INTERACTION-SYMMETRIC DECODING FOR CLOSED-LOOP TRAJECTORY PLANNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modeling multi-modal driving behaviors under dense agent interactions remains a fundamental challenge for autonomous closed-loop trajectory planning, and recent generative planners such as Diffusion Planner and Flow Planner have emerged as the dominant paradigm. However, these planners inherit a one-sided scene conditioning pattern from their transformer-based decoders, in which trajectory tokens query scene tokens through cross-attention while no complementary channel lets scene elements assign constraint responsibility back to the trajectory hypotheses. This misaligns with the joint multi-agent formulation they adopt, in which every agent's future should be conditioned by all others symmetrically. To address this inductive-bias mismatch, this paper introduces interaction-symmetric decoding, a design principle that recasts scene-trajectory coupling as a dual assignment problem, pairing forward demand assignment (what scene information each trajectory hypothesis retrieves) with reverse constraint-responsibility assignment (which trajectory hypotheses each scene element should influence). We instantiate it in SymDiff Planner, whose Interaction-Symmetric DiT decoder couples a Demand–Responsibility Assignment module that realizes the assignment dual with an Adaptive Symmetry Modulation module that regulates the reverse assignment strength by scene context. Extensive experiments on the large-scale nuPlan benchmark show that SymDiff Planner achieves state-of-the-art closed-loop performance among learning-based planners, with the most pronounced gains on the challenging Test14-hard split dominated by dense agent interactions and long-tail driving cases. Code and pre-trained checkpoints will be released upon acceptance.

1 INTRODUCTION

Autonomous driving critically hinges on the planning module, which converts perceptual understanding into kinematically feasible, safe, and comfortable motion commands (Chen et al., 2024a; Mozaffari et al., 2020). Among planning settings, closed-loop trajectory planning, which is continuously coupled with downstream control and reactive traffic, has emerged as the most demanding regime, where modeling defects compound into safety-critical failures. Driven by large-scale benchmarks such as nuPlan (Caesar et al., 2021), planning paradigms have rapidly evolved along rule-based, learning-based, and hybrid pipelines (Treiber et al., 2000; Hawke et al., 2020; Dauner et al., 2023; Cheng et al., 2024a), yet stable near-human closed-loop planning in complex open-world traffic remains an open challenge.

Rule-based planners offer strong interpretability but rely on handcrafted heuristics that limit adaptability in long-tail scenarios (Hallgarten et al., 2024). Learning-based planners scale by imitating human driving, yet struggle to capture its multi-modal nature and remain vulnerable to out-of-distribution closed-loop states (Codevilla et al., 2019; Cheng et al., 2024b), often falling back on rule-based post-processing (Dauner et al., 2023; Cheng et al., 2024a) that reintroduces the very limitations they seek to overcome. Diffusion models offer a compelling alternative by naturally characterizing multi-modal trajectory distributions (Yang et al., 2024; Zheng et al., 2025; Janner et al., 2022); Diffusion Planner (Zheng et al., 2025) and the concurrent Flow Planner (Tan et al., 2026), based respectively on a Diffusion Transformer (DiT) decoder and flow matching with spatiotemporal fusion, have rapidly established generative planners as a leading paradigm for closed-loop autonomous driving.

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.

Despite this progress, prevailing generative planners inherit a subtle yet consequential inductive-bias mismatch from their transformer-based decoders. The joint sample $\mathbf{X} = [X_{\text{ego}}, X_1, \ldots, X_M]$ they aim to generate is by construction symmetric across agents: every neighbor is a first-class predictive target whose future shapes the ego's feasible motion space. Yet standard decoders implement scene-trajectory coupling through a one-sided cross-attention pattern, in which noisy trajectory tokens actively query fixed scene-context tokens while no complementary channel lets scene elements assign constraint responsibility back to the evolving trajectory hypotheses. This missing scene-to-trajectory pathway manifests most severely in dense traffic, unprotected turns, and reactive multi-agent scenarios where closed-loop robustness is most demanded.

Motivated by this observation, we introduce interaction-symmetric decoding, a design principle that models scene-trajectory coupling as a dual assignment problem: demand assignment determines what scene information each trajectory hypothesis retrieves, and constraint-responsibility assignment determines which trajectory hypotheses each scene element should influence. Note that "symmetric" here refers to the treatment of these two assignment directions, not the update of the underlying token states. Scene tokens remain stable observations while trajectory tokens are progressively refined. We instantiate this principle in SymDiff Planner, whose core Interaction-Symmetric DiT (IS-DiT) decoder combines a Demand–Responsibility Assignment (DRA) module that realizes the dual at every refinement step with an Adaptive Symmetry Modulation (ASM) module that adaptively regulates the reverse assignment strength according to scene context.

In summary, our contributions are as follows:

- We provide the first systematic characterization of one-sided scene conditioning as an inductive-bias mismatch in existing generative closed-loop planners, at odds with the symmetric joint multi-agent formulation they adopt.
- We introduce interaction-symmetric decoding as the missing inductive bias, instantiated in the IS-DiT decoder via a DRA module that realizes the assignment dual and an ASM module that adaptively regulates the reverse assignment strength across scenes.
- Extensive experiments on nuPlan Val14, Test14-random, and Test14-hard under both NR-CLS and R-CLS show that SymDiff Planner achieves state-of-the-art closed-loop performance among learning-based planners, and a plug-in study further shows that DRA generalizes across regression-based, flow-matching, and diffusion backbones.

2 Related Work

2.1 Rule-Based Trajectory Planning

Rule-based planners encode driving expertise as handcrafted decision logic, offering strong interpretability that has long made them the de facto choice in industrial deployments (Chen et al., 2024b; Cao et al., 2022; Huang et al., 2024a). Representative works range from classical models such as IDM (Treiber et al., 2000), MOBIL (Kesting et al., 2007), and Apollo EM (Fan et al., 2018) to field-tested systems like Stanley (Thrun et al., 2006); more recently, PDM-Closed (Dauner et al., 2023) attained near-optimal closed-loop scores on nuPlan (Caesar et al., 2021). Nevertheless, their reliance on handcrafted heuristics fundamentally limits adaptability to long-tail interactions (Hallgarten et al., 2024), motivating the shift toward learning-based paradigms.

2.2 Learning-Based Trajectory Planning

Learning-based planning, dominated by the imitation-learning (IL) paradigm, exploits large-scale driving data (Caesar et al., 2021; Sun et al., 2023; Codevilla et al., 2019) to bypass the explicit priors of rule-based methods. Early CNN- (Kendall et al., 2019) and RNN-based (Bansal et al., 2018) efforts gave way to Transformer backbones for spatio-temporal modeling (Hu et al., 2023; Cheng et al., 2024b), followed by interaction-oriented variants such as GameFormer (Huang et al., 2023), DTPP (Huang et al., 2024b), PLUTO (Cheng et al., 2024a), BeTopNet (Liu et al., 2024), and CarPlanner (Zhang et al., 2025). Nevertheless, pure behavior-cloning planners face two fundamental limitations: MSE objectives cannot fit the multi-modal nature of human driving (Codevilla et al., 2019; Cheng et al., 2024b), and out-of-distribution closed-loop states often force fall-back

on rule-based post-processing (Dauner et al., 2023; Cheng et al., 2024a) that reintroduces the very limitations IL seeks to overcome.

