# GEOMETRICALLY ALIGNED DIFFUSION FOR END-TO-END AUTONOMOUS DRIVING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

While generative diffusion models have emerged as a powerful paradigm for multimodal trajectory generation, their deployment in safety-critical autonomous driving is fundamentally bottlenecked by a critical misalignment: unconstrained statistical generation frequently violates physical, geometric, and kinematic constraints. To address this, we propose STEER, a unified framework that systematically embeds physical inductive biases into the generative lifecycle through a two-fold alignment strategy. At the macro-geometric level, STEER grounds the generative manifold in physical space by introducing Scene-Conditioned Anchor generation (SCA) for topology-compliant prior initialization, while leveraging Graph Attention Agent Interaction (GAI) to enforce spatial locality and prevent attention entropy collapse during iterative denoising. At the micro-kinematic level, STEER guarantees high-order trajectory feasibility and deterministic execution by coupling Adaptive Dual-Path Refinement (ADPT) with a Hierarchical Trajectory Selection (HTS) mechanism. Specifically, ADPT fuses global intent with local autoregressive modeling for kinematic smoothness, while HTS decodes the generative density through joint optimization over predictive utility, topological boundaries, and motion feasibility. Extensive experiments on the NAVSIM benchmark demonstrate that STEER establishes a new state-of-the-art, successfully bridging statistical generative modeling with physically grounded embodied planning.

## 1 INTRODUCTION

Autonomous driving in complex, interactive urban environments fundamentally requires modeling highly multimodal and uncertain future states Bansal et al. (2018); Bharilya & Kumar (2024); Xu et al. (2025). Recently, generative foundation models Jiang et al. (2025); Li et al. (2025c), particularly diffusion models Ho et al. (2020); Rombach et al. (2022), have catalyzed a paradigm shift in trajectory planning and motion generation. By formulating driving as a conditional generation task, pioneering works such as Scenediffuser Jiang et al. (2024), DiffusionDrive Liao et al. (2025), and DiffusionDriveV2 Zou et al. (2025) have demonstrated unprecedented capabilities in capturing the joint distribution of multi-agent behaviors and generating diverse, human-like trajectories.

Despite these promising strides, deploying diffusion-based generative models in safety-critical physical systems remains severely bottlenecked by a critical misalignment: the unconstrained, purely statistical nature of the generative manifold clashes with the strict spatial, kinematic, and topological constraints of the real world. Current diffusion planners suffer from a systemic lack of physical inductive biases across their entire pipeline, spanning two fundamental dimensions:

At the macro-geometric level, existing models rely on rigid initialization using static, scene-agnostic anchors derived from offline K-Means clustering Chai et al. (2019); Phan-Minh et al. (2020); Liao et al. (2025). These fixed priors fail to dynamically adapt to local road topologies, limiting generalization and slowing convergence. Furthermore, standard decoders employ permutation-invariant attention mechanisms that treat agents as an isotropic set Ngiam et al. (2021); Nayakanti et al. (2022). This lack of geometric grounding causes attention dilution in dense traffic, where spurious correlations from distant agents destabilize denoising.

At the micro-kinematic level, standard models rely on unconstrained, monolithic iterative denoising Janner et al. (2022); Jiang et al. (2023b), struggling to capture fine-grained temporal kinematics



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

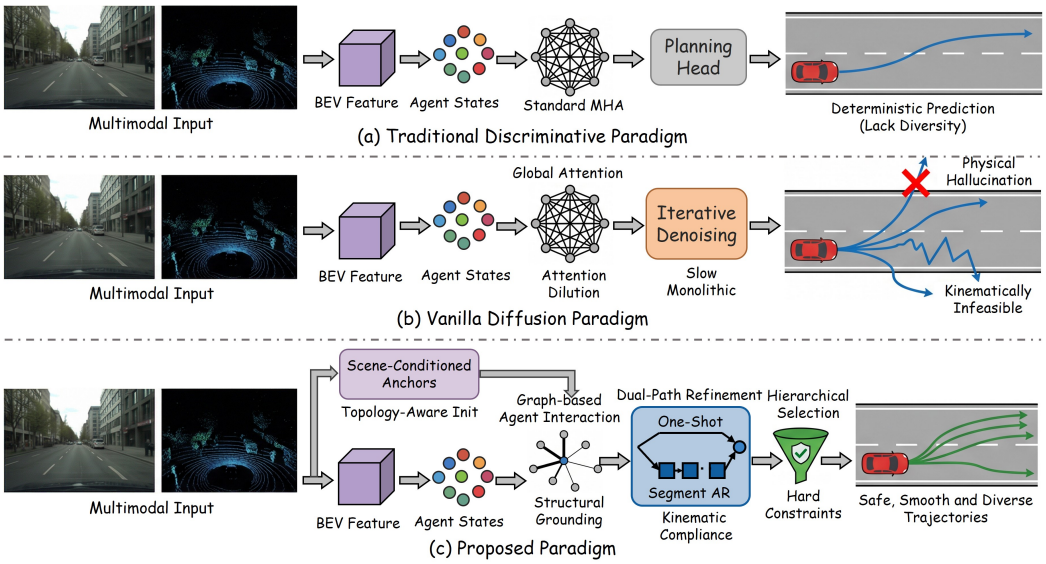



Figure 1: Conceptual comparison of trajectory planning paradigms. (a) Traditional discriminative methods lack multimodal diversity. (b) Vanilla diffusion models generate diverse but often kinematically infeasible trajectories with physical hallucinations due to a severe lack of structural inductive biases. (c) Our proposed STEER systematically injects physical priors into every stage of the diffusion pipeline, successfully aligning the unconstrained generative manifold with the strict safety and kinematic constraints of real-world driving.

and forcing a suboptimal trade-off between global structural intent and local smoothness. Finally, pure data-driven resolution fails to respect hard physical limits, generating out-of-distribution hallucinations Zhang et al. (2025); Datta & Sundararaman (2025) that cannot be safely collapsed into executable control actions via simple heuristics.

To bridge this fundamental gap, we present STEER, a next-generation diffusion framework that reconceptualizes trajectory synthesis as a physically grounded, coarse-to-fine generation process (Fig. 1). STEER systematically injects structural inductive biases into every stage of the diffusion pipeline through a two-fold alignment strategy. On the **macro-geometric level**, Scene-Conditioned Anchor Generation (SCA) replaces static priors with map-adaptive initializations for rapid convergence, while Graph Attention-based Agent Interaction (GAI) anchors agents to local environmental manifolds via spatial decay priors, suppressing attention entropy collapse in complex traffic. On the **micro-kinematic level**, Adaptive Dual-Path Trajectory Refinement (ADPT) dynamically fuses one-shot global predictions with segment-wise progressive autoregressive modeling to ensure high-order derivative continuity, while Hierarchical Trajectory Selection (HTS) decodes the multimodal density through joint optimization over closed-loop utility, topological boundaries, and kinematic penalties. In summary, our main contributions are as follows:

- **A Novel Paradigm for Diffusion Planning:** We propose STEER, a unified end-to-end framework that resolves the critical misalignment between statistical generative manifolds and physical reality by systematically embedding structural inductive biases throughout the generative lifecycle.
- **Topology- and Geometry-Grounded Denoising:** We introduce context-adaptive anchor generation and spatial-decay agent interactions, overcoming the long-standing challenges of rigid static priors and attention dilution in dense traffic.
- **Kinematic-Compliant Refinement & Safe Decoding:** We develop a dual-path refinement mechanism coupled with hierarchical decision-theoretic selection, guaranteeing high-order kinematic smoothness and physically bounded execution without hallucinations.
- **State-of-the-Art Performance:** Extensive evaluations on the challenging NAVSIM benchmark demonstrate that STEER achieves new state-of-the-art results across safety, physical feasibility, and generation efficiency.

# 2 Related Work

## 2.1 End-to-End Autonomous Driving

End-to-end autonomous driving seeks to jointly learn scene understanding and driving decisions from sensory observations. Recent approaches have progressively moved from multi-task planning-oriented frameworks such as UniAD Hu et al. (2023) to compact vectorized representations in VAD Jiang et al. (2023a) and probabilistic multi-modal planning in VADv2 Chen et al. (2024). SparseDrive Sun et al. (2025) further explores sparse scene representations and hierarchical trajectory selection, while subsequent methods investigate more efficient multi-modal planning and trajectory representations Jiang et al. (2023a); Chen et al. (2024); Sun et al. (2025); Liao et al. (2025). More recently, diffusion models have been introduced as a generative paradigm for end-to-end planning. MotionDiffuser Jiang et al. (2023b) demonstrates the capability of diffusion models to capture diverse motion distributions, followed by DiffusionDrive Liao et al. (2025) and Diffusion Planner Zheng et al. (2025), which explore efficient diffusion-based trajectory generation for autonomous driving. These methods establish diffusion as a promising alternative to deterministic or vocabulary-based trajectory prediction, but existing designs primarily focus on generative modeling and sampling efficiency. In contrast, STEER investigates how the diffusion trajectory space can be better grounded in the scene through geometry-aware initialization and interaction, while maintaining efficient multi-modal generation.

