Temporally Decoupled Diffusion Planning for Autonomous Driving

Xiang Li[1], **Bikun Wang**[1*], **John Zhang**[1], **Jianjun Wang**[1]

[1]Bosch Cross-Domain Computing Solutions
wodlxwhy@gmail.com,{Bikun.WANG, external.John.ZHANG, Jianjun.WANG2}@cn.bosch.com

Abstract

Motion planning in dynamic urban environments requires balancing immediate safety with long-term goals. While diffusion models effectively capture multi-modal decision-making, existing approaches treat trajectories as monolithic entities, overlooking heterogeneous temporal dependencies where near-term plans are constrained by instantaneous dynamics and far-term plans by navigational goals. To address this, we propose Temporally Decoupled Diffusion Model (TDDM), which reformulates trajectory generation via a noise-as-mask paradigm. By partitioning trajectories into segments with independent noise levels, we implicitly treat high noise as information voids and weak noise as contextual cues. This compels the model to reconstruct corrupted near-term states by leveraging internal correlations with better-preserved temporal contexts. Architecturally, we introduce a Temporally Decoupled Adaptive Layer Normalization (TD-AdaLN) to inject segment-specific timesteps. During inference, our Asymmetric Temporal Classifier-Free Guidance utilizes weakly noised far-term priors to guide immediate path generation. Evaluations on the nuPlan benchmark show TDDM approaches or exceeds state-of-the-art baselines, particularly excelling in the challenging Test14-hard subset.

Code — https://github.com/wodlx/TDDM

Introduction

Motion planning is one of the core technical challenges in autonomous driving (Chen et al. 2024a; Aradi 2022), with the objective of generating safe, comfortable, and human-like trajectories in dynamically changing and uncertain environments (Chen et al. 2024b). Traditional planning methods, such as search-based (Stilman and Kuffner 2008) or optimization-based approaches (Fan et al. 2018), rely on precise environmental models and complex heuristic rules (Treiber, Hennecke, and Helbing 2000; Dauner et al. 2023). Although they can be highly efficient and perform well in scenarios covered by these rules, they often struggle to cope with the highly dynamic and interactive nature of complex urban scenarios (Leonard et al. 2009).

In recent years, learning-based planning has emerged as the mainstream paradigm. By learning driving policies from large-scale data, these methods can better generalize to complex and variable real-world scenarios, reducing the reliance on hand-crafted rules. Among learning-based approaches, imitation learning (IL) (Pomerleau 1988; Bansal, Krizhevsky, and Ogale 2018), especially end-to-end (E2E) methods (Hu et al. 2023; Cheng, Chen, and Chen 2024), has achieved significant success. By directly learning the mapping from perception to control, it greatly simplifies the complexity of the traditional planning framework. However, classic imitation learning methods face two core challenges. The first is *distribution shift* (Cheng et al. 2024), where the state distribution encountered during testing differs from the training data, leading to compounding errors. The second is the difficulty in capturing the inherent *multi-modal decision-making* (Cui et al. 2019; Chen et al. 2024c) characteristic of real-world driving scenarios. Faced with the same situation, a human driver might make several equally valid and safe

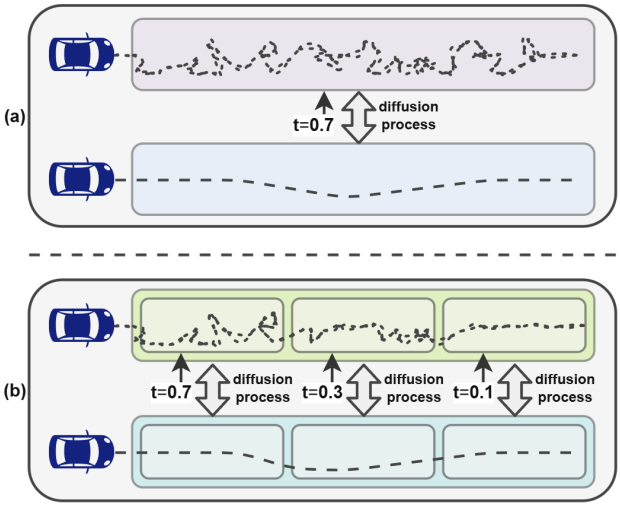

Figure 1: **The comparison of our temporal decoupled diffusion model and full sequence diffusion model.** (a) full sequence diffusion model. (b) temporal decoupled diffusion model (Ours). It can be seen that the biggest difference between our model and the full sequence model is the support for independent diffusion processes on temporal segments, where t denotes the diffusion timestep.

*Corresponding author.

decisions, whereas a single regression target often yields overly conservative or averaged-out policies.

To resolve this, recent literature shifts from deterministic regression to generative sequence modeling. To align with the broader automated planning community, it is crucial to distinguish these paradigms: classical planning searches for an optimal action sequence based on explicit logical constraints, whereas trajectory generation reframes the task as conditional probability modeling. Instead of explicit logical search, generative models learn to sample diverse, contextually plausible future trajectories from a learned data distribution.

Driven by this generative formulation, diffusion models (Ho, Jain, and Abbeel 2020; Song et al. 2021) have been introduced into the field of trajectory planning, owing to their powerful ability to model complex data distributions and their flexibility in being guided by different conditions to achieve various goals. By redefining the planning task as a conditional trajectory generation problem, diffusion models can start from random noise and, through a denoising process guided by scene context and kinematic constraints, generate a diverse set of plausible trajectories that are consistent with the scene. The success of applying diffusion models to trajectory generation has been demonstrated in several works (Jiang et al. 2023b; Liao et al. 2025; Zheng et al. 2025). However, existing diffusion-based planning methods often treat the entire trajectory as a monolithic, indivisible entity (Zheng et al. 2025). This ignores the varying dependencies of different temporal segments of the trajectory in the scene context. For instance, the initial segment of a trajectory (near-term plan) is more dependent on the vehicle's precise current state and the instantaneous dynamics of the surrounding environment, whereas the final segment (far-term plan) is more focused on achieving long-term navigational goals.