2.3 Generative Trajectory Planning

Generative models, including diffusion (Janner et al., 2022; Chi et al., 2025) and flow-matching (Tan et al., 2026), map noise to structured trajectories via iterative refinement, a good fit for the multi-modal nature of human driving (Chung et al., 2023). Beyond the decision-making pioneers Diffuser (Janner et al., 2022) and Diffusion Policy (Chi et al., 2025), Diffusion-ES (Yang et al., 2024) and DiffusionDrive (Liao et al., 2025) apply diffusion to autonomous driving via guided sampling and anchor-based truncated diffusion, and ChainFlow-VLA (Wang et al., 2026) couples an autoregressive proposal generator with VLM-guided residual diffusion for end-to-end driving. Closest to ours are Diffusion Planner (Zheng et al., 2025) and Flow Planner (Tan et al., 2026): the former adopts a Diffusion Transformer (DiT) decoder for joint ego planning and neighbor prediction, while the latter uses flow-matching with fine-grained trajectory tokenization and classifier-free guidance. Yet both share the one-sided cross-attention design discussed in the Introduction, which misaligns with the symmetric joint formulation they adopt. Unlike generic bidirectional or co-attention designs that symmetrically update both modalities, the required remedy is asymmetric at the state-update level yet symmetric at the assignment level, since scene tokens are deterministic observations while trajectory tokens are the noisy hypotheses to be denoised. To the best of our knowledge, SymDiff Planner is the first to explicitly identify and address this interaction asymmetry in generative closed-loop planners on nuPlan.

3 Preliminary

This section formalizes the closed-loop trajectory-planning task and reviews the diffusion-based trajectory-generation formulation that SymDiff Planner builds upon, providing unified notation for the IS-DiT decoder introduced in the Methodology section.

3.1 Closed-Loop Trajectory Planning

Unlike open-loop planning or motion prediction, closed-loop planning requires the planner to regenerate a trajectory at every simulation step, convert it into actions through a downstream controller, and drive the simulation forward in a continuously coupled manner, which imposes stronger demands on multi-modal fitting, robustness to distribution shift, and interactive reasoning.

Following Diffusion Planner (Zheng et al., 2025), SymDiff Planner unifies ego planning and neighbor motion prediction into a single joint trajectory-generation task to capture ego-neighbor interaction. Let the planning horizon be T steps with state dimension $d_s = 4$, where each trajectory state $s^k_{ego} = (x_k, y_k, \cos\psi_k, \sin\psi_k)$ encodes position and heading in the ego-centric frame. Given a scene condition $\mathcal{C}$ aggregating the current states, historical trajectories, lane geometry, static obstacles, and navigation route, the planner jointly generates the future trajectories of $P = M + 1$ agents (the ego and its M nearest neighbors):

$$\mathbf{X} = [X_{\text{ego}}, X_1, \ldots, X_M] \in \mathbb{R}^{P\times T\times d_s}, \tag{1}$$

3.2 Diffusion Formulation for Trajectory Generation

Diffusion probabilistic models generate structured outputs by reversing a gradual noising process and impose no restrictive parametric assumption on the target distribution, making them naturally suited to the multi-modal behaviors inherent in driving. Given a clean joint trajectory $\mathbf{X} \sim p_{\text{data}}(\mathbf{X} \mid \mathcal{C})$, the forward process maps $\mathbf{X}$ to a noisy sample $\mathbf{Z}_t$ over continuous diffusion time $t \in [0, 1]$:

$$\begin{aligned} q_t(\mathbf{Z}_t \mid \mathbf{X}) &= \mathcal{N}(\mathbf{Z}_t; \alpha_t\mathbf{X}, \sigma_t^2\mathbf{I}), \\ \mathbf{Z}_t &= \alpha_t\mathbf{X} + \sigma_t\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \mathbf{I}), \end{aligned} \tag{2}$$

where α_t, σ_t ¿ 0 are predefined noise schedules chosen such that $\mathbf{Z}_t$ approaches standard Gaussian noise as $t\to1$. Conditioned on the scene $\mathcal{C}$, the reverse generation process is described by the

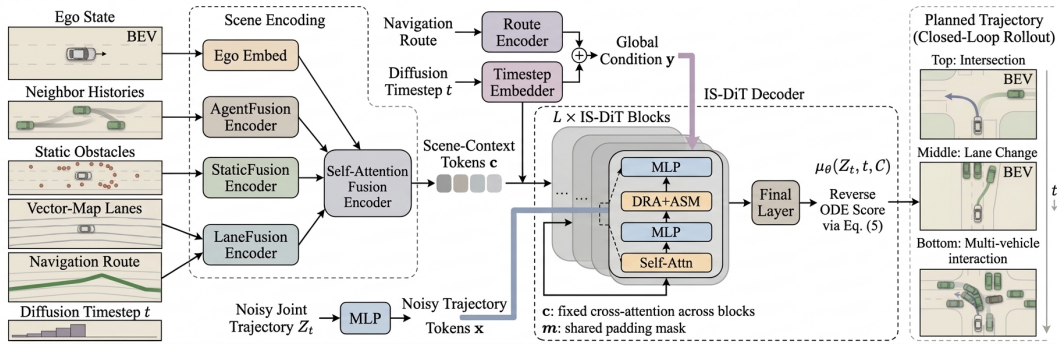


Figure 1: Overall pipeline of SymDiff Planner.

probability-flow ODE:

$$\frac{d\mathbf{Z}_t}{dt} = f(t)\mathbf{Z}_t - \frac{1}{2}g^2(t)\nabla_{\mathbf{Z}_t}\log q_t(\mathbf{Z}_t \mid \mathcal{C}), \tag{3}$$

where $f(t) = \frac{d\log\alpha_t}{dt}$ and $g^2(t) = \frac{d\sigma_t^2}{dt} - 2\sigma_t^2\frac{d\log\alpha_t}{dt}$ are determined by the noise schedule, and the only unknown term is the conditional score function $\nabla_{\mathbf{Z}_t}\log q_t(\mathbf{Z}_t \mid \mathcal{C})$.

For training efficiency, we adopt the clean-sample parameterization in which a network μ_θ directly predicts the clean trajectory $\mathbf{X}$ from the noisy sample, diffusion time, and scene condition:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{X},\mathcal{C},t,\boldsymbol{\epsilon}}\left[\|\mu_\theta(\mathbf{Z}_t, t, \mathcal{C}) - \mathbf{X}\|_2^2\right], \tag{4}$$

By Tweedie's formula, the score function required by the reverse ODE is recovered from the trained network as

$$\nabla_{\mathbf{Z}_t}\log q_t(\mathbf{Z}_t \mid \mathcal{C}) \approx \frac{\alpha_t\mu_\theta(\mathbf{Z}_t, t, \mathcal{C}) - \mathbf{Z}_t}{\sigma_t^2} \tag{5}$$

so that the trained denoising network implicitly provides the conditional score for trajectory generation, which is integrated with DPM-Solver++ (Lu et al., 2022) at inference.

3.3 DiT Decoder

SymDiff Planner instantiates μ_θ with a Diffusion Transformer (DiT) decoder (Peebles & Xie, 2023). The noisy trajectory is projected into D-dimensional trajectory tokens $\mathbf{x} \in \mathbb{R}^{B\times P\times D}$; the scene condition is encoded into scene-context tokens $\mathbf{c} \in \mathbb{R}^{B\times N\times D}$ and a global condition vector $\mathbf{y} \in \mathbb{R}^{B\times D}$ that fuses the route encoding with the diffusion-step embedding. Each DiT block applies four sequential operations, namely self-attention, MLP, cross-attention, and MLP, with $\mathbf{y}$ injected through adaptive Layer Normalization (adaLN) (Peebles & Xie, 2023). Trajectory tokens $\mathbf{x}$ are progressively updated across blocks, while (**c,y**) remain fixed throughout denoising.