## 2.2 Multi-modal Trajectory Generation and Selection

Multi-modal trajectory generation is essential for autonomous driving since identical observations may admit multiple feasible behaviors. Existing approaches represent this uncertainty using trajectory anchors or discrete maneuver vocabularies Lee et al. (2017); Chai et al. (2019); Liu et al. (2022); Chen et al. (2024), latent-variable models Yuan et al. (2021), or generative models such as diffusion Jiang et al. (2023b); Liao et al. (2025). Beyond diversity, generated trajectories should respect road geometry, surrounding-agent interactions, and vehicle dynamics. Accordingly, recent methods incorporate map and interaction constraints during prediction Jiang et al. (2023b), or employ explicit trajectory scoring and reranking to select feasible actions Dauner et al. (2023); Sun et al. (2025). PDM Dauner et al. (2023), for example, evaluates candidate trajectories using physics- and rule-based criteria, while SparseDrive adopts hierarchical selection with collision-aware rescoring Sun et al. (2025; 2026). Nevertheless, trajectory generation and physical selection are often handled as largely separate stages. STEER instead connects these two aspects through scene-conditioned trajectory anchors, geometry-aligned interaction, adaptive trajectory refinement, and hierarchical physical selection, encouraging the generated trajectory distribution to remain consistent with both scene geometry and downstream driving constraints.

# 3 Methodology

The overall architecture of our proposed STEER framework is illustrated in Fig. 2. We formulate diffusion-based trajectory planning as a structurally grounded, coarse-to-fine generation process. The pipeline consists of four key components: (1) a SCA module (Sec. 3.1) for topology-aware initialization; (2) a GAI module (Sec. 3.2) to enforce distance-aware spatial grounding during denoising; (3) an ADPT module (Sec. 3.3) to ensure strict kinematic compliance; and (4) a HTS mechanism (Sec. 3.4) to safely collapse the generated multimodal distribution into an executable action.

## 3.1 Scene-Conditioned Anchor Generation

Pure diffusion models typically initiate the reverse process from an isotropic Gaussian distribution or fixed, scene-agnostic priors derived from offline clustering. In autonomous driving, this unconstrained initialization severely violates the highly structured topological manifold of real-world roads, leading to slow convergence and out-of-distribution failures. To address this fundamental flaw, we propose the SCA mechanism to dynamically inject topology-aware priors based on the current context.

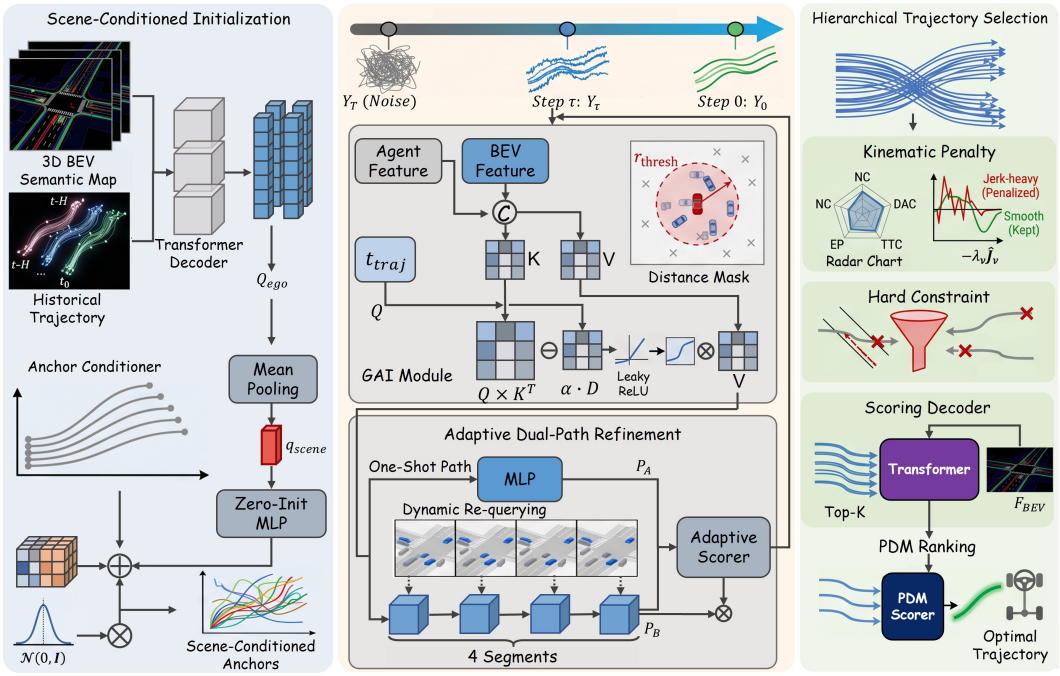



Figure 2: Detailed architecture of the proposed STEER framework. SCA: Ego queries are aggregated to predict dynamic offsets via a zero-initialized MLP, adapting base anchors to the current road topology for a high-quality diffusion starting point. Diffusion Denoising: At each step, the GAI module enforces a sparse, distance-aware interaction graph by injecting a spatial decay prior. Subsequently, the ADPT module dynamically fuses one-shot global predictions with segment-wise autoregressive modeling to ensure strict kinematic compliance. HTS: The generated multimodal distribution is safely collapsed into a single optimal trajectory through progressive filtering, including kinematic jerk penalization, hard constraint masking, and closed-loop PDM reranking.

Given ego queries $Q_{ego} \in \mathbb{R}^{N \times d}$ from the perception backbone, where $N$ denotes the number of queries and $d$ is the hidden feature dimension, we first extract a global scene representation $q_{scene} \in \mathbb{R}^d$ via mean pooling. A zero-initialized MLP then maps $q_{scene}$ to a dynamic offset $\Delta\mu \in \mathbb{R}^{L \times T \times 2}$. Here, $L$ represents the number of multimodal anchor modes, $T$ is the future prediction horizon (i.e., the number of trajectory waypoints), and 2 corresponds to the 2D spatial coordinates $(x, y)$. The scene-conditioned anchor mean is formulated as $\mu_{scene} = \mu_{base} + \Delta\mu$, where $\mu_{base}$ is the base learnable anchor. During training, we employ the reparameterization trick to sample the final anchors $A = \mu_{scene} + \sigma_{base} \odot \epsilon$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, encouraging multimodal exploration. During inference, we directly utilize $\mu_{scene}$ to ensure deterministic predictions. This context-aware initialization significantly accelerates the subsequent diffusion convergence. Detailed architectural configurations and mathematical formulations of SCA are provided in Appendix A.1.

### 3.2 Graph Attention-based Agent Interaction

Initialized with the scene-conditioned anchors $A$, the diffusion decoder begins the iterative denoising process. To ensure safe navigation, the ego vehicle must accurately attend to surrounding agents. However, standard Multi-Head Attention (MHA) is inherently permutation-invariant and spatially blind. In dense physical traffic, relying solely on semantic similarity leads to attention entropy collapse; the model establishes spurious correlations with distant, irrelevant agents, destabilizing the denoising process.

To explicitly inject spatial locality as a geometric inductive bias, we propose the GAI module. We first enrich semantic agent features with local BEV context via bilinear grid sampling. To enforce spatial locality, we compute the Euclidean distance matrix $\mathbf{D}$ between trajectory centroids and agents, masking out agents beyond a threshold $r_{\text{thresh}}$. Crucially, we explicitly encode an inverse

distance law into the attention mechanism by subtracting the scaled distance from the semantic logits. The final attention weight $\mathbf{A}_{ij}$ is computed using a LeakyReLU activation to maintain gradient flow for sparse patterns:

$$\mathbf{A}_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\frac{\mathbf{q}_i \mathbf{k}_j^T}{\sqrt{d_h}} - \alpha \|\mathbf{p}_i - \mathbf{p}_j\|_2\right)\right)}{\sum_{k \in \mathcal{N}(i)} \exp\left(\text{LeakyReLU}\left(\frac{\mathbf{q}_i \mathbf{k}_k^T}{\sqrt{d_h}} - \alpha \|\mathbf{p}_i - \mathbf{p}_k\|_2\right)\right)}. \tag{1}$$

This distance-biased graph attention ensures that physically closer agents exert a stronger influence on the trajectory. See Appendix A.2 for the comprehensive architectural diagram and tensor operations of GAI.

### 3.3 Adaptive Dual-Path Trajectory Refinement

While the GAI module provides robust spatial grounding, standard diffusion denoising is monolithic. This creates an inherent optimization conflict: the network struggles to simultaneously satisfy low-frequency global topological positioning and high-frequency local kinematic smoothness. To decouple this optimization, we introduce the ADPT module, which concurrently executes one-shot and autoregressive (AR) predictions to reconcile global intent with strict physical feasibility.

The one-shot branch provides fast, globally consistent predictions via an MLP: $\mathbf{P}_A = \mathbf{P}_{noisy} \oplus \mathcal{M}_{\text{one-shot}}(\mathbf{F}_{traj})$. Simultaneously, the progressive AR branch captures fine-grained kinematics by dividing the $T$ waypoints into $K$ segments. For each segment, it dynamically re-queries local BEV features and predicts local residuals conditioned on the previous segment's output, yielding $\mathbf{P}_B$. To dynamically balance global positioning and local smoothness, an Adaptive Trajectory Scorer evaluates the geometric morphological features of both candidates, outputting normalized weights $[w_A, w_B]$ via a Softmax head. The final refined trajectory is obtained by the weighted fusion:

$$\mathbf{P}_{final} = w_A \mathbf{P}_A + w_B \mathbf{P}_B. \tag{2}$$

We defer the rigorous mathematical formulation and efficiency analysis of the segment-wise AR mechanism to Appendix A.3.

### 3.4 Hierarchical Trajectory Selection

After the iterative denoising and refinement, the diffusion model outputs a diverse set of candidate trajectories. To safely collapse this generative distribution into a single executable action, we propose a Hierarchical Trajectory Selection (HTS) mechanism.