Inspired by recent advances in sequential modeling from the video generation domain, we draw an analogy between trajectory generation and the generation of video frame sequences. Cutting-edge methods in video generation, such as Diffusion Forcing (Chen et al. 2025), achieve higher quality and longer-term video extrapolation by applying independent noise to different frames and modeling their temporal relationships. Similarly, works like CausVid (Yin et al. 2025) have further explored efficient causal structures to improve generation speed and interactivity. These works collectively highlight the importance of decoupling sequential units and meticulously modeling their internal correlations in sequence generation tasks.

To this end, we propose a novel method named the **Temporally Decoupled Diffusion Model (TDDM)**, which aims to model the internal temporal dependencies within a trajectory more finely, thereby generating plans that are more consistent and forward-looking. Our main contributions can be summarized as follows:

- We introduce a novel temporally-decoupled training paradigm. We partition the trajectory into multiple temporal segments and apply different levels of noise independently to each segment during training. This randomized temporal masking mechanism significantly enhances the model's ability to capture complex temporal correlations.
- We design an asymmetric temporal guidance strategy for inference. By extending Classifier-Free Guidance (Dhariwal and Nichol 2021) to the temporal dimension, our strategy uses weakly noised far-term priors as a condition to guide the generation of a near-term plan. This effectively frames the inference process as a goal-directed trajectory completion task, enhancing the long-term consistency of the plan.
- We develop an efficient model architecture that supports temporal decoupling. Our architecture, based on the diffusion transformer (Peebles and Xie 2023) framework, incorporates a Temporally Decoupled Adaptive Layer Normalization (TD-AdaLN) module. This allows for the independent injection of diffusion timestep information for each temporal segment, providing architectural support for our proposed training and inference procedures.
- Comprehensive evaluations on the large-scale nuPlan (Caesar et al. 2022) benchmark robustly demonstrate the superiority of our proposed model. The results show that TDDM achieves performance that approaches or exceeds that of state-of-the-art learning-based methods, particularly in the most challenging long-tail scenarios, where it exhibits exceptional robustness and planning consistency.

Related Work

Imitation-based Planning. Imitation learning, which learns a policy by mimicking expert demonstrations, is a cornerstone of autonomous driving research. Behavioral Cloning (BC) has been particularly prevalent in end-to-end architectures that map sensor inputs directly to control actions. The architectural evolution of these systems is well-documented: early approaches (Pomerleau 1988) employed simple ***Convolutional Neural Networks*** (CNN). To capture the temporal nature of driving, subsequent works such as ChauffeurNet (Bansal, Krizhevsky, and Ogale 2018) incorporated ***Recurrent Neural Networks*** (RNN). More recently, the ***Transformer*** architecture has become dominant, with models like planTF (Cheng et al. 2024) leveraging its capacity for long-horizon sequence modeling to imitate expert trajectories. To surpass the limitations of vanilla imitation learning, hierarchical approaches like PLUTO (Cheng, Chen, and Chen 2024), which decouples lateral and longitudinal control. A prominent trend is the development of unified end-to-end frameworks (Weng et al. 2024; Jiang et al. 2023a; Hu et al. 2023), exemplified by UniAD (Hu et al. 2023), which integrates perception, prediction, and planning into a single model. However, a persistent challenge for these methods is effectively capturing the multi-modal nature of driving decisions. While some methods, such as VADv2 (Chen et al. 2024c), address this by discretizing the action space into a large vocabulary, this challenge has broadly motivated the exploration of more powerful generative paradigms, notably diffusion models (Ho, Jain, and Abbeel 2020; Song et al. 2021).