Among these four operations, only the cross-attention step

$$\mathbf{x} \leftarrow \mathbf{x} + \mathrm{MHA}(\mathrm{LN}(\mathbf{x}), \mathbf{c}, \mathbf{c}), \tag{6}$$

mediates the interaction between trajectory and scene tokens. This step follows a one-sided assignment pattern in which trajectory tokens actively query fixed scene-context tokens, without any complementary channel that allows scene elements to assign constraint responsibility back to the evolving trajectory hypotheses. As identified in the Introduction, this one-sided pattern misaligns with the symmetric joint formulation of Eq. 1. SymDiff Planner retains the diffusion formulation in Eqs. (2)–(5) and replaces Eq. (6) with the proposed IS-DiT block, which realizes bidirectional interaction assignment and adaptively modulates its strength according to scene context.

4 Methodology

SymDiff Planner inherits the diffusion-based joint trajectory generation framework in the Preliminary and replaces the standard DiT decoder with the proposed Interaction-Symmetric DiT (IS-DiT) decoder, which decomposes scene-trajectory coupling into a forward Demand–Responsibility Assignment (DRA) module and an Adaptive Symmetry Modulation (ASM) module.

4.1 Overall Pipeline

The overall pipeline is illustrated in Figure 1. The four heterogeneous modalities in $\mathcal{C}$ (ego state, neighbor histories, static obstacles, and vector-map lanes) are encoded by four dedicated modules and aggregated by a self-attention Fusion Encoder into scene-context tokens $\mathbf{c}$. The navigation route and the diffusion timestep t are encoded and summed element-wise into the global condition $\mathbf{y}$. The noisy joint trajectory $\mathbf{Z}_t$ is projected by an MLP into trajectory tokens $\mathbf{x}$. The decoder stacks L IS-DiT blocks; only $\mathbf{x}$ propagates between blocks, while ($\mathbf{c}$, $\mathbf{y}$) remain fixed. The output of the last block is mapped back to the trajectory space by a Final Layer to produce $\mu_\theta(\mathbf{Z}_t, t, \mathcal{C})$, which is substituted into Eq. (5) to obtain the reverse-ODE score.

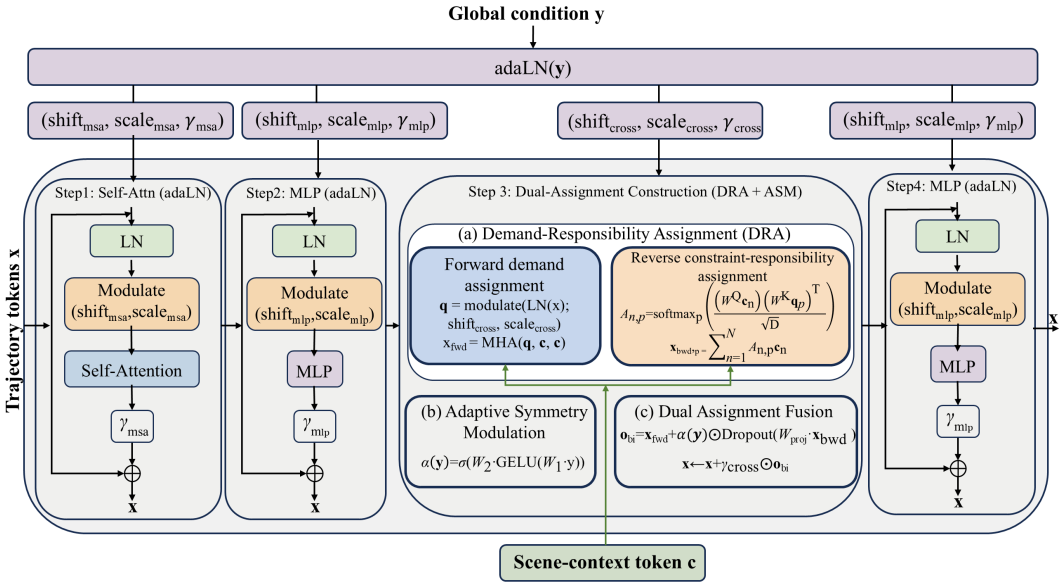


Figure 2: Internal structure of the IS-DiT block.

4.2 IS-DiT Block

The internal structure of an IS-DiT block is illustrated in Figure 2. Steps 1, 2, and 4 follow the standard DiT design (self-attention, MLP, and MLP with adaLN modulation), while Step 3 replaces the trajectory-query cross-attention in Eq. (6) with the proposed dual-assignment construction. Correspondingly, the adaLN modulation from $\mathbf{y}$ is extended with three parameters $(\text{shift}_{\text{cross}}, \text{scale}_{\text{cross}}, \gamma_{\text{cross}}) \in \mathbb{R}^{B\times D}$ for the bidirectional interaction assignment and its residual gate.

Demand–Responsibility Assignment (DRA). The cross-modulated query is first computed as

$$\mathbf{q} = \text{modulate}(\text{LN}(\mathbf{x}); \text{shift}_{\text{cross}}, \text{scale}_{\text{cross}}) \in \mathbb{R}^{B\times P\times D}, \tag{7}$$

where $(\mathbf{z}, \text{shift}, \text{scale}) = \mathbf{z} \odot (1 + \text{scale}) + \text{shift}$. The dual assignment then proceeds along two complementary branches that share $\mathbf{q}$ but implement different assignment directions.

Forward demand assignment. With $\mathbf{q}$ as the query and $\mathbf{c}$ as the key/value, the standard forward cross-attention aggregates scene features according to how much each trajectory hypothesis demands from the scene:

$$\mathbf{x}_{\text{fwd}} = \text{MHA}(\mathbf{q}, \mathbf{c}, \mathbf{c}) \in \mathbb{R}^{B\times P\times D}, \tag{8}$$

This branch preserves the trajectory-query aggregation pattern of the original DiT decoder.

Reverse constraint-responsibility assignment. To establish a scene-initiated alignment path, an attention map along the reverse query-key direction is explicitly computed. Let $W^{\text{Q}}, W^{\text{K}} \in \mathbb{R}^{D\times D}$ be projection matrices private to the reverse branch, not shared with the internal Q/K/V projections

of the forward branch. The reverse attention weights are defined as

$$A_{n,p} = \mathrm{softmax}_p \frac{(W^{\mathrm{Q}}\mathbf{c}_n)(W^{\mathrm{K}}\mathbf{q}_p)^{\mathrm{T}}}{\sqrt{D}} \in \mathbb{R}^{B\times N\times P}, \tag{9}$$

where the softmax is applied along the dimension indexed by p, such that $\sum_{p=1}^{\mathrm{P}} A_{n,p} = 1$ for every scene-context token $\mathbf{c}_n$. Intuitively, $A_{n,p}$ encodes the constraint responsibility of the n-th scene-context token with respect to the p-th trajectory token: it quantifies which trajectory hypotheses each scene element should influence. The same map is then used to aggregate scene features back onto the trajectory axis

$$\mathbf{x}_{\mathrm{bwd},p} = \sum_{n=1}^{N} A_{n,p}\mathbf{c}_n \in \mathbb{R}^{B\times D}, \quad p = 1, 2, \ldots, P, \tag{10}$$

yielding $\mathbf{X}_{\mathrm{bwd}} \in \mathbb{R}^{B\times P\times D}$. Every layer thus obtains an explicit scene-initiated pathway that lets scene elements cast constraint votes back onto trajectory tokens.

Adaptive Symmetry Modulation (ASM). Not all scenes call for equally strong multi-agent coupling. To let the strength of the reverse assignment vary with scene context, a $\mathbf{y}$-dependent, sample-conditional scalar gate is applied to the reverse branch only:

$$\alpha(\mathbf{y}) = \sigma(W_2 \cdot \mathrm{GELU}(W_1 \cdot \mathbf{y})), \quad \alpha \in \mathbb{R}^{B\times 1\times 1}, \tag{11}$$

where W_1, W_2 are learnable linear projections and $\sigma(\cdot)$ denotes the sigmoid activation. Since $\mathbf{y}$ combines the route-level condition with the diffusion-step embedding, $\alpha(\mathbf{y})$ adapts the openness of the reverse assignment path for each scene and each denoising stage.

