utilize truncated denoising for the diffusion models without additional classifier guidance. DiffAD utilizes action conditional guidance for E2E driving [44]. Diffusion-ES [49], Gen-Drive [19], and Diffusion-Planner [54] utilize customized reward models as test-time guidance for the non-E2E driving tasks. To the best of our knowledge, MPA is the first work incorporating the driving reward model for the inference-time scaling of E2E driving agents.

6. Conclusion

In this work, we introduce MPA, a general framework for improving the closed-loop trustworthiness of E2E autonomous driving agents. MPA begins by generating high-quality counterfactual trajectories through geometry-consistent rollouts in a 3DGS-based simulation environment. This results in a better data coverage while preserving visual fidelity. With the counterfactual dataset, MPA further trains a diffusion-based policy adapter to refine base policy predictions and leverages a multi-principle value model to guide inference-time decision making. These components allow pretrained agents to generate and evaluate multiple trajectory proposals, selecting actions that optimize long-term driving outcomes. Experimental results on nuScenes data and HUGSIM benchmark demonstrate MPA's effectiveness in boosting safety and generalizability.

Limitations. Despite these promising results, our approach assumes reliable rendering from 3DGS under constrained trajectory deviations and currently decouples value modeling from policy optimization. Future work includes extending our current results to diverse driving datasets, exploring the online RL training over the 3DGS simulator, and deploying MPA to the multi-modal foundation models to enhance reasoning capability for more severe distribution shifts in autonomous driving.

References

[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B Tenenbaum, Tommi S Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *The Eleventh International Conference on Learning Representations*. 8

[2] Vahid Behzadan and Arslan Munir. Vulnerability of deep reinforcement learning to policy induction attacks. In *International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 262–275. Springer, 2017.

[3] H Caesar, V Bankiti, AH Lang, S Vora, VE Liong, Q Xu, A Krishnan, Y Pan, G Baldan, and O Beijbom. nuscenes: A multimodal dataset for autonomous driving. arxiv. 2019. 5, 7

[4] Wei-Jer Chang, Francesco Pittaluga, Masayoshi Tomizuka, Wei Zhan, and Manmohan Chandraker. Controllable safety-critical closed-loop traffic simulation via guided diffusion. *arXiv preprint arXiv:2401.00391*, 2023. 8

[5] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024. 1

[6] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022. 1, 5

[7] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems*, 37:28706–28719, 2024. 2, 5, 7

[8] Wenhao Ding, Yulong Cao, Ding Zhao, Chaowei Xiao, and Marco Pavone. Realgen: Retrieval augmented generation for controllable traffic scenarios. *arXiv preprint arXiv:2312.13303*, 2023. 8

[9] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017. 1, 7

[10] Robert Dyro, Matthew Foutter, Ruolin Li, Luigi Di Lillo, Edward Schmerling, Xilin Zhou, and Marco Pavone. Realistic extreme behavior generation for improved av testing. *arXiv preprint arXiv:2409.10669*, 2024. 8

[11] Hao Gao, Shaoyu Chen, Bo Jiang, Bencheng Liao, Yiang Shi, Xiaoyang Guo, Yuechuan Pu, Haoran Yin, Xiangyu Li, Xinbang Zhang, et al. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. *arXiv preprint arXiv:2502.13144*, 2025. 1, 7, 8

[12] Ruiyuan Gao, Kai Chen, Enze Xie, Lanqing Hong, Zhenguo Li, Dit-Yan Yeung, and Qiang Xu. Magicdrive: Street view generation with diverse 3d geometry control. *arXiv preprint arXiv:2310.02601*, 2023. 8

[13] Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. *arXiv preprint arXiv:2405.17398*, 2024. 1

[14] Junhao Ge, Zuhong Liu, Longteng Fan, Yifan Jiang, Jiaqi Su, Yiming Li, Zhejun Zhang, and Siheng Chen. Unraveling the effects of synthetic data on end-to-end autonomous driving. *arXiv preprint arXiv:2503.18108*, 2025. 1

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 5

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 8

[17] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022. 1, 7

[18] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023. 1, 5, 7

[19] Zhiyu Huang, Xinshuo Weng, Maximilian Igl, Yuxiao Chen, Yulong Cao, Boris Ivanovic, Marco Pavone, and Chen Lv. Gen-drive: Enhancing diffusion generative driving policies with reward modeling and reinforcement learning fine-tuning. *arXiv preprint arXiv:2410.05582*, 2024. 8

[20] Zhiyu Huang, Zixu Zhang, Ameya Vaidya, Yuxiao Chen, Chen Lv, and Jaime Fernández Fisac. Versatile scene-consistent traffic scenario generation as optimization with diffusion. *arXiv preprint arXiv:2404.02524*, 2024. 8

[21] Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022. 8

[22] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized

scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023. 1, 5, 7

[23] Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9644–9653, 2023. 8

[24] Chiyu Max Jiang, Yijing Bai, Andre Cornman, Christopher Davis, Xiukun Huang, Hong Jeon, Sakshum Kulshrestha, John Wheatley Lambert, Shuangyu Li, Xuanyu Zhou, et al. Scenediffuser: Efficient and controllable driving simulation initialization and rollout. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 8

[25] Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, et al. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 629–636. IEEE, 2024. 2

[26] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1

[27] Norman Lehtomaki, NJAM Sandell, and Michael Athans. Robustness results in linear-quadratic gaussian based multivariable control designs. *IEEE Transactions on Automatic Control*, 26(1):75–93, 1981. 2