We first evaluate all candidates using a coarse scorer that predicts sub-scores for Non-Collision, Drivable Area Compliance, Time-to-Collision, Ego Progress, and Comfort. To explicitly ensure physical feasibility, we penalize the base reward $R_{\text{base}}$ using normalized kinematic speed and heading jerks ($\hat{J}_v, \hat{J}_\theta$):

$$R_{\text{coarse}} = R_{\text{base}} - \left(\lambda_v \hat{J}_v + \lambda_\theta \hat{J}_\theta\right). \tag{3}$$

Next, we apply strict heuristic filters to eliminate fundamentally unsafe candidates, masking out trajectories that exhibit backward movement or severe off-road deviations. From the remaining valid trajectories, a spatial diversity-aware algorithm extracts the Top-$K$ candidates. These are fed into a Fine Scorer Decoder for rigorous evaluation using high-resolution BEV and image contexts. Finally, the highest-quality candidates undergo closed-loop predictive reranking via a Predictive Driver Model (PDM). Following the standard test-time search paradigm established by recent state-of-the-art methods on the NAVSIM benchmark (e.g., HydraMDP, DiffusionDriveV2), utilizing a lightweight simulator for final trajectory selection is crucial for ensuring absolute safety. Ultimately, the trajectory with the highest PDM score is selected and smoothed as the final executable output.

## 4 Experiments

In this section, we comprehensively evaluate the proposed STEER framework. We first detail the experimental setup, including the datasets, evaluation metrics, and implementation details. Subsequently, we compare our method against state-of-the-art baselines on the highly competitive

NAVSIM v1 and v2 leaderboards to demonstrate its superior planning performance. Finally, we conduct extensive ablation studies to validate the efficacy of each proposed architectural component and analyze the impact of critical hyperparameters.

## 4.1 Experimental Setup

### 4.1.1 Datasets

We evaluate our framework using NAVSIM (Navigation Simulation), a comprehensive autonomous driving benchmark built upon the massive nuPlan dataset Caesar et al. (2021), which encompasses over 1,200 hours of real-world driving data. Unlike traditional open-loop benchmarks that merely evaluate displacement errors (e.g., ADE/FDE), NAVSIM employs a high-fidelity, simulation-based Predictive Driver Model (PDM) Dauner et al. (2023) for closed-loop-like evaluation. This environment perfectly serves our objective: assessing whether generative diffusion models can synthesize trajectories that are not only statistically diverse but also strictly compliant with real-world physical and topological manifolds.

### 4.1.2 Evaluation Metrics

To provide a holistic assessment of safety, topological compliance, and kinematic feasibility, we report performance across both the NAVSIM v1 and the recently introduced, more rigorous NAVSIM v2 evaluation protocols. **NAVSIM v1 Metrics.** The primary metric is the **PDM Score** Dauner et al. (2024), a comprehensive metric formulated as a weighted aggregation of several critical sub-scores: Non-Collision (NC), Drivable Area Compliance (DAC), Time-to-Collision (TTC), Ego Progress (EP), and Comfort (C). These metrics evaluate fundamental safety, structural grounding, and longitudinal progress. **NAVSIM v2 Metrics.** While v1 captures basic safety, vanilla diffusion models often exploit its loopholes by generating mathematically collision-free but physically unexecutable trajectories with high-frequency jitter. To address this, the v2 protocol introduces stricter physical constraints, notably the Extended Comfort (EC) metric. EC heavily penalizes lateral acceleration and kinematic jerk, demanding precise continuous control. The refined EPDM score integrates EC and applies stricter thresholds for DAC and TTC.

### 4.1.3 Implementation Details

**Architecture.** We adopt an ImageNet-pretrained ResNet-34 He et al. (2016) as the shared image-LiDAR backbone. Camera inputs are resized to $1024 \times 256$. The vision encoder includes a 2-layer transformer (4 heads, expansion ratio 4). The BEV grid is $256 \times 256$, covering a $64\text{m} \times 64\text{m}$ ego-centric area ($[-32\text{m}, 32\text{m}]$ in $X, Y$). We detect up to 30 agents per frame, employing a GAI for agent interactions and gated centerline attention (initial logit $-5.0$). The planning transformer comprises 3 layers ($d = 256$, $d_{\text{ffn}} = 1024$, 8 heads) without dropout.

**Trajectory Generation.** Replacing fixed anchors, we propose 10 learnable, scene-conditioned anchors dynamically generated via ego and scene queries. To prevent mode collapse, we apply diversity regularization with a standard deviation warm-up over 5,000 steps. The model predicts 4 mode groups for a 4s horizon at 0.5s intervals (9 poses). Trajectory generation is formulated as a diffusion process parameterized by a 1D conditional U-Net (channels $[256, 512, 1024]$, kernel size 3, step embedding 256). We use 1,000 diffusion steps for training and 2 DDIM steps for inference.

**Training.** The network is optimized using AdamW (weight decay $10^{-4}$) and a MultiStepLR schedule decaying at epoch 70. The image encoder learning rate is scaled by $0.5$ for stable fine-tuning. Dropout is set to $0.1$ for all embedding, residual, and attention layers. The total loss is a weighted sum of trajectory $\mathcal{L}_1$ ($\lambda = 12$), classification ($\lambda = 10$), and regression ($\lambda = 8$); diffusion ($\lambda = 20$); anchor diversity ($\lambda = 0.01$); agent classification ($\lambda = 10$) and box regression ($\lambda = 1$); and BEV segmentation ($\lambda = 14$).

## 4.2 Main Results

We evaluate STEER on the competitive NAVSIM v1 and v2 `navtest` leaderboards (Tables 1 and 2).

Table 1: Quantitative results on the NAVSIM v1 `navtest` leaderboard. Our proposed STEER establishes a new state-of-the-art, outperforming all existing methods across the primary PDMS metric and most sub-metrics.

| Method | Reference | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|---|
| VADv2 Chen et al. (2024) | ICLR 2026 | 97.2 | 89.1 | 91.6 | **100** | 76.0 | 80.9 |
| UniAD Hu et al. (2023) | CVPR 2023 | 97.8 | 91.9 | 92.9 | **100** | 78.8 | 83.4 |
| Transfuser Chitta et al. (2022) | TPAMI 2022 | 97.7 | 92.8 | 92.8 | **100** | 79.2 | 84.0 |
| PARA-Drive Weng et al. (2024) | CVPR 2024 | 97.9 | 92.4 | 93.0 | 99.8 | 79.3 | 84.0 |
| DRAMA Yuan et al. (2024) | arxiv 2024 | 98.0 | 93.1 | 94.8 | **100** | 80.1 | 85.5 |
| GoalFlow Xing et al. (2025) | CVPR 2025 | 98.3 | 93.8 | 94.3 | **100** | 79.8 | 85.7 |
| HydraMDP Li et al. (2024) | arxiv 2024 | 98.3 | 96.0 | 94.6 | **100** | 78.7 | 86.5 |
| HydraMDP++ Li et al. (2025a) | arxiv 2025 | 97.6 | 96.0 | 93.1 | **100** | 80.4 | 86.6 |
| ARTEMIS Feng et al. (2025) | RAL 2026 | 98.3 | 95.1 | 94.3 | **100** | 81.4 | 87.0 |
| FUMP Liu et al. (2025) | arxiv 2025 | 98.1 | 96.2 | 94.2 | **100** | 82.0 | 87.8 |
| DiffusionDrive Liao et al. (2025) | CVPR 2025 | 98.2 | 96.2 | 94.7 | **100** | 82.2 | 88.1 |
| WoTE Li et al. (2025b) | ICCV 2025 | 98.5 | 96.8 | 94.9 | 99.9 | 81.9 | 88.3 |
| DIVER Song et al. (2025) | arxiv 2025 | 98.5 | 96.5 | 94.9 | **100** | 82.6 | 88.3 |
| DriveSuprim Yao et al. (2026) | AAAI 2026 | 97.8 | 97.3 | 93.6 | **100** | 86.7 | 89.9 |
| DiffusionDriveV2 Zou et al. (2025) | arxiv 2025 | 98.3 | 97.9 | 94.8 | 99.9 | 87.5 | 91.2 |
| ipad Guo et al. (2025) | arxiv 2025 | 98.6 | 98.3 | 94.9 | **100** | 88.0 | 91.7 |
| SparseDriveV2 Sun et al. (2026) | arxiv 2026 | 98.5 | 98.4 | 95.0 | 99.9 | 88.6 | 92.0 |
| STEER | - | **99.0** | **99.1** | **96.8** | **100** | **90.1** | **93.8** |

Table 2: Quantitative results on the NAVSIM v2 `navtest` leaderboard. Evaluated under extended and more rigorous metrics, STEER consistently achieves the highest overall EPDMS, demonstrating superior safety and driving efficiency.