Dual Assignment Fusion. The two branches are then fused with weight $\alpha(\mathbf{y})$ and written back to the trajectory features through a γ_{cross}-gated residual:

$$\mathbf{o}_{\mathrm{bi}} = \mathbf{x}_{\mathrm{fwd}} + \alpha(\mathbf{y}) \odot \mathrm{Dropout}(W_{\mathrm{proj}}\mathbf{x}_{\mathrm{bwd}}), \tag{12}$$

$$\mathbf{x} \leftarrow \mathbf{x} + \gamma_{\mathrm{cross}} \odot \mathbf{o}_{\mathrm{bi}}. \tag{13}$$

Here γ_{cross} controls the overall strength of the cross-attention residual, whereas $\alpha(\mathbf{y})$ controls the relative weight of the reverse branch within the fusion. Decoupling the two avoids the coarse-grained control that would result from regulating both with a single scalar. Together, Eqs. (7)–(13) replace the standard cross-attention step in Eq. (6) and constitute the third step of an IS-DiT block. To ensure that the newly introduced reverse path does not disrupt the pretrained backbone at initialization, we adopt an adaLN-Zero-style initialization that zero-initializes W_{proj}, the three new adaLN modulation parameters, and W_2 in Eq. (11) with b_2 = -1, so that the entire dual-assignment step writes nothing to $\mathbf{x}$ at the first training step.

5 Experiment

Extensive experiments are conducted on the large-scale nuPlan closed-loop planning benchmark to validate the effectiveness of SymDiff Planner. For brevity, the full experimental protocol is deferred to the Appendix A, comprising the dataset and benchmark splits (Appendix A.1.1), evaluation metrics (Appendix A.1.2), training and inference details (Appendix A.1.3 and A.1.4), and baseline configurations (Appendix A.2). In the main text, we first benchmark SymDiff Planner against representative rule-based, learning-based, hybrid, and generative planners, and then analyze the individual and joint contributions of DRA and ASM through structured ablation studies.

5.1 Main Results

As shown in Table 1, SymDiff Planner achieves state-of-the-art closed-loop performance among learning-based planners without rule-based refinement, attaining the best NR-CLS on all three splits and the best R-CLS on both Val14 and Test14-random. The gains are most pronounced on the challenging Test14-hard split, where SymDiff Planner surpasses the previous-best Diffusion Planner and Flow Planner by 2.05 and 1.57 NR-CLS points respectively, a regime dominated by dense agent interactions and long-tail cases in which decoder-level interaction reasoning is most critical. The

advantage further extends to reactive traffic, where SymDiff Planner outperforms both baselines on Val14 and Test14-random under R-CLS while remaining competitive on Test14-hard.

With an added rule-based post-processing module, SymDiff Planner w/ refine. attains the highest NR-CLS on both Val14 (94.53) and Test14-random (94.93) among all rule-based, hybrid, and generative baselines, while remaining competitive on the other metrics.

Table 1: Closed-loop performance of SymDiff Planner and representative baselines across the three nuPlan evaluation splits under both non-reactive (NR-CLS) and reactive (R-CLS) settings (best in bold).

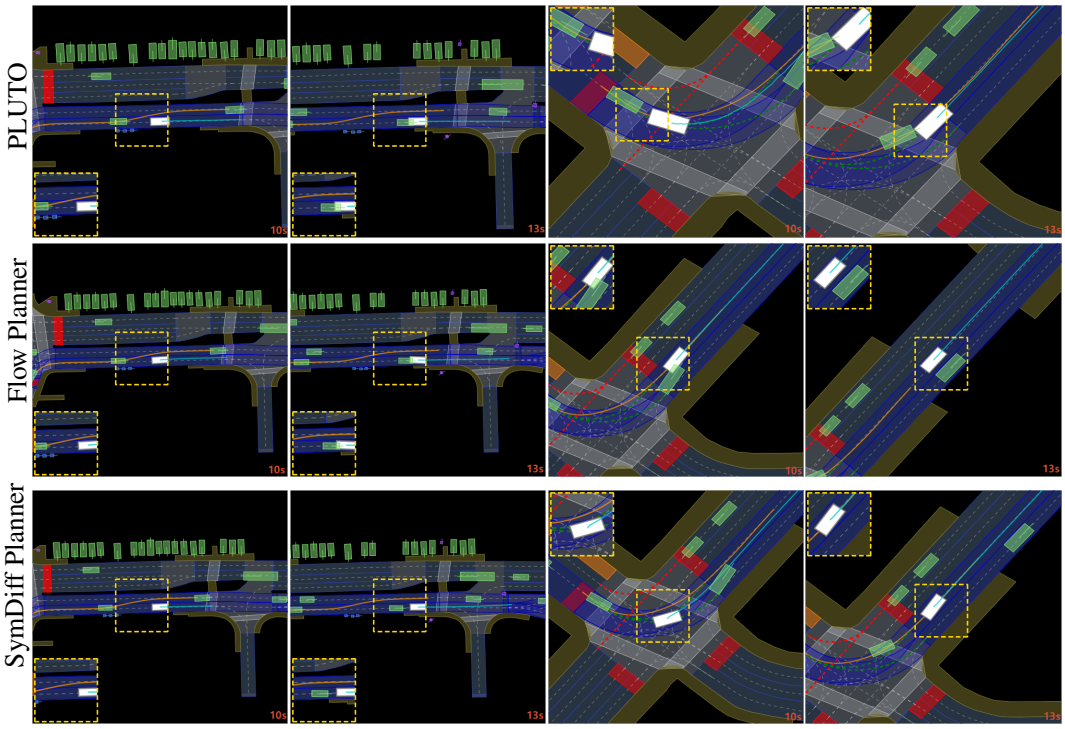
Type	Planner	Val14		Test14-hard		Test14-random	
		NR-CLS	R-CLS	NR-CLS	R-CLS	NR-CLS	R-CLS
Expert	Log-replay	93.53	80.32	85.96	68.80	94.03	75.86
Rule-based & Hybrid	IDM	75.60	77.33	56.15	62.26	70.39	74.42
	PDM-Closed	92.84	92.12	65.08	75.19	90.05	91.63
	PDM-Hybrid	92.77	92.11	65.99	76.07	90.10	91.28
	GameFormer	79.94	79.78	68.70	67.05	83.88	82.05
	PLUTO	92.88	76.88	80.08	76.88	92.23	90.29
	Diffusion Planner w/ refine.	94.26	92.90	78.87	82.00	94.80	91.75
	Flow Planner w/ refine.	94.31	92.38	78.64	80.25	94.79	92.40
	SymDiff Planner w/ refine. (ours)	94.53	92.54	78.65	81.23	94.93	92.32
Learning-based	PDM-Open*	53.53	54.24	33.51	35.83	52.81	57.23
	GameFormer w/o refine.	13.32	8.69	7.08	6.69	11.36	9.31
	PlanTF	84.27	76.95	69.70	61.61	85.62	79.58
	PLUTO w/o refine.*	88.89	78.11	70.03	59.74	89.90	78.62
	Diffusion Planner	89.87	82.80	75.99	69.22	89.19	82.93
	Flow Planner	90.43	83.31	76.47	**70.42**	89.88	82.93
	SymDiff Planner (ours)	**90.56**	**83.40**	**78.04**	69.97	**91.32**	**83.55**

Qualitative Case Study. Figure 3 shows bird's-eye-view (BEV) rollouts of PLUTO, Flow Planner, and SymDiff Planner on two representative interactive scenarios: (a) lane changing and (b) starting left turn (ego in white, neighbors in green). In (a), a fast follower closes in from behind during the lane change: both PLUTO and Flow Planner underestimate its closing speed and are struck from the rear, whereas SymDiff Planner keeps an adequate longitudinal buffer and completes the maneuver safely. In (b), at an unsignalized left turn with opposing traffic, Flow Planner turns prematurely and collides with the right-side vehicle, while PLUTO turns too conservatively and is rear-ended; only SymDiff Planner synchronizes its turning cadence with the surrounding traffic and completes the turn without collision. In this left-turn scenario, the ego-state profiles in Figure 4 further corroborate the timing effect: SymDiff Planner initiates the lateral offset and heading rotation in phase with the expert log-replay, whereas Flow Planner turns too early and overshoots the entry speed, while PLUTO delays both. Together, these results qualitatively support the quantitative gains in Table 1, indicating that interaction-symmetric decoding tightens ego-scene synchronization where multi-agent timing is decisive.