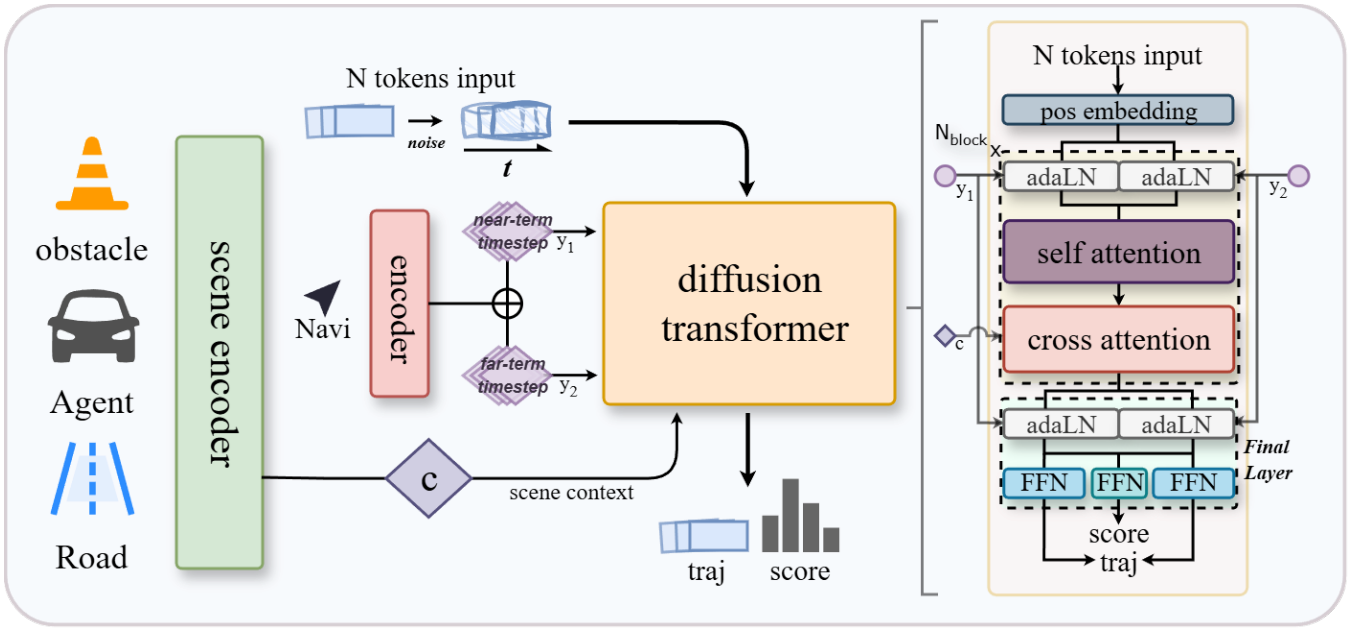

Figure 2: **Overview of the temporally decoupled diffusion model.** We adopt the diffusion transformer (DiT) architecture as the decoder, with the encoder responsible for encoding environmental context information (including static obstacles, agents, and roads). The decoder can accept independent timesteps from temporal segments through temporally decoupled adaLN.

Diffusion Models for Trajectory Planning. Diffusion models reframe motion planning as a conditional generation task. They iteratively refine a random noise tensor into one or more feasible trajectories, conditioned on scene context such as sensor data and navigational goals. This paradigm inherently supports the generation of diverse, high-quality outputs and has shown remarkable performance. MotionDiffuser (Jiang et al. 2023b) was among the first to apply this concept to multi-agent motion forecasting. Hybrid approaches like Diffusion-ES (Yang et al. 2024) combine diffusion models with evolutionary strategies, using the former to generate a high-quality initial policy population for the latter to optimize, enhancing robustness in uncertain scenarios. To improve efficiency, DiffusionDrive (Liao et al. 2025) initiates the denoising process from learned anchors, accelerating inference and improving planning accuracy. Meanwhile, DiffusionPlanner (Zheng et al. 2025) casts planning as a joint prediction problem for all agents and enables style-conditioned trajectory generation. Despite their success, these methods share a fundamental modeling assumption: they treat the entire future trajectory as a monolithic, indivisible entity. This monolithic denoising approach overlooks the heterogeneous nature of temporal dependencies within a trajectory. For instance, near-term waypoints are heavily constrained by immediate collision avoidance, whereas long-term waypoints are dictated by global route adherence. This simplification restricts the model's ability to generate trajectories that optimally balance short-term safety with long-term consistency.

Decoupled Modeling in Sequence Generation. Inspiration for addressing this limitation is drawn from recent advances in other sequence generation domains, particularly video synthesis. There, researchers have highlighted the benefits of decoupling sequential units. For example, Diffusion Forcing (Chen et al. 2025) applies noise independently to video frames, forcing the model to learn fine-grained causal relationships. This principle is extended in works like CausVid (Yin et al. 2025) and DFoT (Song et al. 2025), which further investigate causal architectures and guided generation on independently noised units. These studies collectively demonstrate that by decomposing a sequence and explicitly modeling inter-unit dependencies, it is possible to significantly improve generation quality and temporal coherence. Motivated by this principle, we introduce the concept of ***temporal decoupling*** to trajectory planning. We posit that a trajectory should be modeled not as a monolithic whole, but as a sequence of interconnected temporal segments. By subjecting these segments to independent yet coordinated diffusion processes, our model learns a more expressive representation of temporal dependencies. This allows for a more nuanced control over the planning process, enabling our model to reconcile short-term reactivity with long-term goal consistency—a critical capability largely unaddressed by prior monolithic diffusion-based planners.

Preliminary

Conditional Diffusion Model

A conditional diffusion model learns to reverse a gradual noising process to generate data samples conditioned on some context c. The forward process is a fixed Markov chain that incrementally adds Gaussian noise to the original data τ^0 over a series of timesteps. The data distribution at any

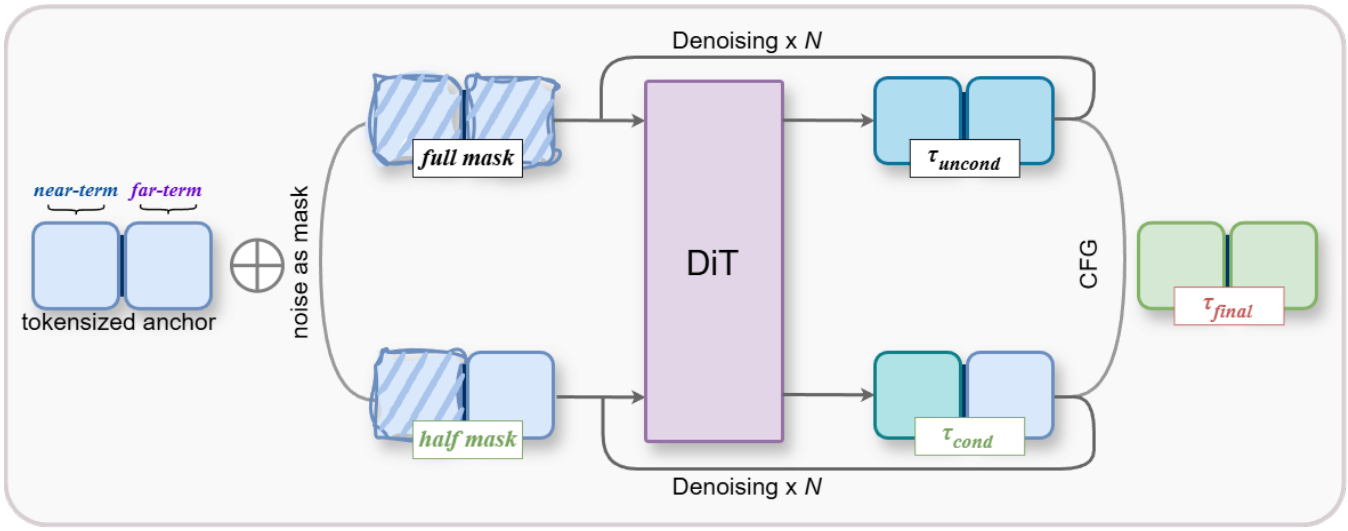

Figure 3: **Pipeline of the Asymmetric Temporal CFG**. The trajectory is a fusion of outputs from an unconditional path and a conditional path. The unconditional path performs standard full-sequence diffusion. The conditional path, enabled by an asymmetric temporal mask, leverages a nearly clean future prior to guide the denoising generation of the past segment.