| Method | NC ↑ | DAC ↑ | DDC ↑ | TLC ↑ | EP ↑ | TTC ↑ | LK ↑ | HC ↑ | EPDMS ↑ |
|---|---|---|---|---|---|---|---|---|---|
| VADv2 Chen et al. (2024) | 97.3 | 91.7 | 98.2 | 99.9 | 77.6 | 92.7 | 66.0 | **100** | 76.6 |
| TransFuser Chitta et al. (2022) | 97.7 | 92.8 | 98.3 | 99.9 | 79.2 | 92.8 | 67.6 | **100** | 77.8 |
| HydraMDP Li et al. (2024) | 97.5 | 96.3 | 98.3 | 99.6 | 80.1 | 93.0 | 65.5 | **100** | 79.8 |
| HydraMDP++ Li et al. (2025a) | 97.9 | 96.5 | 98.9 | **100** | 79.2 | 93.4 | 67.2 | **100** | 80.6 |
| ARTEMIS Feng et al. (2025) | 98.3 | 95.1 | 98.6 | 99.8 | 81.5 | 97.4 | 96.5 | **100** | 83.1 |
| DriveSuprim Yao et al. (2026) | 97.5 | 96.5 | 99.4 | 99.6 | 88.4 | 96.6 | 95.5 | 98.3 | 83.1 |
| DiffusionDrive Liao et al. (2025) | 98.2 | 95.9 | 99.4 | 99.8 | 87.5 | 97.3 | 96.8 | 98.3 | 84.5 |
| DiffusionDriveV2 Zou et al. (2025) | 97.7 | 96.6 | 99.2 | 99.8 | 88.9 | 97.2 | 96.0 | 97.8 | 85.5 |
| SparseDriveV2 Sun et al. (2026) | 98.1 | 98.1 | 99.6 | 99.8 | 91.1 | 97.3 | 96.9 | 98.2 | 86.7 |
| STEER | **99.0** | **98.8** | 98.5 | 99.5 | **91.7** | **98.7** | 94.2 | 95.5 | **87.2** |

**Results on NAVSIM v1.** STEER establishes a new state-of-the-art with a PDMS of 93.8, outperforming SparseDriveV2 Sun et al. (2026) by a significant +1.8 margin. It sets new records in critical safety metrics (NC 99.0, DAC 99.1, TTC 96.8) without sacrificing driving efficiency (EP 90.1).

**Results on NAVSIM v2.** Under the more rigorous v2 protocol, STEER achieves the highest EPDMS of 87.2. While some baselines adopt overly conservative strategies to maximize specific sub-metrics (e.g., LK, HC), STEER strikes the optimal balance between aggressive progress (highest EP) and strict safety (highest NC and TTC). To visually corroborate this superior kinematic feasibility and structural compliance, Appendix C provides qualitative comparisons demonstrating STEER's ability to eliminate high-frequency jitter and centerline deviations in complex interactive scenarios.

### 4.3 Ablation Studies

#### 4.3.1 Efficacy of Core Components

To validate the effectiveness of our proposed modules, we conduct a progressive ablation study on the NAVSIM dataset, as detailed in Table 3. Starting from a vanilla diffusion baseline with a PDMS of 89.2, integrating the SCA module improves the PDMS to 89.9 and the EP to 84.5, demonstrating that topology-aware initialization eases optimization and slightly improves inference speed to 13.97 FPS. Replacing standard global attention with our GAI module further boosts the PDMS to 90.7 and the TTC to 95.2, proving that a distance-aware interaction graph effectively eliminates spurious correlations while maintaining high computational efficiency. The subsequent addition of the ADPT module yields a substantial performance leap, driving the PDMS to 92.0 and achieving a perfect

Comf score of 100, which validates its crucial role in ensuring strict kinematic compliance through autoregressive refinement. Finally, equipping the HTS mechanism pushes the overall performance to a state-of-the-art PDMS of 93.8. By rigorously filtering candidates against physical boundaries, HTS maximizes critical safety metrics with an NC of 99.0 and a DAC of 99.1. Although evaluating multiple candidates increases latency, the final system still operates at a viable 7.79 FPS, striking an optimal balance between uncompromising driving safety and real-time deployment requirements.

Table 3: Progressive architectural ablation on the NAVSIM dataset. We sequentially integrate Scene-Conditioned Anchor (SCA), Graph Attention-based Agent Interaction (GAI), Adaptive Dual-Path Trajectory Refinement (ADPT), and Hierarchical Trajectory Selection (HTS) into a vanilla diffusion baseline. The consistent improvements across all primary metrics validate the necessity of injecting physical and structural priors into each stage of the generative pipeline. Latency is measured on a single NVIDIA A100 GPU.

| SCA | GAI | ADPT | HTS | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ | Latency (ms) ↓ | FPS ↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 98.3 | 98.0 | 94.3 | 99.2 | 83.4 | 89.2 | 72.56 | 13.54 |
| ✓ | | | | 98.4 | 98.3 | 94.4 | 99.6 | 84.5 | 89.9 | 71.56 | 13.97 |
| ✓ | ✓ | | | 98.5 | 97.7 | 95.2 | 99.6 | 86.4 | 90.7 | 71.27 | 14.03 |
| ✓ | ✓ | ✓ | | 98.4 | 97.7 | 94.7 | 100 | 87.5 | 92.0 | 79.06 | 12.65 |
| ✓ | ✓ | ✓ | ✓ | **99.0** | **99.1** | **96.8** | **100** | **90.1** | **93.8** | 128.36 | 7.79 |

### 4.3.2 Efficacy of the Dual-Path Design

To validate the necessity of the ADPT module, we compare our full dual-path design against a baseline utilizing only the one-shot refinement branch, as detailed in Table 4. The results indicate that relying solely on the one-shot branch leads to overly conservative driving behavior. While it achieves marginally higher safety sub-scores by playing it safe, it suffers from a significantly lower EP score of 85.0, resulting in a suboptimal overall PDMS of 90.4. By dynamically fusing the one-shot predictions with the segment-wise autoregressive branch, our model gains the fine-grained kinematic confidence required to navigate complex scenarios efficiently rather than simply yielding. Consequently, the dual-path fusion substantially boosts the EP to 87.5 and the overall PDMS to 92.0, all while maintaining a perfect Comfort score of 100. This demonstrates that the ADPT module successfully reconciles global structural planning with assertive and smooth local execution.

Table 4: Efficacy of the ADPT module. We compare our dual-path design against using only the one-shot branch. Relying solely on the one-shot branch results in overly conservative driving behavior, evidenced by a significantly lower EP score. By integrating the autoregressive branch, our dual-path fusion effectively boosts longitudinal progress and the overall PDMS while maintaining perfect kinematic comfort.

| Configuration | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|
| Only One-shot Branch | 98.5 | 97.9 | 95.5 | 100 | 85.0 | 90.4 |
| Dual-Path Fusion | 98.4 | 97.7 | 94.7 | 100 | 87.5 | 92.0 |

### 4.3.3 Impact of Progressive Trajectory Selection

To thoroughly evaluate the HTS mechanism and dispel potential concerns regarding metric dependency on the PDM simulator, we conduct a progressive ablation on the selection strategies, as shown in Table 5. When relying solely on the learned neural scorer without any explicit physical filtering, the model achieves a baseline PDMS of 92.0. By introducing our proposed kinematic penalties and hard topological constraints, the performance significantly improves to a PDMS of 92.8, with notable gains in safety metrics (NC 98.8, DAC 98.5). This crucial intermediate result demonstrates that our diffusion pipeline inherently generates high-quality, physically feasible manifolds, and the performance gain is not merely a byproduct of test-time search. Finally, incorporating the PDM reranking as a deterministic safety guardrail, a standard practice in recent state-of-the-art methods, further pushes the boundaries, maximizing critical safety and progress metrics to achieve the ultimate PDMS of 93.8.

Table 5: Impact of progressive trajectory selection strategies. To dispel concerns regarding metric dependency, we ablate the final PDM reranking step. Relying solely on our proposed kinematic penalties and hard constraints already yields a highly competitive PDMS of 92.8, proving that our diffusion pipeline inherently generates physically feasible manifolds.

| Selection Strategy | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|
| Baseline (Neural Scorer Only) | 98.4 | 97.7 | 94.7 | **100** | 87.5 | 92.0 |
| + Kinematic Penalty & Hard Constraints | 98.8 | 98.5 | 95.9 | **100** | 89.2 | 92.8 |
| + PDM Reranking (Full HTS) | **99.0** | **99.1** | **96.8** | **100** | **90.1** | **93.8** |

#### 4.3.4 Effectiveness of Topology-Aware Initialization

To validate the fundamental motivation behind our SCA module, we compare different diffusion initialization strategies in Table 6. Initiating the reverse diffusion process from pure isotropic Gaussian noise yields the lowest overall performance (PDMS 88.4). Without any structural priors, the unconstrained generative manifold struggles to align with complex road topologies, which is evident from its poor DAC and EP. Adopting static K-Means anchors, a common practice in previous diffusion planners, provides a better starting point, improving the PDMS to 90.5. However, their rigid, scene-agnostic nature fundamentally limits their adaptability to diverse or out-of-distribution intersections. In stark contrast, our proposed SCA dynamically adapts the initial proposals to the specific environmental context. This topology-aware initialization provides a highly accurate starting manifold, significantly boosting the DAC to 99.1 and the EP to 90.1. Ultimately, SCA achieves a state-of-the-art PDMS of 93.8, proving that injecting geometric and topological priors at the very beginning of the diffusion pipeline is crucial for safe and efficient trajectory generation. Extended ablations validating the internal designs of these modules are detailed in Appendix B.