5.2 Ablation Studies

Three ablations trace the source of the gains: DRA is first plugged into three heterogeneous backbones to test whether it acts as a general-purpose interaction module (Table 2); DRA and ASM are then jointly ablated at the sub-metric level (Table 3) and probed by the ego's attention to each category of surrounding elements (Figure 5); finally, sensitivity to the cross-attention residual strength is analyzed (Table 4).

DRA as a Backbone-Agnostic Interaction Module. To test whether interaction-symmetric decoding is tied to our diffusion backbone or acts as a backbone-agnostic inductive bias, DRA is plugged into three planners from distinct families, PLUTO (regression-based), Flow Planner (flow matching), and Diffusion Planner (diffusion), without backbone-specific tuning, and evaluated on five interaction-centric Val14 scenarios. As shown in Table 2, DRA yields consistent average gains of +3.42, +1.95, and +3.68 respectively, peaking on the starting-right-turn scenario (+9.04 for Diffusion Planner, +3.47 for Flow Planner, +4.33 for PLUTO).



(a) Changing Lane (b) Starting Left Turn

Figure 3: Bird's-eye-view rollouts of closed-loop interaction behaviors. Qualitative comparison of PLUTO, Flow Planner, and SymDiff Planner in two representative interactive scenarios: (a) lane changing and (b) starting left turn. Each row corresponds to a planner and each column to a representative timestamp of the closed-loop rollout, showing the ego future planning, ego history, and neighboring-agent histories.

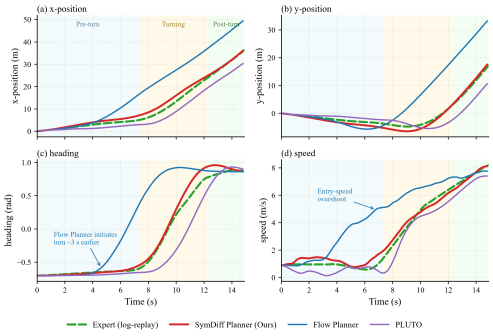


Figure 4: Ego-state evolution in a representative starting-left-turn scenario.

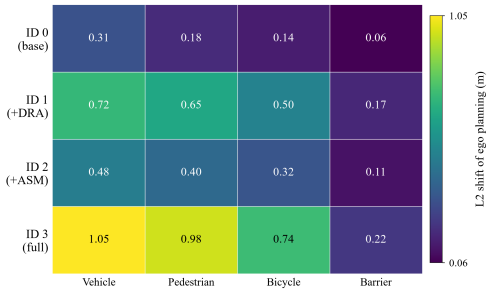


Figure 5: Ego attention to surrounding elements across ablation variants.

Complementarity of DRA and ASM. Table 3 reports the overall NR-CLS and five sub-metrics across the union of the three nuPlan splits for four variants (IDs 0–3: base, +DRA, +ASM, full IS-DiT). DRA alone (ID 0→1) gives the larger single-component gain (+1.42) and ASM alone (ID 0→2) a smaller one (+0.64), both already lifting Collision and Speed. Combining them (ID 3) raises the overall Score to 89.93, exceeding the sum of the individual gains, and attains the best on four of the five sub-metrics while remaining competitive on TTC. Qualitative comparisons on two Test14-random scenarios are given in Appendix A.4.

Table 2: Effectiveness of DRA as a plug-in interaction module across five interaction-centric Val14 scenarios. DRA is integrated into three representative planners spanning heterogeneous architectural families: PLUTO, Flow Planner, and Diffusion Planner. Scores are scenario-level closed-loop metrics; Avg. denotes the mean over the selected scenarios, and ΔAvg. denotes the improvement over the respective backbone without DRA.

Planner	DRA	Starting left turn	Starting right turn	Changing lane	Near multiple vehicles	Following lane with lead	Avg.	ΔAvg.
PLUTO	×	86.15	80.62	90.47	95.04	90.99	87.43	–
	✓	88.40	84.95	92.10	95.65	93.15	90.85	**+3.42**
Flow Planner	×	86.23	83.73	83.59	92.22	98.30	89.15	–
	✓	88.95	87.20	86.45	93.80	99.10	91.10	**+1.95**
Diffusion Planner	×	83.82	76.47	83.67	94.74	91.20	85.98	–
	✓	86.04	85.51	86.20	95.26	95.31	89.66	**+3.68**

Table 3: Sub-metric ablation of DRA and ASM aggregated across the three nuPlan evaluation splits.

ID	+DRA	+ASM	Score	Collision	Comfort	Similar	TTC	Speed
0	×	×	87.52	95.93	94.28	89.65	90.52	94.65
1	✓	×	88.94	96.47	94.81	90.81	90.70	97.78
2	×	✓	88.16	96.20	94.45	91.39	**91.23**	97.39
3	✓	✓	**89.93**	**96.60**	**95.08**	**92.07**	90.88	**97.87**

Table 4: Sensitivity to residual scaling factor w on Test14-random NR-CLS.

DRA weight	Score
1.0	**91.31**
0.8	90.62
0.6	88.77
0.4	88.54
0.2	87.36

Ego Attention to Surrounding Elements. To verify these gains stem from genuinely sharpened scene-trajectory coupling rather than incidental optimization, each variant is probed for its dependence on individual element categories. For each scene, it is run once with the original condition and once with all elements of a given category (Vehicle, Pedestrian, Bicycle, or Barrier) masked, and the L2 shift of the 8-second ego trajectory serves as a proxy for the ego's attention to that class. As shown in Figure 5, sensitivity grows across all four categories as DRA and ASM are progressively added, with DRA contributing most of the gain and ASM sharpening it further, and the full IS-DiT decoder attaining the largest sensitivity on the three reactive categories (Vehicle 1.05 m, Pedestrian 0.98 m, Bicycle 0.74 m). Even the static Barrier category shows a modest but consistent lift, indicating the sharpened attention extends beyond reactive agents to road-boundary constraints.

Sensitivity to the Cross-Attention Residual Strength. The residual gate γ_{cross} in Eq. (13) is scaled by a factor $w \in 0.2, 0.4, 0.6, 0.8, 1.0$ at inference to assess its effect on closed-loop performance. Since γ_{cross} modulates the whole cross-attention contribution (both branches after dual-assignment fusion), w jointly attenuates the scene-trajectory interaction written back at every decoder block. As shown in Table 4, the Test14-random NR-CLS decreases near-monotonically from 91.31 at $w = 1.0$ to 87.36 at $w = 0.2$; $w = 1.0$ is therefore used in all experiments.