[28] Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, et al. Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*, 2024. 1

[29] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024. 5, 8

[30] Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*, 2025. 8

[31] William Ljungbergh, Adam Tonderski, Joakim Johnander, Holger Caesar, Kalle Åström, Michael Felsberg, and Christoffer Petersson. Neuroncap: Photorealistic closed-loop safety testing for autonomous driving. In *European Conference on Computer Vision*, pages 161–177. Springer, 2024. 1, 3

[32] Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*. 7

[33] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1

[34] Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G Allievi, Senem Velipasalar, and Liu Ren. Vlp: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14760–14769, 2024. 7

[35] Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Agile autonomous driving using end-to-end deep imitation learning, 2019. 5

[36] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. 8

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 4, 5

[38] Shuhan Tan, Boris Ivanovic, Xinshuo Weng, Marco Pavone, and Philipp Kraehenbuehl. Language conditioned traffic generation. In *Conference on Robot Learning*, pages 2714–2752. PMLR, 2023. 8

[39] Shuhan Tan, Kelvin Wong, Shenlong Wang, Sivabalan Manivasagam, Mengye Ren, and Raquel Urtasun. Scenegen: Learning to generate realistic traffic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 892–901, 2021. 8

[40] Ran Tian, Boyi Li, Xinshuo Weng, Yuxiao Chen, Edward Schmerling, Yue Wang, Boris Ivanovic, and Marco Pavone. Tokenize the world into object-level knowledge to address long-tail events in autonomous driving. *arXiv preprint arXiv:2407.00959*, 2024. 7

[41] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, XianPeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. In *8th Annual Conference on Robot Learning*. 7

[42] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. Neurad: Neural rendering for autonomous driving. *arXiv preprint arXiv:2311.15260*, 2023. 1

[43] Kerong Wang, Hanye Zhao, Xufang Luo, Kan Ren, Weinan Zhang, and Dongsheng Li. Bootstrapped transformer for offline reinforcement learning. *Advances*

in Neural Information Processing Systems, 35:34748–34761, 2022. 7

[44] Tao Wang, Cong Zhang, Xingguang Qu, Kun Li, Weiwei Liu, and Chang Huang. Diffad: A unified diffusion modeling approach for autonomous driving. *arXiv preprint arXiv:2503.12170*, 2025. 8

[45] Yuting Xie, Xianda Guo, Cong Wang, Kunhua Liu, and Long Chen. Advdiffuser: Generating adversarial safety-critical driving scenarios via guided diffusion. *arXiv preprint arXiv:2410.08453*, 2024. 8

[46] Ziyang Xie, Zhizheng Liu, Zhenghao Peng, Wayne Wu, and Bolei Zhou. Vid2sim: Realistic and interactive simulation from video for urban navigation. *arXiv preprint arXiv:2501.06693*, 2025. 1

[47] Chejian Xu, Ding Zhao, Alberto Sangiovanni-Vincentelli, and Bo Li. Diffscene: Diffusion-based safety-critical scenario generation for autonomous vehicles. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023. 8

[48] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *ECCV*, 2024. 1

[49] Brian Yang, Huangyuan Su, Nikolaos Gkanatsios, Tsung-Wei Ke, Ayush Jain, Jeff Schneider, and Katerina Fragkiadaki. Diffusion-es: Gradient-free planning with diffusion for autonomous driving and zero-shot instruction following. *arXiv preprint arXiv:2402.06559*, 2024. 4, 8

[50] Xuemeng Yang, Licheng Wen, Yukai Ma, Jianbiao Mei, Xin Li, Tiantian Wei, Wenjie Lei, Daocheng Fu, Pinlong Cai, Min Dou, et al. Drivearena: A closed-loop generative simulation platform for autonomous driving. *arXiv preprint arXiv:2408.00415*, 2024. 1, 2, 3, 8

[51] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1389–1399, June 2023. 1

[52] Yihang Yao, Zhepeng Cen, Wenhao Ding, Haohong Lin, Shiqi Liu, Tingnan Zhang, Wenhao Yu, and Ding Zhao. Oasis: Conditional distribution shaping for offline safe reinforcement learning. *Advances in Neural Information Processing Systems*, 37:78451–78478, 2024. 7

[53] Jiang-Tian Zhai, Ze Feng, Jinhao Du, Yongqiang Mao, Jiang-Jiang Liu, Zichang Tan, Yifu Zhang, Xiaoqing Ye, and Jingdong Wang. Rethinking the open-loop evaluation of end-to-end autonomous driving in nuscenes. *arXiv preprint arXiv:2305.10430*, 2023. 5

[54] Yinan Zheng, Ruiming Liang, Kexin Zheng, Jinliang Zheng, Liyuan Mao, Jianxiong Li, Weihao Gu, Rui Ai, Shengbo Eben Li, Xianyuan Zhan, et al. Diffusion-based planning for autonomous driving with flexible guidance. *arXiv preprint arXiv:2501.15564*, 2025. 4, 8

[55] Ziyuan Zhong, Davis Rempe, Yuxiao Chen, Boris Ivanovic, Yulong Cao, Danfei Xu, Marco Pavone, and Baishakhi Ray. Language-guided traffic simulation via scene-level diffusion. In *Conference on Robot Learning*, pages 144–177. PMLR, 2023. 8

[56] Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. *arXiv preprint arXiv:2412.01718*, 2024. 1, 2, 3, 5