Table 6: Ablation on initialization strategies. We compare our Scene-Conditioned Anchor (SCA) against standard initialization methods. SCA significantly outperforms both pure Gaussian noise and static K-Means anchors, demonstrating the necessity of injecting topology-aware priors for complex driving scenarios.

| Initialization Method | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|
| Pure Gaussian Noise | 97.5 | 96.2 | 93.0 | 98.5 | 85.1 | 88.4 |
| Static K-Means Anchors | 98.1 | 97.8 | 94.5 | 99.2 | 87.6 | 90.5 |
| Scene-Conditioned Anchors (Ours) | **99.0** | **99.1** | **96.8** | **100** | **90.1** | **93.8** |

## 5 Conclusion and Limitations

We presented STEER, a principled end-to-end autonomous driving framework that bridges the gap between flexible diffusion-based trajectory generation and the geometric and kinematic requirements of real-world driving. By progressively injecting structural and motion priors through SCA, GAI, ADPT, and HTS, STEER generates diverse trajectories while maintaining geometric consistency and physical feasibility. Extensive experiments on the NAVSIM benchmarks demonstrate the effectiveness of this design, with STEER achieving state-of-the-art performance while maintaining strong planning efficiency.

Despite these promising results, STEER still has several limitations. First, its reliance on 2D BEV representations may limit its ability to model complex 3D interactions, such as multi-level roads and overpasses. Incorporating richer 3D representations, such as occupancy networks, is a natural direction for future work. Second, extreme out-of-distribution scenarios remain challenging. Future research could therefore explore uncertainty-aware fallback mechanisms to identify unfamiliar situations and enable more conservative and reliable driving behaviors.

## AI ASSISTANCE

Large language models were used to assist with coding, language polishing, drafting manuscript sections, suggesting relevant literature, and discussing methodological choices. The use of LLMs as experimental reasoning backends is described in the methodology and experimental setup.

## REFERENCES

Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.

Vibha Bharilya and Neetesh Kumar. Machine learning for autonomous vehicle's trajectory prediction: A comprehensive survey, challenges, and future research directions. *Vehicular Communications*, 46:100733, 2024.

Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.

Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *arXiv preprint arXiv:1910.05449*, 2019.

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.

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.

Shounak Datta and Dhanasekar Sundararaman. Evaluating hallucination in large vision-language models based on context-aware object similarities. *arXiv preprint arXiv:2501.15046*, 2025.

Daniel Dauner, Marcel Hallgarten, Andreas Geiger, and Kashyap Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Conference on Robot Learning*, pp. 1268–1281. PMLR, 2023.

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.

Renju Feng, Ning Xi, Duanfeng Chu, Rukang Wang, Zejian Deng, Anzheng Wang, Liping Lu, Jinxiang Wang, and Yanjun Huang. Artemis: Autoregressive end-to-end trajectory planning with mixture of experts for autonomous driving. *IEEE Robotics and Automation Letters*, 11(1):226–233, 2025.

Ke Guo, Haochen Liu, Xiaojun Wu, Jia Pan, and Chen Lv. ipad: Iterative proposal-centric end-to-end autonomous driving. *arXiv preprint arXiv:2505.15111*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

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*, pp. 17853–17862, 2023.

Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8306–8316. IEEE, 2023a.

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*, pp. 9644–9653, 2023b.

Chiyu M Jiang, Yijing Bai, Andre Cornman, Christopher Davis, Xiukun Huang, Hong Jeon, Sakshum Kulshrestha, John Lambert, Shuangyu Li, Xuanyu Zhou, et al. Scenediffuser: Efficient and controllable driving simulation initialization and rollout. *Advances in Neural Information Processing Systems*, 37:55729–55760, 2024.

Hao Jiang, Zhipeng Zhang, Yu Gao, Zhigang Sun, Yiru Wang, Yuwen Heng, Shuo Wang, Jinhao Chai, Zhuo Chen, Hao Zhao, et al. Flowdrive: Energy flow field for end-to-end autonomous driving. *arXiv preprint arXiv:2509.14303*, 2025.

Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 2165–2174. IEEE, 2017.

Kailin Li, Zhenxin Li, Shiyi Lan, Yuan Xie, Zhizhong Zhang, Jiayi Liu, Zuxuan Wu, Zhiding Yu, and Jose M Alvarez. Hydra-mdp++: Advancing end-to-end driving via expert-guided hydra-distillation. *arXiv preprint arXiv:2503.12820*, 2025a.

Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27137–27146, 2025b.

Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. *arXiv preprint arXiv:2506.08052*, 2025c.

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.

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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12037–12047, 2025.

Lin Liu, Caiyan Jia, Ziying Song, Hongyu Pan, Bencheng Liao, Wenchao Sun, Yongchang Zhang, Lei Yang, and Yandan Luo. Fully unified motion planning for end-to-end autonomous driving. *arXiv preprint arXiv:2504.12667*, 2025.

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European conference on computer vision*, pp. 531–548. Springer, 2022.

Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. *arXiv preprint arXiv:2207.05844*, 2022.

Jiquan Ngiam, Benjamin Caine, Vijay Vasudevan, Zhengdong Zhang, Hao-Tien Lewis Chiang, Jeffrey Ling, Rebecca Roelofs, Alex Bewley, Chenxi Liu, Ashish Venugopal, et al. Scene transformer: A unified architecture for predicting multiple agent trajectories. *arXiv preprint arXiv:2106.08417*, 2021.

Tung Phan-Minh, Elena Corina Grigore, Freddy A Boulton, Oscar Beijbom, and Eric M Wolff. Covernet: Multimodal behavior prediction using trajectory sets. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14074–14083, 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Ziying Song, Lin Liu, Hongyu Pan, Bencheng Liao, Mingzhe Guo, Lei Yang, Yongchang Zhang, Shaoqing Xu, Caiyan Jia, and Yadan Luo. Diver: Reinforced diffusion breaks imitation bottlenecks in end-to-end autonomous driving. *arXiv preprint arXiv:2507.04049*, 2025.

Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8795–8801. IEEE, 2025.

Wenchao Sun, Xuewu Lin, Keyu Chen, Zixiang Pei, Xiang Li, Yining Shi, and Sifa Zheng. Sparsedrivev2: Scoring is all you need for end-to-end autonomous driving. *arXiv preprint arXiv:2603.29163*, 2026.

Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15449–15458, 2024.

Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long, and Wei Yin. Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 1602–1611, 2025.

Miao Xu, Zhi Liu, Bingyi Wang, and Shengyan Li. A survey of autonomous driving trajectory prediction: Methodologies, challenges, and future prospects. *Machines*, 13(9), 2025.

Wenhao Yao, Zhenxin Li, Shiyi Lan, Zi Wang, Xinglong Sun, Jose M Alvarez, and Zuxuan Wu. Drivesuprim: Towards precise trajectory selection for end-to-end planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11910–11918, 2026.

Chengran Yuan, Zhanqi Zhang, Jiawei Sun, Shuo Sun, Zefan Huang, Christina Dao Wen Lee, Dongen Li, Yuhang Han, Anthony Wong, Keng Peng Tee, et al. Drama: An efficient end-to-end motion planner for autonomous driving with mamba. *arXiv preprint arXiv:2408.03601*, 2024.

Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9813–9823, 2021.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. Siren's song in the ai ocean: A survey on hallucination in large language models. *Computational Linguistics*, 51(4):1373–1418, 2025.

Yinan Zheng, Ruiming Liang, Kexin Zheng, Jinliang Zheng, Liyuan Mao, Jianxiong Li, Weihao Gu, Rui Ai, Shengbo Li, Xianyuan Zhan, et al. Diffusion-based planning for autonomous driving with flexible guidance. In *International conference on learning representations*, volume 2025, pp. 37207–37227, 2025.

Jialv Zou, Shaoyu Chen, Bencheng Liao, Zhiyu Zheng, Yuehao Song, Lefei Zhang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Diffusiondrivev2: Reinforcement learning-constrained truncated diffusion modeling in end-to-end autonomous driving. *arXiv preprint arXiv:2512.07745*, 2025.

# TECHNICAL APPENDICES AND SUPPLEMENTARY MATERIAL

This supplementary material provides further details to support the main text. Sec. A details the comprehensive implementation specifics of our method and the evaluated baselines. Sec. B presents extended quantitative results and additional ablation analyses. Finally, Sec. C illustrates supplementary qualitative visualizations, demonstrating the effectiveness of our framework in complex interactive scenarios.

## A ADDITIONAL IMPLEMENTATION DETAILS

### A.1 SCENE-CONDITIONED ANCHOR GENERATION

We introduced the SCA mechanism to replace static, scene-agnostic priors with dynamically adapted initializations. Here, we provide the formal mathematical formulations, specific architectural configurations, and a detailed discussion of its theoretical advantages when integrated into the diffusion pipeline.

#### A.1.1 FORMAL FORMULATION AND ARCHITECTURAL SPECIFICS

**Contextual Feature Aggregation.** The ego queries $Q_{ego} \in \mathbb{R}^{N\times d}$ extracted from the multi-layer Transformer decoder encapsulate rich, high-dimensional spatio-temporal information. Specifically, these queries encode the static road geometry, intersection topological structures, surrounding agents' kinematic intents, and the ego-vehicle's historical states. To obtain a holistic understanding of the current driving context, the global scene representation $q_{scene}$ is formally aggregated via mean pooling along the query dimension:

$$q_{scene} = \frac{1}{N}\sum_{i=1}^{N} Q_{ego}^{(i)}. \tag{4}$$

**Anchor Conditioner Architecture.** The Anchor Conditioner is parameterized as a lightweight Multi-Layer Perceptron (MLP) equipped with a ReLU activation function. It maps the global scene feature to the scene-dependent offset tensor $\Delta\mu \in \mathbb{R}^{L\times T\times 2}$:

$$\Delta\mu = \text{MLP}(q_{scene}). \tag{5}$$

A critical architectural design is the zero-initialization of the weights and biases in the final linear layer of this MLP. This guarantees that at the beginning of the training phase, the predicted offset $\Delta\mu$ is strictly zero, acting as an identity mapping ($\mu_{scene} = \mu_{base}$). This design preserves the prior distribution of the base learnable anchors in the early stages, preventing gradient instability and mode collapse.