6 Conclusion

We introduce SymDiff Planner, a novel learning-based framework for closed-loop trajectory planning that advances interaction modeling through interaction-symmetric decoding, which formalizes scene-trajectory coupling as a dual assignment problem over forward demand and reverse constraint-responsibility assignment. It is instantiated in the IS-DiT decoder, whose DRA module realizes the assignment dual and whose ASM module regulates the reverse assignment strength by scene context. Over the nuPlan benchmark, SymDiff Planner achieves state-of-the-art closed-loop performance among learning-based planners, and a plug-in study shows the proposed inductive bias generalizes across heterogeneous backbones. Due to space limit, more discussion on limitations and future direction can be found in Appendix A.5.

AI USE STATEMENT

Generative AI tools, including ChatGPT, were used solely to assist with English-language editing, LaTeX formatting, and citation-key conversion. They were not used to generate claims, design or conduct experiments, generate or process experimental data, or analyze and interpret the reported results. All AI-assisted edits and bibliographic entries were manually reviewed and verified by the authors against the original sources. The authors take full responsibility for the final content of this work, including all text, claims, results, and artifacts.

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.

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.

Yulong Cao, Wei ShangGuan, Baigen Cai, Liang Chai, and Weihua Qiu. Predictive trajectory planning for on-road autonomous vehicles based on a spatiotemporal risk field. *IEEE Intelligent Transportation Systems Magazine*, 15(1):400–420, 2022.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10164–10183, 2024a.

Yihan Chen, Hong Dong, and Zhen Li. Asynchronous spatial-temporal allocation for trajectory planning of heterogeneous multi-agent systems. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8504–8509. IEEE, 2024b.

Jie Cheng, Yingbing Chen, and Qifeng Chen. PLUTO: Pushing the limit of imitation learning-based planning for autonomous driving. *arXiv preprint arXiv:2404.14327*, 2024a.

Jie Cheng, Yingbing Chen, Xiaodong Mei, Bowen Yang, Bo Li, and Ming Liu. Rethinking imitation-based planners for autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14123–14130. IEEE, 2024b.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10–11):1684–1704, 2025.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2023.

Felipe Codevilla, Eder Santana, Antonio M. López, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9329–9338, 2019.

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.

Haoyang Fan, Fan Zhu, Changchun Liu, Liangliang Zhang, Li Zhuang, Dong Li, Wei Zhu, Jiangtao Hu, Hongye Li, and Qi Kong. Baidu Apollo EM motion planner. *arXiv preprint arXiv:1807.08048*, 2018.

Marcel Hallgarten, Jose Zapata, Moritz Stoll, Katrin Renz, and Andreas Zell. Can vehicle motion planning generalize to realistic long-tail scenarios? In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5388–5395. IEEE, 2024.

Jeffrey Hawke, Richard Shen, Ciprian Gurau, Siddharth Sharma, Daniele Reda, Nikolay Nikolov, Przemyslaw Mazur, Sean Micklethwaite, Nicholas Griffiths, Amar Shah, and Alex Kendall. Urban driving with conditional imitation learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 251–257. IEEE, 2020.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, and Lewei Lu. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17853–17862, 2023.

Huijing Huang, Yang Liu, Jin Liu, Qi Yang, Jianqiang Wang, David Abbink, and Arkady Zgonnikov. General optimal trajectory planning: Enabling autonomous vehicles with the principle of least action. *Engineering*, 33:63–76, 2024a.

Zhiyu Huang, Haochen Liu, and Chen Lv. GameFormer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3903–3913, 2023.

Zhiyu Huang, Peter Karkus, Boris Ivanovic, Yuxiao Chen, Marco Pavone, and Chen Lv. DTPP: Differentiable joint conditional prediction and cost evaluation for tree policy planning in autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6806–6812. IEEE, 2024b.

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

Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8248–8254. IEEE, 2019.

Arne Kesting, Martin Treiber, and Dirk Helbing. General lane-changing model MOBIL for car-following models. *Transportation Research Record*, 1999(1):86–94, 2007.

Bencheng Liao, Shaoyu Chen, Hongyang Yin, Bo Jiang, Cheng Wang, Shijie Yan, Xiangyu Zhang, Xiang Li, Ying Zhang, Qian Zhang, and Xinggang Wang. DiffusionDrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12037–12047, 2025.

Haichao Liu, Long Chen, Yu Qiao, Chen Lv, and Hongyang Li. Reasoning multi-agent behavioral topology for interactive autonomous driving. In *Advances in Neural Information Processing Systems*, volume 37, pp. 92605–92637, 2024.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022.

Saeid Mozaffari, Omar Y. Al-Jarrah, Mehrdad Dianati, Paul Jennings, and Alex Mouzakitis. Deep learning-based vehicle behavior prediction for autonomous driving applications: A review. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):33–47, 2020.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Qiao Sun, Shuo Zhang, Dan Ma, Jing Shi, Daiwei Li, Siqi Luo, Yu Wang, Nan Xu, Gang Cao, and Hang Zhao. Large trajectory models are scalable motion predictors and planners. *arXiv preprint arXiv:2310.19620*, 2023.

Tianyi Tan, Yinan Zheng, Ruiming Liang, Zexu Wang, Kexin Zheng, Jinliang Zheng, Jianxiong Li, Xianyuan Zhan, and Jingjing Liu. Flow matching-based autonomous driving planning with advanced interactive behavior modeling. In *Advances in Neural Information Processing Systems*, volume 38, pp. 38310–38335, 2026.

Sebastian Thrun, Michael Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, and Kenny Lau. Stanley: The robot that won the DARPA grand challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.

Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62(2):1805, 2000.

Xiyang Wang, Xinlin Wang, Tingguang Zhou, Gong Chen, Xingtai Gui, Zhi Xu, Xiaolei Wu, Feiyang Tan, Hangning Zhou, and Mu Yang. ChainFlow-VLA: Causal flow planning with vision-language models. *arXiv preprint arXiv:2605.23270*, 2026.

Brian Yang, Han Su, Nikolaos Gkanatsios, Tsung-Wei Ke, Ayush Jain, Jeff Schneider, and Katerina Fragkiadaki. Diffusion-ES: Gradient-free planning with diffusion for autonomous and instruction-guided driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15342–15353, 2024.

Di Zhang, Jialong Liang, Kai Guo, Shuang Lu, Qifan Wang, Rui Xiong, Zhen Miao, and Yue Wang. CarPlanner: Consistent auto-regressive trajectory planning for large-scale reinforcement learning in autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17239–17248, 2025.

Yinan Zheng, Ruiming Liang, Kexin Zheng, Jinliang Zheng, Liyuan Mao, Jianxiong Li, Wenhao Gu, Rui Ai, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Diffusion-based planning for autonomous driving with flexible guidance. In *International Conference on Learning Representations*, 2025.

A APPENDIX

A.1 EXPERIMENTAL SETUP

A.1.1 DATASET

We conduct experiments on the large-scale nuPlan benchmark (Caesar et al., 2021), which comprises approximately 1,300 hours of real-world urban driving logs across four cities and covers 75 predefined scenario types. Following the standard protocol of prior work (Zheng et al., 2025; Tan et al., 2026), we sample one million driving segments for training, each recorded at 10 Hz. For every training sample, we consider the lane and navigation information within a 100 m radius around the ego vehicle at the current frame, together with the two-second history trajectories of the nearest surrounding agents. Each input modality is padded to a fixed length, and attention masking is used to eliminate padded tokens during fusion.

Table 5: Hyperparameters of SymDiff Planner.