timestep i can be expressed in a closed form:

$$q(\tau^i \mid \tau^0) = \mathcal{N}\left(\tau^i; \sqrt{\bar{\alpha}^i}\,\tau^0, (1-\bar{\alpha}^i)\mathbf{I}\right), \tag{1}$$

where τ^i is the noised data at timestep $i \in [0,1]$, and $\{\bar{\alpha}^i\}$ is a predefined noise schedule.

The reverse process involves training a neural network $f_\theta(\tau^i, c, i)$, typically parameterized to predict the added noise, to denoise τ^i back towards τ^0 given the condition c. During inference, starting from pure Gaussian noise τ^T, the model iteratively applies the learned denoising function to generate a clean data sample τ^0. This iterative process can be represented as:

$$p_\theta(\tau^0 \mid c) = \int p(\tau^T) \prod_{t=1}^{T} p_\theta(\tau^{t-1} \mid \tau^t, c) d\tau^{1:T}. \tag{2}$$

For practical inference, this continuous-time formulation is approximated by discretizing the time interval into a finite number of steps T, where the integral is resolved through an iterative sampling procedure.

Anchor-based Trajectory Vocabulary

To structure the motion planning problem for a generative model, we discretize the continuous action space into a predefined trajectory vocabulary (Liao et al. 2025), $V = \{v_i\}_{i=1}^{M}$. This approach transforms the problem from regressing a continuous path to selecting and refining the best trajectory prototype from a diverse set.

This vocabulary is created by applying the k-means clustering algorithm to a large-scale dataset of expert-driven trajectories, such as nuPlan (Caesar et al. 2022). Each resulting cluster centroid $v_i \in V$ serves as a trajectory anchor, representing a distinct, kinematically feasible driving maneuver. Each anchor is a prototype trajectory τ composed of a sequence of T_h waypoints, where $\tau = \{(x_t, y_t, \phi_t)\}_{t=1}^{T_h}$, capturing the vehicle's position and heading over the planning horizon.

Problem Formulation

Building on the concepts above, we formulate motion planning as a conditional generation task. The core idea is to learn a denoising model that can refine a set of noisy trajectory anchors into a final multimodal trajectory distribution, conditioned on the scene context c.

Specifically, we first tokenize each trajectory anchor $v \in V$ by segmenting it into N temporal segments: $\{\tau_1^0, \tau_2^0, \ldots, \tau_N^0\}$. During training, we apply independent noise to each segment. For the n-th segment τ_n^0, we sample an independent diffusion timestep $i_n \in [0,1]$ and a standard Gaussian noise $\epsilon_n \sim \mathcal{N}(0, \mathbf{I})$. The resulting noised segment $\tau_n^{i_n}$ is calculated as:

$$\tau_n^{i_n} = \sqrt{\bar{\alpha}^{i_n}}\tau_n^0 + \sqrt{1-\bar{\alpha}^{i_n}}\epsilon_n. \tag{3}$$

This independent noising forces the model to learn complex temporal dependencies between segments, rather than relying on uniform noise patterns.

The overall objective is to learn the weights θ of a denoising network f_θ that takes the scene context c and the set of noised anchors as input, and outputs a refined set of trajectories $\{\hat{\tau}_k\}$ along with their corresponding confidence scores $\{\hat{s}_k\}$:

$$\{\hat{s}_k, \hat{\tau}_k\}_{k=1}^{M} = f_\theta(\{\{\tau_{k,n}^{i_n}\}_{n=1}^{N}\}_{k=1}^{M}, c). \tag{4}$$

Essentially, the model learns to jointly denoise all anchors and predict which one represents the optimal plan for the given context.

Approach

Temporally Decoupled Diffusion Model

We propose the **Temporally Decoupled Diffusion Model (TDDM)**. Inspired by recent video generation advancements (Chen et al. 2025; Song et al. 2025) that treat videos as image sequences, we leverage their structural commonalities with trajectory generation. We reformulate the trajectory generation task as a denoising process that is decoupled in the time dimension. Our core idea is to partition a complete trajectory into multiple temporal tokens and apply independent random noise to these tokens during training. This compels the model to not only learn the kinematic smoothness within each token but also to leverage global context to understand and reconstruct the complex temporal correlations between tokens.

To realize our proposed Temporally Decoupled Diffusion Model, the architecture must address two principal challenges: first, it must accommodate the application of distinct diffusion timestep encodings to each temporally decoupled segment; second, it must ensure kinematic consistency across the entire planning horizon. The overall architecture designed for this is shown in Figure 2.