**Standard Deviation Parameterization.** During the training phase's reparameterization trick ($A = \mu_{scene} + \sigma_{base} \odot \epsilon$), the learnable standard deviation $\sigma_{base}$ is not optimized directly to avoid negative values. Instead, it is parameterized in the log-space:

$$\sigma_{base} = \exp(\log \sigma_{learnable}), \tag{6}$$

where $\log \sigma_{learnable} \in \mathbb{R}^{L\times T\times 2}$ is a freely optimizable parameter tensor.

#### A.1.2 INTEGRATION AND THEORETICAL ADVANTAGES

The generated scene-conditioned anchors $A$ are subsequently fed into the DDIM denoising loop as the starting state $\mathbf{Y}^{\mathcal{T}}$. Compared to traditional fixed anchors derived from offline clustering (e.g., K-Means on the training set), our dynamic SCA mechanism offers three distinct advantages:

- **Scene-Aware Initialization:** Instead of forcing the diffusion model to denoise from a generic, average trajectory, SCA provides a highly accurate starting point tailored to the specific road topology (e.g., curving lanes) and current traffic conditions.
- **Accelerated Convergence:** By starting closer to the target data manifold, the high-quality initialization effectively reduces the generative distance, thereby requiring significantly fewer denoising iterations to achieve kinematic smoothness.

- **Enhanced Robustness for OOD Scenarios:** Static anchors often struggle to generalize to complex, out-of-distribution (OOD) or long-tail driving scenarios (e.g., irregular intersections). By learning scene-dependent diversity E2E, SCA ensures that the initial proposals remain topologically valid even in rare environments.

### A.2 Graph Attention-based Agent Interaction

We introduced the Graph Attention-based Agent Interaction (GAI) module to enforce spatial locality and mitigate attention dilution. Here, we provide the comprehensive architectural diagram (illustrated in Fig. A.1) and elaborate on the rigorous tensor operations and hyperparameter configurations omitted from the main text due to space constraints.

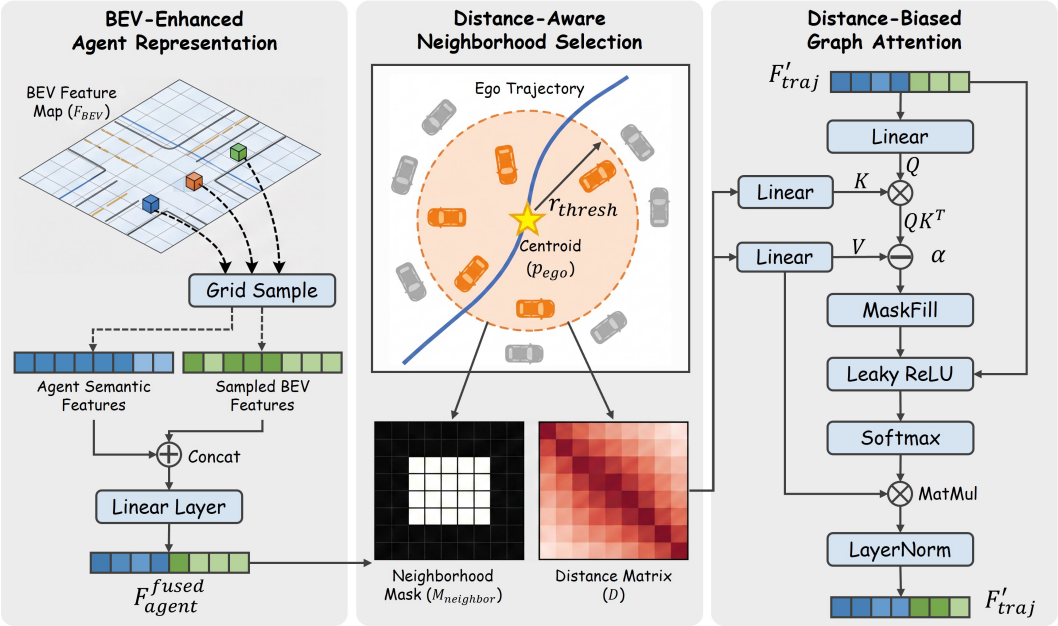



Figure A.1: **Detailed architecture of the GAI module.** The module comprises three sequential operations: (a) BEV-Enhanced Agent Representation via grid sampling; (b) Distance-Aware Neighborhood Selection using trajectory centroids; and (c) Distance-Biased Graph Attention with a learnable spatial decay prior.

#### A.2.1 BEV-Enhanced Agent Representation

While the main text briefly mentions bilinear grid sampling, the exact feature fusion process is executed as follows. Given the pure semantic agent features $\mathbf{F}_{\text{agent}} \in \mathbb{R}^{B\times A\times C}$ and the high-dimensional BEV feature map $\mathbf{F}_{\text{BEV}} \in \mathbb{R}^{B\times C_{\text{BEV}}\times H\times W}$, we first project the BEV features to the model dimension $C$ via a $1\times 1$ convolution $\phi_{\text{BEV}}$. We then sample the local visual context at each agent's normalized continuous coordinate $\mathbf{p}_{\text{norm}} \in [-1,1]^2$:

$$\mathbf{F}^{\text{sample}}_{\text{BEV}} = \text{GridSample}(\phi_{\text{BEV}}(\mathbf{F}_{\text{BEV}}), \mathbf{p}_{\text{norm}}) \in \mathbb{R}^{B\times A\times C}. \tag{7}$$

To capture the semantic-spatial coupling, the sampled context is concatenated with the intrinsic agent features and fused via a linear projection layer $\phi_{\text{fuse}}$:

$$\mathbf{F}^{\text{fused}}_{\text{agent}} = \phi_{\text{fuse}}([\mathbf{F}_{\text{agent}}, \mathbf{F}^{\text{sample}}_{\text{BEV}}]) \in \mathbb{R}^{B\times A\times C}. \tag{8}$$

#### A.2.2 Dynamic Neighborhood Masking

To construct the distance-aware mask mentioned in the main text, we must dynamically track the spatial relationship between the ego vehicle's planned trajectories and the surrounding agents. Let

$\mathbf{p}_{\text{traj}} \in \mathbb{R}^{B\times GN\times T\times 2}$ denote the trajectory waypoints for $G$ goals and $N$ samples over $T$ timesteps. We compute the spatial centroid $\mathbf{p}_{\text{ego}}$ of each trajectory mode by averaging across the temporal dimension:

$$\mathbf{p}_{\text{ego}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{p}_{\text{traj}}[:,:,t,:] \in \mathbb{R}^{B\times GN\times 2}. \tag{9}$$

Subsequently, we compute the pairwise Euclidean distance matrix $\mathbf{D} \in \mathbb{R}^{B\times GN\times A}$ between $\mathbf{p}_{\text{ego}}$ and the agent positions $\mathbf{p}_{\text{agent}}$. The dynamic neighborhood mask $\mathbf{M}_{\text{neighbor}}$ is then formulated to filter out agents beyond the spatial threshold $r_{\text{thresh}}$ (empirically set to $20.0$ meters) and to exclude invalid padded agents ($\mathbf{M}_{\text{valid}}$):

$$\mathbf{M}_{\text{neighbor}} = (\mathbf{D} < r_{\text{thresh}}) \wedge \mathbf{M}_{\text{valid}}. \tag{10}$$

### A.2.3 Attention Computation and Post-processing

Building upon the final attention weight equation provided in the main text, the intermediate distance-biased logits $\mathbf{S}_{\text{dist}}$ are computed by subtracting the scaled distance matrix from the standard scaled dot-product:

$$\mathbf{S}_{\text{dist}} = \frac{\mathbf{Q}_h\mathbf{K}_h^T}{\sqrt{d_h}} - \alpha\cdot\mathbf{D}. \tag{11}$$

In our implementation, the learnable temperature parameter $\alpha \in \mathbb{R}^+$ is initialized to $1.0$, and the LeakyReLU activation utilizes a negative slope of $\beta = 0.2$. This specific activation design is crucial: unlike a standard Softmax that aggressively suppresses negative logits, the LeakyReLU allows weak gradients to propagate for low-attention agents, significantly improving training stability under sparse attention patterns. Finally, the aggregated attention output is processed through a linear projection, Dropout ($p = 0.1$), and a residual connection with Layer Normalization to yield the enhanced trajectory features.

## A.3 Adaptive Dual-Path Trajectory Refinement

While Sec. 3.3 of the main text provides a high-level overview of the ADPT module, this section details its rigorous mathematical formulation, architectural specifics, and the design choices made to ensure computational efficiency during the iterative diffusion process.

Given the interaction feature $\mathbf{F}_{traj} \in \mathbb{R}^{N\times C}$ and the intermediate trajectory $\mathbf{P}_{noisy} \in \mathbb{R}^{N\times T\times 2}$ predicted by the diffusion decoder (where $N$ is the number of modes and $T$ is the future time horizon), the ADPT module generates and adaptively fuses two candidate trajectories: $\mathbf{P}_A$ (one-shot) and $\mathbf{P}_B$ (autoregressive).