Type	Parameter	Symbol	Value
Scene input	Num. neighboring vehicles	–	32
	Num. past timestamps per agent	T_{hist}	21
	Dim. neighboring vehicles	D_{neighbor}	11
	Num. lanes	–	70
	Num. points per polyline	N_{pts}	20
	Dim. lane features	D_{lane}	12
	Num. navigation-route lanes	K	25
	Num. predicted neighboring vehicles	M	10
Model	Num. IS-DiT decoder blocks	L	3
	Hidden dimension	D	192
	Num. attention heads	h	6
	Dropout rate	–	0.1
	DRA scaling coefficient	w	1.0
Training	Optimizer	–	AdamW
	Batch size	–	2048
	Learning rate	–	5×10^{-4}
	Warm-up epochs	–	5
	Total epochs	–	500
	EMA decay	–	0.999
Inference	Noise coefficient	$\beta_{\min}, \beta_{\max}$	0.1, 20.0
	Temperature	–	0.5
	Temperature (w/ refine)	–	0.1
	Denoise step	–	10

We evaluate closed-loop planning performance on three official benchmark splits: Val14 (Dauner et al., 2023) containing 1,118 scenarios, Test14-random (Cheng et al., 2024b) containing over 200 randomly sampled scenarios, and Test14-hard (Cheng et al., 2024b) containing 272 worst-performing scenarios identified by the rule-based PDM planner (Dauner et al., 2023). Each split covers the same 14 scenario types specified by the nuPlan leaderboard, with Test14-hard concentrating on challenging cases such as dense traffic interactions and unprotected left turns, making it particularly suitable for evaluating closed-loop robustness in long-tail driving.

A.1.2 Evaluation Metrics

We adopt the official nuPlan closed-loop score as the primary metric. The score ranges from 0 to 100, with higher values indicating better planning quality. Depending on how neighboring agents are simulated, the closed-loop score is reported in two modes: non-reactive (NR-CLS), in which surrounding agents strictly replay their logged trajectories, and reactive (R-CLS), in which surrounding agents are controlled by the rule-based IDM (Treiber et al., 2000) and dynamically respond to ego actions. NR-CLS mainly reflects the intrinsic quality of the planned trajectory, while R-CLS additionally assesses closed-loop robustness under the train/test distribution shift induced by reactive traffic. We therefore report both NR-CLS and R-CLS on all three splits, yielding six evaluation metrics in total. For the sub-metric ablation in Table 3, we additionally report five closed-loop sub-metrics (Collision, Comfort, Similar, TTC, and Speed) defined by the nuPlan closed-loop protocol. Following the nuPlan convention, each sub-metric is reported as a compliance score in [0, 100] with higher values indicating better performance. Specifically, Collision denotes the fraction of scenarios without ego-at-fault collisions, TTC the fraction in which the time-to-collision remains within the safety bound, Comfort the fraction satisfying the jerk and acceleration thresholds, Speed the fraction of speed-limit compliance, and Similar the trajectory similarity between the planned ego trajectory and the expert log-replay.

A.1.3 Training Details

SymDiff Planner is trained on 8 NVIDIA A800 GPUs with a total batch size of 2048 for 500 epochs. We adopt the AdamW optimizer ($\beta_1 = 0.9, \beta_2 = 0.999$) with an initial learning rate of 5×10^{-4}, together with a cosine schedule and a 5-epoch linear warmup. Exponential moving average (EMA) with decay 0.999 is applied to model parameters to stabilize training. Across all ablation variants, we share the same optimization and inference protocol so that the reported differences reflect only architectural and initialization changes. Complete hyperparameter values, including scene-encoder inputs, model dimensions, and inference-time settings, are summarized in Table 5.

A.1.4 Inference Details

SymDiff Planner adopts a linear noise schedule with $(\beta_{\text{min}}, \beta_{\text{max}}) = (0.1, 20.0)$ and solves the reverse diffusion process using DPM-Solver++ in 10 denoising steps, striking a favorable balance between trajectory quality and sampling efficiency. Following prior practice, we apply low-temperature sampling with temperature 0.5 for the standalone SymDiff Planner and 0.1 for SymDiff Planner w/ refine, since a lower temperature yields more deterministic trajectories that better cooperate with downstream refinement. Under this configuration, SymDiff Planner produces an 8-second planning trajectory at 10 Hz on a single A800 GPU, which meets the real-time requirement for closed-loop deployment.

A.2 Baseline

Each baseline evaluated on nuPlan is briefly summarized below, along with its reproduction source.

· **IDM** (Treiber et al., 2000): A classical rule-based car-following method implemented in the official nuPlan devkit.

· **PDM** (Dauner et al., 2023): A strong planning framework from the nuPlan Planning Challenge, consisting of the rule-based PDM-Closed, the learning-based PDM-Open, and the hybrid PDM-Hybrid that combines learned prediction with rule-based trajectory selection.

· **GameFormer** (Huang et al., 2023): A transformer-based interactive prediction and planning method that models the interaction between the ego vehicle and surrounding agents through game-theoretic reasoning.

· **PlanTF** (Cheng et al., 2024b): A transformer-based imitation-learning planner that systematically studies architecture choices for closed-loop trajectory planning.

· **PLUTO** (Cheng et al., 2024a): A strong imitation-learning planner that improves closed-loop performance through contrastive representation learning, tailored data augmentation, and reference-line-based priors.

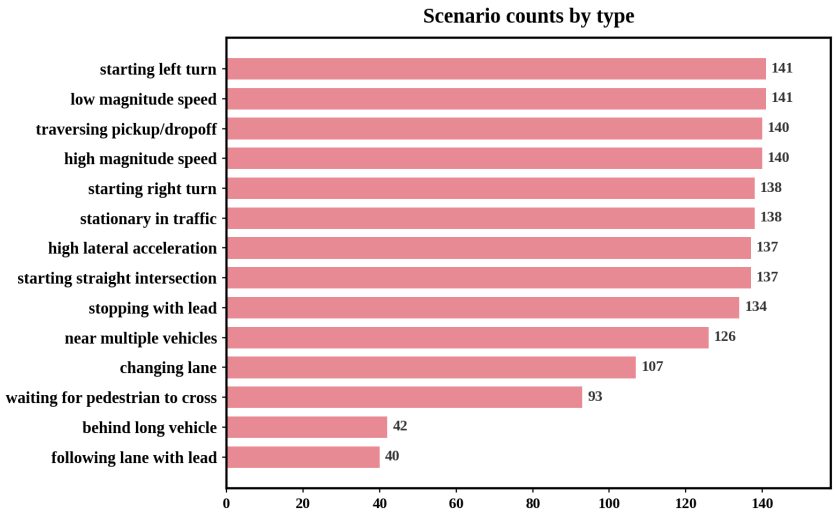


Figure 6: Per-category scenario counts in the sub-metric ablation pool. The pool aggregates all 1,654 scenarios from Val14, Test14-random, and Test14-hard, spanning the 14 scenario categories defined by the nuPlan leaderboard, with per-category counts shown on the right.

· **Diffusion Planner** (Zheng et al., 2025): A diffusion-based planner that formulates ego planning and neighbor motion prediction as a joint trajectory-generation task and adopts a Diffusion Transformer decoder for multi-modal planning.

· **Flow Planner** (Tan et al., 2026): A concurrent flow-matching-based planner that introduces fine-grained trajectory tokenization, interaction-enhanced spatiotemporal fusion, and classifier-free guidance for interactive behavior generation.

For IDM, the built-in implementation in the official nuPlan devkit is used. Checkpoints for PDM (PDM-Closed, PDM-Open, PDM-Hybrid) are sourced from the released PDM codebase. PlanTF and PLUTO are evaluated with the official checkpoints released by the authors, with PLUTO w/o refine. obtained by skipping the post-processing pipeline; GameFormer is trained from the official codebase, and GameFormer w/o refine. is further reported by skipping the refinement step. Diffusion Planner and Flow Planner are reproduced with their respective official implementations. SymDiff Planner w/ refine. adopts the same rule-based trajectory-scoring refinement module used by Diffusion Planner (Zheng et al., 2025) and Flow Planner (Tan et al., 2026).