The model's input processing begins with the decomposition of the trajectory into N temporal segments. We conceptually cluster these N segments into G macro-groups. N can be flexibly adjusted, whereas G dictates the instantiation of group-specific architectural components. Figure 2 illustrates our primary experimental setup where $G = 2$. Standard group-specific positional encodings are applied to each segment, which are then processed through a shared MLP projection layer, $\mathcal{F}_{\text{pre}}$, mapping each group τ_g into a partial hidden dimension. These G group features are concatenated along the channel dimension to form a unified feature tensor h:

$$h_g = \mathcal{F}_{\text{pre}}(\text{pos}(\tau_g)); \quad h = \text{Concat}(h_1, h_2, \ldots, h_G). \quad (5)$$

A key innovation is our **Temporally Decoupled Adaptive Layer Normalization (TD-AdaLN)** mechanism for injecting the independent diffusion timesteps. For each macro-group $g \in \{1, \ldots, G\}$, a conditional vector y_g is constructed by combining its group-specific diffusion timestep encoding t_g with a shared navigation encoding:

$$y_g = \mathcal{F}_{\text{time}}(t_g) + \mathcal{F}_{\text{navi}}(\text{navi}). \quad (6)$$

Here, navigation information $navi \in \mathbb{R}^{(K\times P)\times D_{\text{route}}}$ provides essential route guidance, extracted from the nuPlan benchmark as K route lanes with P points containing D_{route} coordinate features. Within each transformer block, these conditional vectors are used to generate segment-specific modulation parameters, $\text{params}_g = \mathcal{F}_{\text{adaLN}}(y_g)$. These parameters are then concatenated along the channel dimension to form a complete modulation tensor, params, which is subsequently applied to the main feature tensor h to conditionally modulate its normalization statistics.

The model's backbone consists of a series of DiT blocks where Self-Attention and Cross-Attention modules operate on the complete feature map h. Self-Attention facilitates information fusion among different temporal segments, enforcing internal consistency. Cross-Attention allows all segments to integrate unified external scene information c. Specifically, c is the fused vectorization of the nearest neighbors (a 2s history of kinematics, size, and category), map elements (polylines, traffic lights, speed limits), and static obstacles. These encoded context features serve as Key/Value pairs in the Cross-Attention layers. Finally, a dedicated feed-forward network (FFN) predicts the confidence score for the whole trajectory, while the segments are decoded independently by their corresponding group-specific FFNs.

Asymmetric Temporal CFG

Temporal decoupling enables unprecedented flexibility in reference. Rather than starting from a single, homogeneous noise state, we can assign distinct denoising starting points to different temporal segments. Based on this, we have designed a novel inference pipeline, as shown in Figure 3.

Our inference process performs two parallel model forward passes at each denoising step. However, unlike traditional CFG, these two passes do not simply represent the "with/without global condition." Instead, they represent two different temporal generation hypotheses:

1. **Unconditional Path**: This path represents the original full-sequence diffusion mode, where we start from a trajectory vocabulary composed entirely of Gaussian noise without temporal decoupling. That is, the same noise is added to all segments, and the noise level is consistent at each denoising step.
2. **Conditional Path**: Conversely, the Conditional Path defines a deterministic assumption about the long-term future. We divide the trajectory segments into two parts: near-term plan (e.g., the first $N/2$ segments) and far-term plan (e.g., the last $N/2$ segments). At each inference step, we construct an asymmetric, mixed-noise-level input: (1) **Near-term**: Start from full noise. (2) **Far-term**: Are always kept as the weakly noised original prototype (anchor) form, corresponding to a diffusion timestep of 0.001. This poses a completion task: given a future target, generate the optimal current path.
3. **CFG**: After obtaining the outputs from both paths, we fuse them according to the CFG formula:

$$\hat{\tau}^0_{g,\text{final}} = \hat{\tau}^0_{g,\text{uncond}} + w \cdot \left(\hat{\tau}^0_{g,\text{cond}} - \hat{\tau}^0_{g,\text{uncond}}\right), \quad (7)$$

the guidance scale w controls the adherence strength of the generated result to the long-term target, where $w > 1$ enforces stronger conditional guidance and $0 < w < 1$ weakens it. It is important to note that we actually use the reparameterization trick to directly predict the clean trajectory.

Training

Our training process begins by applying independent noise to each temporal segment. These segments are then conceptually divided into groups (e.g., near-term and far-term), where all segments within the same group share a common, randomly sampled diffusion timestep.

With the necessary components defined in the preliminary section, our training objective is designed as a multi-task

Type	Planner	Val14		Test14-hard		Test14	
		NR	R	NR	R	NR	R
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	90.15	76.19	90.05	91.63
	PDM-Hybrid	92.77	92.11	65.99	70.92	90.02	91.28
	GameFormer	79.94	79.78	68.70	70.05	79.88	82.05
	PLUTO	92.88	86.88	80.08	76.88	92.23	90.29
Learning-based	PDM-Open	53.53	54.24	33.51	33.89	52.81	57.23
	UrbanDriver	68.57	64.11	50.40	49.95	51.83	67.15
	PlanTF	84.27	76.95	69.70	61.61	85.36	79.58
	PLUTO w/o refine.	88.89	78.11	70.03	59.74	<u>89.90</u>	78.62
	Diffusion Planner	**89.76**	**82.56**	<u>75.67</u>	**68.56**	89.19	**82.55**
	TDDM (Ours)	**89.81**	<u>79.78</u>	**77.95**	<u>65.20</u>	**90.4**	<u>80.63</u>

Table 1: **Comparison of planning performance on nuPlan under both non-reactive (NR) and reactive (R) closed-loop evaluation.** Higher scores indicate better performance. Within each method group, the best result is highlighted in **bold**, and the second best is <u>underlined</u>. TDDM is trained on only 200k nuPlan scenarios.

ID	Traj. Token.	TD-AdaLN	Independent Noise	Asymmetric CFG	nuPlan (Test14-hard)
1	×	×	×	×	75.91
2	✓	×	×	×	76.60
3	✓	✓	×	×	74.94
4	✓	×	✓	×	73.76
5	✓	✓	✓	×	76.88
6	✓	✓	✓	✓	**77.95**

Table 2: **Ablation study of key components on the nuPlan Test14-hard benchmark.** ID 1 refers to a baseline diffusion planner with an anchor. We analyze the impact of sequentially adding Trajectory Tokenization (Traj. Token.), Temporally Decoupled AdaLN (TD-AdaLN), Independent Noise, and Asymmetric CFG. For Trajectory Tokenization, a consistent number of tokens is used. The model architecture incorporating Temporally Decoupled AdaLN is illustrated in Figure 2.