### A.3.1 One-Shot Refinement Branch

The one-shot branch $\mathcal{M}_{\text{one-shot}}$ serves as the baseline path, providing fast, globally consistent structural predictions. Parameterized as a Multi-Layer Perceptron (MLP), it directly maps the high-dimensional interaction feature $\mathbf{F}_{traj}$ to the trajectory residuals $\Delta\mathbf{P}_A \in \mathbb{R}^{N\times T\times 3}$ (comprising $x, y$ coordinates and heading angle $\theta$) for all $T$ time steps simultaneously:

$$\mathbf{P}_A = \mathbf{P}_{noisy} \oplus \mathcal{M}_{\text{one-shot}}(\mathbf{F}_{traj}), \tag{12}$$

where $\oplus$ denotes element-wise residual addition. For the heading angle $\theta$, we apply a $\tanh$ normalization scaled by $\pi$ to bound the predictions. Crucially, to ensure an identity mapping at the beginning of the training phase and prevent early divergence, the weights and biases of the final regression layer in $\mathcal{M}_{\text{one-shot}}$ are strictly zero-initialized.

### A.3.2 Progressive Autoregressive Branch

Standard autoregressive architectures (e.g., RNNs or GRUs) introduce severe sequential bottlenecks, making them computationally prohibitive within a diffusion denoising loop. To capture fine-grained temporal kinematics efficiently, we propose a lightweight **Segment-wise Context-aware** mechanism.

We divide the $T$ waypoints into $K$ consecutive segments, each containing $S = T/K$ points. In our implementation for a 4-second horizon at 2Hz, $T = 8$, $K = 4$, and $S = 2$. For the $k$-th segment, the progressive refinement unfolds as follows:

**1. Dynamic BEV Re-querying:** As the trajectory is progressively refined, the spatial footprint of the ego vehicle shifts. We utilize the initial coordinates of the current segment $\mathbf{P}_{noisy}^{(k)}$ to re-query local spatial features $\mathbf{F}_{seg}^{(k)}$ from the global BEV feature map $\mathbf{F}_{BEV}$ via bilinear grid sampling.

**2. Context Fusion:** To replace the heavy hidden state transitions of RNNs, we employ a lightweight MLP. The context $\mathbf{C}^{(k)}$ for the current segment is formed by concatenating the re-queried BEV feature $\mathbf{F}_{seg}^{(k)}$, the global trajectory feature $\mathbf{F}_{global}$ (obtained via mean pooling of $\mathbf{F}_{traj}$), and the sinusoidal positional encoding of the refined output from the previous segment $\text{PE}(\mathbf{P}_B^{(k-1)})$:

$$\mathbf{C}^{(k)} = \text{MLP}_{\text{context}}\left(\left[\mathbf{F}_{seg}^{(k)}, \mathbf{F}_{global}, \text{PE}(\mathbf{P}_B^{(k-1)})\right]\right). \tag{13}$$

**3. Local Residual Prediction:** Combining the local spatial features and the temporal context, a regression head predicts the residuals for the current segment:

$$\Delta\mathbf{P}_B^{(k)} = \text{RegHead}\left(\left[\mathbf{F}_{seg}^{(k)}, \mathbf{C}^{(k)}\right]\right), \quad \mathbf{P}_B^{(k)} = \mathbf{P}_{noisy}^{(k)} \oplus \Delta\mathbf{P}_B^{(k)}. \tag{14}$$

By operating on segments ($K = 4$) rather than individual waypoints ($T = 8$), this branch halves the sequential computational graph depth, maintaining high inference efficiency.

### A.3.3 Adaptive Trajectory Scorer and Geometric Features

The one-shot branch excels at global topological positioning, whereas the AR branch is superior at local kinematic smoothness. To dynamically balance these properties, the Adaptive Trajectory Scorer computes fusion weights based on the *geometric morphological features* of the candidates.

Specifically, the geometric encoder $\mathcal{E}_{geo}$ is designed to explicitly capture kinematic feasibility. It computes the discrete first-order and second-order derivatives (approximating velocity $\mathbf{v}$ and acceleration $\mathbf{a}$) of the candidate trajectories along the temporal dimension. These kinematic profiles are processed by a 1D Convolutional layer to yield the morphological embeddings $\mathbf{E}_A$ and $\mathbf{E}_B$:

$$\mathbf{E}_A = \mathcal{E}_{geo}(\mathbf{P}_A), \quad \mathbf{E}_B = \mathcal{E}_{geo}(\mathbf{P}_B). \tag{15}$$

Subsequently, the original interaction feature is concatenated with these geometric embeddings and passed through a scoring MLP to output normalized weights $w_A$ and $w_B$:

$$[w_A, w_B] = \text{Softmax}\left(\text{MLP}_{score}\left([\mathbf{F}_{traj}, \mathbf{E}_A, \mathbf{E}_B]\right)\right). \tag{16}$$

The final refined trajectory is obtained via the weighted sum $\mathbf{P}_{final} = w_A\mathbf{P}_A + w_B\mathbf{P}_B$. This dynamic gating mechanism allows the model to rely on the AR branch for complex, high-curvature maneuvers requiring strict smoothness, while leaning on the one-shot branch for straightforward, high-speed cruising.

## B Ablation Studies

### B.0.1 Impact of Denoising Steps

We investigate the impact of the number of diffusion denoising steps on planning quality and convergence speed, as detailed in Table A.1. The results demonstrate that our model maintains highly stable and robust performance across varying step counts. Notably, even with a single denoising step, the model achieves a strong PDMS of 93.2. We observe that setting the denoising steps to 2 provides the optimal trade-off between planning accuracy and inference efficiency. Under this configuration, the model achieves an outstanding PDMS of 93.8 and the highest EP of 90.1. Although increasing the steps to 7 yields a marginal improvement in PDMS to 94.0, it inherently incurs a linear increase in computational overhead due to the iterative nature of the diffusion process. The ability to achieve top-tier performance with only 2 denoising steps strongly validates the effectiveness of our proposed SCA module. By providing a highly accurate and context-aware initialization, the model significantly accelerates the diffusion convergence process, eliminating the need for lengthy iterative refinement during inference.

Table A.1: Ablation study on the number of denoising steps. Setting the denoising steps to 2 yields an optimal balance.

| Steps (T) | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|
| 1 | 99.1 | 98.9 | 96.0 | 100 | 89.6 | 93.2 |
| 2 | **99.0** | **99.1** | **96.8** | **100** | **90.1** | **93.8** |
| 3 | 99.2 | 98.3 | 96.3 | 100 | 89.6 | 93.0 |
| 4 | 99.0 | 98.2 | 96.8 | 100 | 89.5 | 93.1 |
| 5 | 99.2 | 98.6 | 97.2 | 99.8 | 89.8 | 93.5 |
| 6 | 98.8 | 98.8 | 96.5 | 100 | 89.4 | 93.2 |
| 7 | 99.0 | 99.4 | 97.0 | 99.9 | 90.0 | 94.0 |
| 8 | 98.9 | 99.0 | 97.3 | 100 | 89.3 | 93.6 |
| 9 | 99.2 | 98.9 | 97.4 | 100 | 89.2 | 93.6 |
| 10 | 99.0 | 99.0 | 96.9 | 100 | 89.3 | 93.6 |

Table A.2: Ablation study on the number of trajectory modes.

| Nums | NC ↑ | DAC ↑ | TTC ↑ | Comf. ↑ | EP ↑ | PDMS ↑ |
|---|---|---|---|---|---|---|
| 1 | 98.5 | 97.6 | 96.1 | 100 | 87.5 | 91.5 |
| 3 | 98.8 | 98.5 | 95.6 | 100 | 89.4 | 92.6 |
| 5 | 98.5 | 98.2 | 96.7 | 100 | 88.9 | 92.7 |
| 7 | 98.8 | 98.6 | 95.6 | 100 | 89.0 | 92.6 |
| 10 | 98.9 | 98.9 | 96.4 | 100 | 89.6 | 93.3 |
| 13 | 99.0 | 98.3 | 96.7 | 100 | 89.6 | 93.2 |
| 15 | **99.2** | 98.5 | 96.8 | 100 | 89.5 | 93.3 |
| 17 | **99.2** | 98.6 | 96.2 | 100 | 90.0 | 93.3 |
| 20 | 99.0 | **99.1** | **96.8** | 100 | **90.1** | **93.8** |
| 23 | 98.8 | 98.6 | 96.3 | 100 | 89.9 | 93.2 |
| 25 | 98.9 | 98.1 | 95.9 | 100 | 89.3 | 92.5 |

### B.0.2 Impact of Trajectory Modes

We conduct an ablation study to investigate the impact of the number of trajectory modes on planning quality, as presented in Table A.2. The results indicate a clear correlation between the number of generated modes and the overall driving performance. When utilizing only a single mode, the model achieves a baseline PDMS of 91.5. As the number of modes increases, the model captures a richer set of future possibilities and multimodal behaviors, leading to steady improvements across all safety and progress metrics. The planning performance peaks when the number of modes is set to 20. Under this configuration, the model achieves the highest PDMS of 93.8, along with the best scores with a DAC of 99.1, a TTC of 96.8, and an EP of 90.1. While further increasing the modes to 23 or 25 inherently increases the computational burden, it paradoxically results in a performance degradation, with the PDMS dropping to 92.5. We attribute this decline to the increased difficulty in optimizing an excessively large set of trajectories, which may introduce redundant or conflicting proposals during the diffusion process and distract the model from the optimal path. Therefore, we adopt 20 modes as the default setting to ensure maximum planning accuracy.

## C Qualitative Results

To further demonstrate the superiority of our approach, we conduct a comprehensive qualitative comparison between our STEER and the state-of-the-art DiffusionDriveV2 across diverse and challenging urban driving scenarios.