A.3 Evaluation Coverage for the Sub-Metric Ablation

Unlike the aggregate closed-loop scores reported in Table 1, which are stratified per benchmark split, the sub-metric ablation in Table 3 requires a sufficiently large and diverse scenario pool to yield stable estimates of individual sub-metrics such as Collision, Comfort, TTC, and Speed. To this end, Table 3 is computed over the union of all three official nuPlan evaluation splits, namely Val14, Test14-random, and Test14-hard, comprising 1,654 scenarios in total and spanning all 14 scenario categories defined by the nuPlan leaderboard.

The per-category scenario counts of this pool are illustrated in Figure 6. Interaction-heavy categories such as starting left turn (141), starting right turn (138), near multiple vehicles (126), changing lane (107), and waiting for pedestrian to cross (93) contribute a substantial fraction of the pool, ensuring that the sub-metric contrasts reported in Table 3 are dominated by scenarios where scene-trajectory interaction is decisive. Routine-driving categories such as low magnitude speed (141), high magnitude speed (140), behind long vehicle (42), and following lane with lead (40) are also retained to preserve balanced coverage of sub-metrics that require sufficient non-interactive samples to remain meaningful.

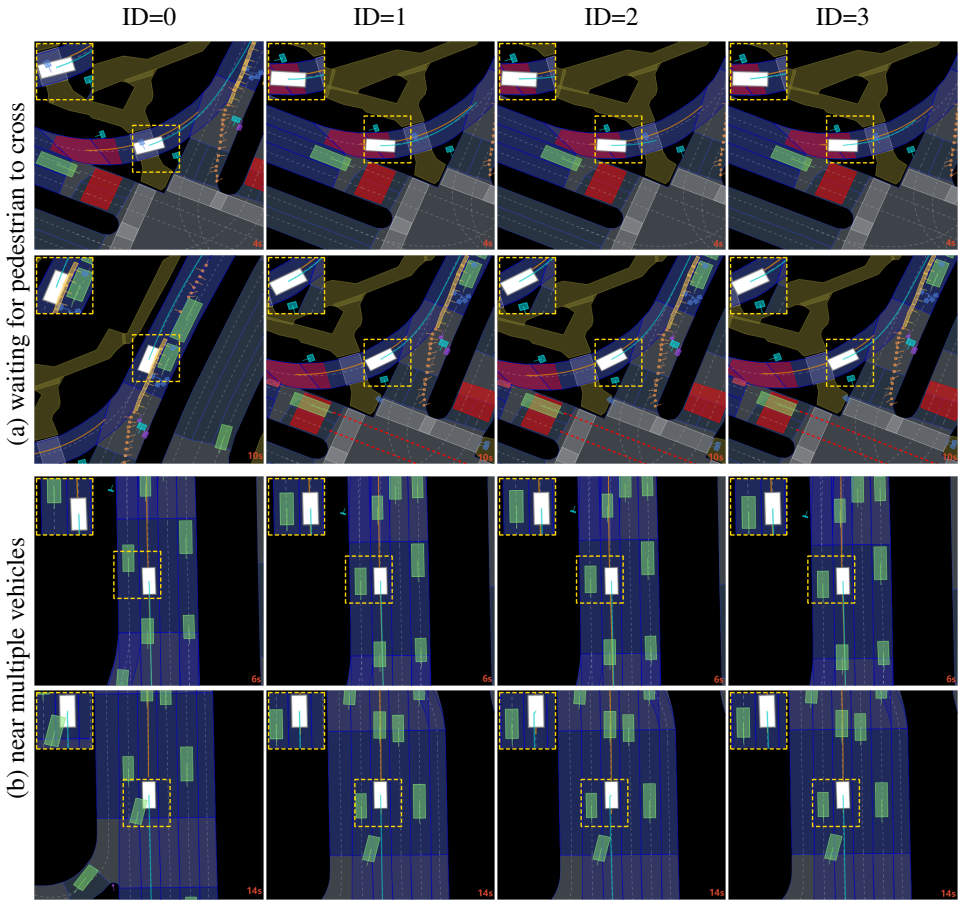


Figure 7: Qualitative ablation of DRA and ASM on Test14-random. Closed-loop planning behaviors of the four ablation variants IDs 0–3 as defined in Table 3 in two representative interactive scenarios: (a) waiting for pedestrian to cross and (b) near multiple vehicles. Each scenario is visualized at two representative timestamps, resulting in four rows in total; each column corresponds to an ablation ID. The ego vehicle is shown in white and the neighboring agents in green.

A.4 Qualitative Ablation of DRA and ASM

Figure 7 visualizes the closed-loop planning behaviors of the four IS-DiT ablation variants (IDs 0–3 as defined in Table 3) on two representative Test14-random scenarios, with each scenario captured at two representative timestamps. In every subfigure, the ego vehicle is shown in white and the neighboring agents in green. In scenario (a) waiting for pedestrian to cross, captured at 4 s and 10 s, the ego approaches an intersection where a pedestrian is crossing its planned path. The base decoder (ID 0) fails to sufficiently slow the ego in response to the anticipated crossing and drifts into the pedestrian's trajectory, resulting in a collision at 4 s. The two partial variants that add DRA or ASM in isolation (IDs 1 and 2) both successfully avoid the collision, yet exhibit visibly less stable stopping profiles across the two timestamps: the ego approaches the crosswalk with narrower safety margins and, in some cases, requires abrupt lateral adjustment before reaching a stop. Only the full IS-DiT decoder (ID 3) maintains a smooth deceleration profile with a stable stop position that respects the pedestrian right-of-way throughout the rollout. In scenario (b) near multiple vehicles, captured at 6 s and 14 s, the ego navigates a densely populated multi-lane setting sur-rounded by multiple neighboring agents. At 6 s, all four variants place the ego safely within its lane with comparable longitu-dinal positions, since the surrounding interactions have not yet fully developed. By 14 s, however, the failure modes of the partial variants become apparent. The base decoder (ID 0) fails to account for a neighboring vehicle closing in from the ad-jacent left lane and drifts into direct contact with it, resulting in a collision. The two partial variants (IDs 1 and 2) both avoid the collision but

exhibit narrower lateral safety margins, with the ego remaining closer to the surrounding vehicles than desir-able despite the addition of DRA or ASM in isolation. Only the full IS-DiT decoder (ID 3) preserves both lane alignment and balanced lateral spacing from all neighboring agents throughout the rollout.

A.5 Limitation & Discussion & Future Work

Here, we discuss the limitations of SymDiff Planner, potential solutions, and interesting future work. First, SymDiff Planner is primarily evaluated on the nuPlan benchmark, and its transferability to broader geographic regions, novel vehicle dynamics, and perception noise beyond the log-replay assumption remains an open question; fine-tuning on domain-specific driving logs or interactive long-tail benchmarks would be a natural next step. Second, ASM relies on the route-level condition and the diffusion-step embedding as a compact proxy for scene-adaptive regulation, so the semantics of the learned gate $\alpha_{\mathbf{y}}$ remains largely implicit; more interpretable scene-aware gates conditioned on explicit complexity measures such as traffic density and conflict risk would enable the reverse-assignment strength to be diagnosed and overridden by upstream reasoning modules. Third, SymDiff Planner relies purely on data-driven learning to acquire safety-critical behaviors and does not natively enforce hard safety, comfort, or feasibility constraints during denoising, which could be addressed by combining the IS-DiT decoder with classifier-guidance mechanisms or feasibility-guided extensions. Overall, although certain limitations exist, extensive experiments demonstrate that interaction-symmetric decoding can meaningfully advance closed-loop trajectory planning, and we hope this work encourages further exploration of scene-trajectory interaction as an underexplored yet high-leverage design axis for future generative planners.