Token Num	Score
1	75.55
2	76.41
4	**77.95**
8	76.03
16	75.26

Table 3: Token ablation

CFG Scale	Score
0.75	76.04
1.00	77.22
1.25	**77.95**
1.50	77.02
1.75	74.70

Table 4: CFG scale ablation

learning paradigm. The core idea is not only for the model to reconstruct each part of the expert trajectory but also to assess whether it originates from the optimal trajectory prototype for the current scene, similar to the setup in previous multi-trajectory generation works (Cheng, Chen, and Chen 2024; Liao et al. 2025). We construct a two-part loss for each prototype: (1) high-fidelity reconstruction of the expert trajectory, starting from the trajectory prototype (anchor) that is closest in L2 distance; (2) accurate identification of which prototype is the best choice for the current scene. This joint training strategy is implemented through a hybrid loss function that combines an L1 reconstruction loss and a binary classification loss, simplified as follows:

$$\mathcal{L}(\theta) = \sum\nolimits_{k=1}^{M} [y_k \mathcal{L}_{\text{rec}}(\hat{\tau}_k^0, \tau_{gt}) + \lambda \mathcal{L}_{\text{BCE}}(\hat{s}_k, y_k)], \quad (8)$$

where $\hat{s}_k$ is the predicted confidence score representing the probability of prototype k being the optimal choice for the current scene and $y_k \in \{0, 1\}$ is the corresponding ground-truth label where only one element y_k equals 1. λ is a hyperparameter balancing the two terms. Since each predicted segment $\hat{\tau}_k^0$ includes one additional overlapping point to smoothly connect adjacent segments, we further penalize the endpoint mismatch by adding a L1 constraint in the reconstruction term.

$$\begin{aligned}\mathcal{L}_{\text{rec}}(\hat{\tau}_k^0, \tau_{gt}) = &||\hat{\tau}_k^0 - \tau_{gt}||_1 \\ &+ \gamma \sum_{g=1}^{G-1} ||\text{end}(\hat{\tau}_{k,g}^0) - \text{start}(\hat{\tau}_{k,g+1}^0)||_1,\end{aligned} \quad (9)$$

where $\text{end}(\cdot)$ and $\text{start}(\cdot)$ denote the last and first waypoints

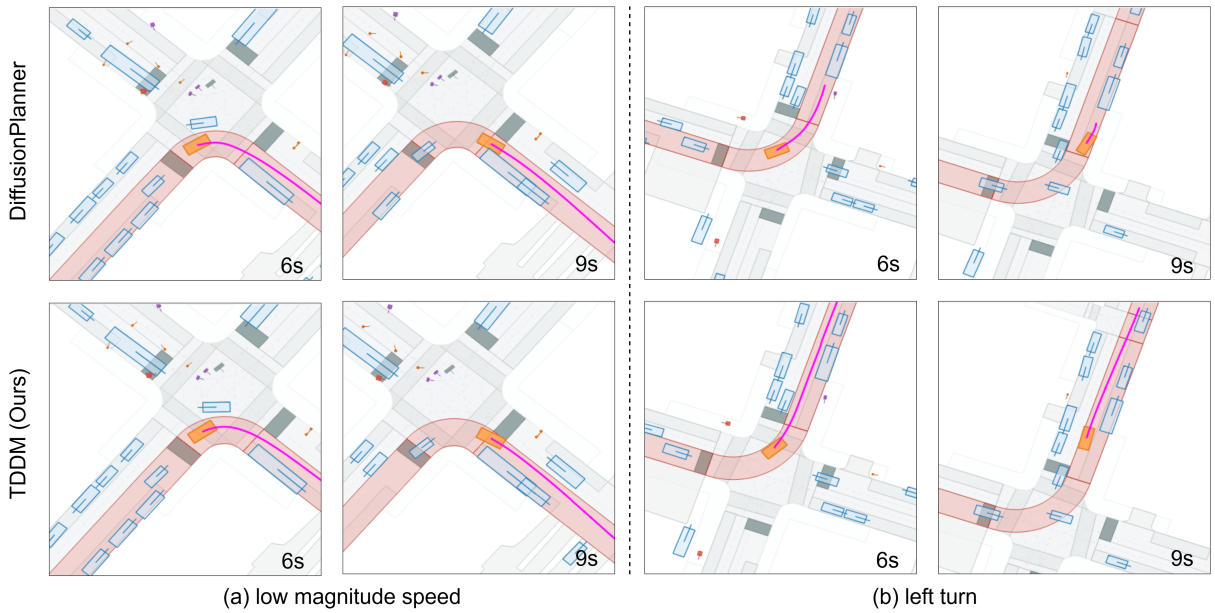

Figure 4: Comparison of Planned Trajectories by Diffusion Planner and our proposed TDDM method across two challenging traffic scenarios. The ego vehicle trajectory is shown in purple.

of a group-specific segment, respectively, and γ is a hyper-parameter to weigh the continuity constraint.

Experiment

Dataset and Benchmarks

We train and evaluate our model on nuPlan (Caesar et al. 2022), a large-scale multi-modal dataset containing ∼1,500 hours of real-world driving data from four cities. nuPlan features a diverse set of urban scenarios, including merging, roundabouts, and complex agent interactions. We test our model in both non-reactive and reactive settings using the following three benchmarks in nuPlan, each covering 14 scenario types: Val14 (1,118 regular validation scenarios), Test14-hard (272 long-tail, high-risk scenarios), and Test14-random (over 200 randomly sampled scenarios).