As illustrated in Fig. A.2, in straight-driving conditions, DiffusionDriveV2 frequently exhibits lateral jitter and high-frequency wiggles, particularly when navigating through complex intersections. Conversely, our STEER consistently generates smooth, kinematically feasible, and strictly lane-centered trajectories. This advantage extends to complex turning maneuvers (Fig. A.3), where the baseline struggles with sharp kinks and noticeable centerline deviations. Our approach, however, produces highly curvature-aligned trajectories. The even distribution of our predicted waypoints further underscores the superior kinematic feasibility and passenger comfort of our method. Finally, we evaluate the models in slow-moving traffic and stopping scenarios (Fig. A.4). While DiffusionDriveV2 suffers from imprecise longitudinal control and uneven deceleration when approaching lead vehicles, our STEER yields significantly safer and smoother stopping profiles. The tightly clustered and uniformly distributed waypoints generated by our model reflect precise speed control, ensuring a safe following distance and enhanced ride comfort under congested stop-and-go conditions.

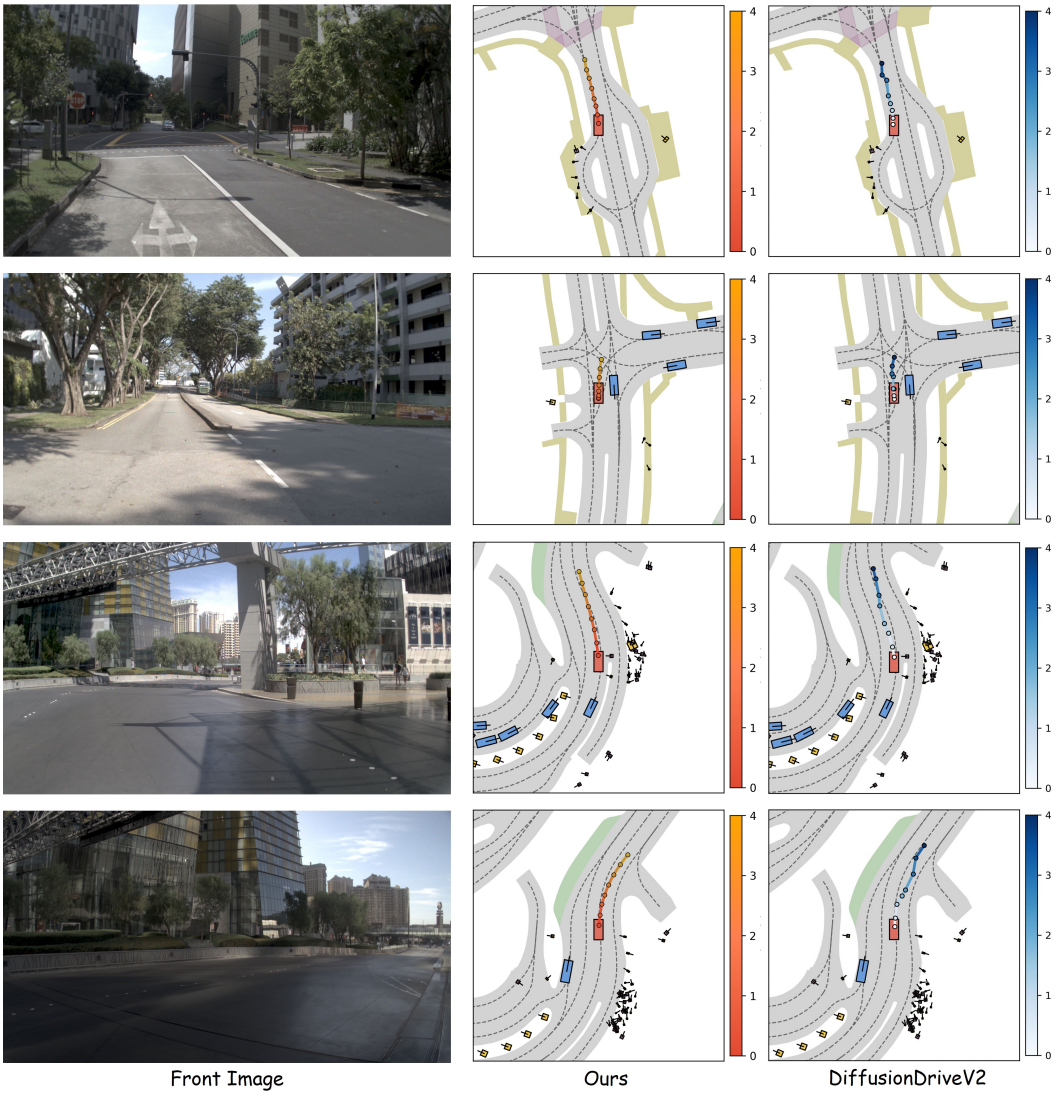



Figure A.2: Qualitative results in straight-driving scenarios. Compared to DiffusionDriveV2 (right, blue), which exhibits lateral jitter and high-frequency wiggles inside complex intersections, our STEER (middle, orange) generates significantly smoother, kinematically feasible, and strictly lane-centered straight trajectories. The color gradient represents the 4-second future horizon.




Figure A.3: Qualitative results in turning scenarios. Compared to DiffusionDriveV2, which struggles with sharp kinks and centerline deviations during complex maneuvers, our STEER generates significantly smoother and highly curvature-aligned trajectories. The even distribution of our waypoints further indicates superior kinematic feasibility and comfort.

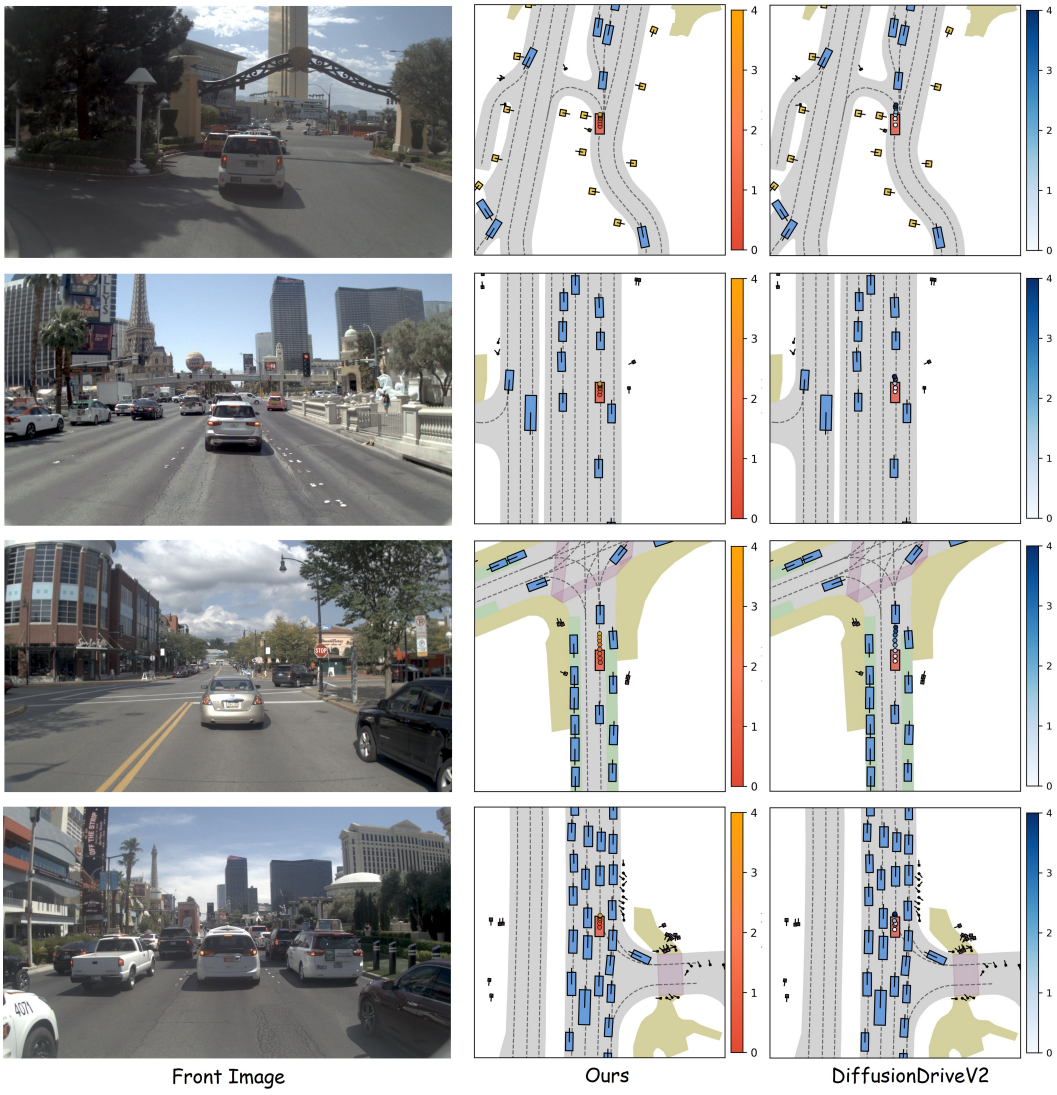



Figure A.4: Qualitative results in slow-moving traffic and stopping scenarios. Compared to DiffusionDriveV2, which struggles with precise longitudinal control and exhibits uneven deceleration when approaching lead vehicles, our STEER generates significantly smoother and safer stopping trajectories. The tightly clustered and evenly distributed waypoints further indicate superior speed control, ensuring a safe following distance and enhanced ride comfort in stop-and-go conditions.