We benchmark our method TDDM, against a comprehensive set of baselines to ensure a thorough evaluation. Our comparison includes classic rule-based planners like the IDM (Treiber, Hennecke, and Helbing 2000) and the nuPlan challenge winner PDM (Dauner et al. 2023) in its various configurations (PDM-Closed, PDM-Open, PDM-Hybrid). Furthermore, we include a range of state-of-the-art learning-based models: the policy-gradient method UrbanDriver (Scheel et al. 2022); the game-theory-inspired GameFormer (Huang, Liu, and Lv 2023); two prominent transformer-based planners, PlanTF (Cheng et al. 2024) and PLUTO (Cheng, Chen, and Chen 2024); and DiffusionPlanner (Zheng et al. 2025), another recent diffusion-based approach, which serves as a direct point of comparison for our methodology.

Implementation Details

Architecture and Training. For a fair comparison, we align with Diffusion Planner (Zheng et al. 2025) by reusing its scene encoder, standard nuPlan perceptual inputs, and most implementation settings. We modify the diffusion backbone into a multi-modal, anchor-based architecture, using 20 trajectory anchors derived via k-means clustering. Due to resource constraints, the model was trained for 500 epochs on a 200k-scenario nuPlan subset using four NVIDIA 3080 GPUs (batch size 320, AdamW optimizer, learning rate 5×10^{-4}).

Inference. We employ a fast inference strategy with only 2 denoising steps via DPM-Solver++ and a VP noise schedule. In each step, only the highest-confidence trajectory is propagated. Classifier-Free Guidance (CFG) is accelerated through parallel batching. The model outputs an 8 seconds trajectory at 10 Hz.

Main Results

Quantitative result. Table 1 presents a quantitative comparison between TDDM and other state-of-the-art planners on the nuPlan benchmark. This benchmark features two core evaluation modes: Non-Reactive (NR) and Reactive (R). Specifically, in R mode, other agents dynamically react to the ego vehicle, whereas in NR mode, they merely replay historical behaviors. **Notably**, TDDM achieves competitive results using only 200k scenarios, highlighting its exceptional data efficiency compared to state-of-the-art learning-based methods like Diffusion Planner, which requires 1M scenarios.

In the non-reactive (NR) closed-loop evaluation, the performance of TDDM is on par with, and in some metrics

even surpasses, the current state-of-the-art learning-based planners. Specifically, on the general Val14 validation set, TDDM achieves a score of 89.81, which is on par with one of the leading learning-based methods, Diffusion Planner (89.76). The superiority of our methodology becomes particularly evident in more challenging scenarios. On the Test14-hard benchmark, which is specifically designed to evaluate robustness in long-tail scenarios, TDDM obtains a score of 77.95, significantly outperforming Diffusion Planner's 75.67. This result robustly demonstrates the effectiveness of the temporal decoupling mechanism in generating more robust and consistent plans when faced with complex and infrequent events. Furthermore, in the Test14-random test set, TDDM maintains a leading position with a score of 90.4.

Althrough TDDM's score in the reactive (R) setting is slightly below the top method, its strong performance in the non-reactive evaluation is more indicative of the planner's intrinsic quality. This, particularly its excellent results on the difficult benchmark, provides compelling evidence of its advanced planning capabilities.

Qualitative comparison. Figure 4 highlights TDDM's superior planning capabilities in two challenging scenarios where the baseline model fails. In the narrow right turn (a), the baseline Diffusion Planner generates a trajectory that collides with a parked bus. In the left turn with an obstacle (b), it makes a critical misjudgment, stopping unnecessarily behind a parked vehicle. In stark contrast, TDDM successfully navigates both situations by producing smooth, safe, and decisive maneuvers. These results demonstrate that our asymmetric temporal guidance effectively prevents the myopic, inconsistent decisions of the baseline, enabling TDDM to generate trajectories that are both safer and more coherent.

Ablation Study

We ablate key components of TDDM on the nuPlan Test14-hard benchmark (Table 2). Our baseline (ID 1), an anchor-based Diffusion Planner, scores 75.91. Simply introducing trajectory tokenization (ID 2), which segments the trajectory without any architectural or training modifications, provides a modest improvement to 76.60, suggesting that representing the trajectory as a sequence of tokens is inherently beneficial. Moreover, we observe an interesting phenomenon that introducing the TD-AdaLN module without independent noise (ID 3) or applying independent noise without the supportive TD-AdaLN architecture (ID 4) degrades performance to 74.94 and 73.76, respectively, while both components are combined (ID 5), the score improves to 76.88. It demonstrates that the TD-AdaLN module is essential for appropriately handling the segment-specific timestep information produced by the independent noise training. This reflects a critical synergy between the model architecture and the training paradigm, indicating that their alignment is necessary to jointly enhance the learning of robust temporal correlations. Finally, by incorporating our Asymmetric Temporal CFG at inference time (ID 6), the performance sees another significant leap to 77.95, validating the effectiveness of using a coherent far-term plan to guide the generation of the near-term trajectory.

Trajectory Tokenization. We investigated the optimal temporal granularity by varying the number of trajectory tokens (N). As detailed in Table 3, the model's performance exhibits a clear trend, peaking at $N = 4$ tokens (a 2-second segment length) with a score of **77.95**. Performance declines when the partition is either too coarse ($N < 4$) or too fine ($N > 4$). This indicates that a moderate granularity strikes an optimal balance between capturing complex temporal dynamics and maintaining long-term kinematic consistency.

Classifier-free Guidance. We ablated the guidance scale w of our Asymmetric Temporal CFG. As shown in Table 4, performance peaks at a guidance scale of $w = 1.25$. The results reveal a clear trend: scores decrease with either insufficient guidance ($w < 1.25$) or excessive guidance ($w > 1.25$). This indicates that the optimal scale strikes a balance between enforcing long-term goal consistency and preserving short-term reactive flexibility.

Conclusion

In this work, we introduced the Temporally Decoupled Diffusion Model (TDDM), a novel framework for autonomous driving motion planning that addresses the limitations inherent in monolithic trajectory generation. By reformulating the planning problem through a temporal decoupling paradigm, our model learns to capture the heterogeneous dependencies across a planning horizon. The core contributions—a temporally-decoupled training scheme with independent noise, a supporting TD-AdaLN architecture, and an Asymmetric Temporal Classifier-Free Guidance strategy for inference—work in synergy to produce trajectories that are both reactive to immediate conditions and consistent with long-term goals.

Experiments on the nuPlan benchmark demonstrate that TDDM **approaches or exceeds** state-of-the-art learning-based methods, showcasing exceptional robustness and coherence on the challenging Test14-hard subset. These results confirm that explicitly modeling distinct temporal structures is vital for high-performance planning in complex urban environments.

Limitiation and future work. Despite its strong performance, TDDM's efficacy in fully reactive, closed-loop simulations is an area for improvement. Future work will pursue several promising directions: enhancing agent interaction, potentially by extending our CFG with game-theoretic principles; removing the reliance on a predefined anchor set by exploring autoregressive generation from historical trajectories; and further exploiting the temporal token structure for more explicit and fine-grained trajectory guidance.

Acknowledgements

This work is part of the project "AI Algorithm and Model Development for Future-Oriented Intelligent Driving Systems" (Grant No. SYG2024087).

References

Aradi, S. 2022. Survey of Deep Reinforcement Learning for Motion Planning of Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 23(2): 740–759.

Bansal, M.; Krizhevsky, A.; and Ogale, A. 2018. ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst. arXiv:1812.03079.

Caesar, H.; Kabzan, J.; Tan, K. S.; Fong, W. K.; Wolff, E.; Lang, A.; Fletcher, L.; Beijbom, O.; and Omari, S. 2022. NuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles. arXiv:2106.11810.

Chen, B.; Martí Monsó, D.; Du, Y.; Simchowitz, M.; Tedrake, R.; and Sitzmann, V. 2025. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37: 24081–24125.

Chen, L.; Wu, P.; Chitta, K.; Jaeger, B.; Geiger, A.; and Li, H. 2024a. End-to-End Autonomous Driving: Challenges and Frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 10164–10183.

Chen, L.; Wu, P.; Chitta, K.; Jaeger, B.; Geiger, A.; and Li, H. 2024b. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, S.; Jiang, B.; Gao, H.; Liao, B.; Xu, Q.; Zhang, Q.; Huang, C.; Liu, W.; and Wang, X. 2024c. VADv2: End-to-End Vectorized Autonomous Driving via Probabilistic Planning. arXiv:2402.13243.

Cheng, J.; Chen, Y.; and Chen, Q. 2024. PLUTO: Pushing the Limit of Imitation Learning-based Planning for Autonomous Driving. arXiv:2404.14327.

Cheng, J.; Chen, Y.; Mei, X.; Yang, B.; Li, B.; and Liu, M. 2024. Rethinking Imitation-based Planners for Autonomous Driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 14123–14130.

Cui, H.; Radosavljevic, V.; Chou, F.-C.; Lin, T.-H.; Nguyen, T.; Huang, T.-K.; Schneider, J.; and Djuric, N. 2019. Multimodal Trajectory Predictions for Autonomous Driving using Deep Convolutional Networks. In *2019 International Conference on Robotics and Automation (ICRA)*, 2090–2096. IEEE Press.

Dauner, D.; Hallgarten, M.; Geiger, A.; and Chitta, K. 2023. Parting with Misconceptions about Learning-based Vehicle Motion Planning. In Tan, J.; Toussaint, M.; and Darvish, K., eds., *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, 1268–1281. PMLR.

Dhariwal, P.; and Nichol, A. 2021. Diffusion Models Beat GANs on Image Synthesis. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 8780–8794. Curran Associates, Inc.

Fan, H.; Zhu, F.; Liu, C.; Zhang, L.; Zhuang, L.; Li, D.; Zhu, W.; Hu, J.; Li, H.; and Kong, Q. 2018. Baidu Apollo EM Motion Planner. arXiv:1807.08048.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17853–17862.

Huang, Z.; Liu, H.; and Lv, C. 2023. 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 (ICCV)*, 3903–3913.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023a. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8350.

Jiang, C.; Cornman, A.; Park, C.; Sapp, B.; Zhou, Y.; Anguelov, D.; et al. 2023b. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9644–9653.

Leonard, J.; How, J.; Teller, S.; Berger, M.; Campbell, S.; Fiore, G.; Fletcher, L.; Frazzoli, E.; Huang, A.; Karaman, S.; Koch, O.; Kuwata, Y.; Moore, D.; Olson, E.; Peters, S.; Teo, J.; Truax, R.; Walter, M.; Barrett, D.; Epstein, A.; Maheloni, K.; Moyer, K.; Jones, T.; Buckley, R.; Antone, M.; Galejs, R.; Krishnamurthy, S.; and Williams, J. 2009. A Perception-Driven Autonomous Urban Vehicle. In Buehler, M.; Iagnemma, K.; and Singh, S., eds., *The DARPA Urban Challenge*, volume 56 of *Springer Tracts in Advanced Robotics*, 163–230. Springer Berlin Heidelberg. ISBN 978-3-642-03990-4.

Liao, B.; Chen, S.; Yin, H.; Jiang, B.; Wang, C.; Yan, S.; Zhang, X.; Li, X.; Zhang, Y.; Zhang, Q.; et al. 2025. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 12037–12047.

Peebles, W.; and Xie, S. 2023. Scalable Diffusion Models with Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4195–4205.

Pomerleau, D. A. 1988. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1.

Scheel, O.; Bergamini, L.; Wolczyk, M.; Osiński, B.; and Ondruska, P. 2022. Urban Driver: Learning to Drive from Real-world Demonstrations Using Policy Gradients. In Faust, A.; Hsu, D.; and Neumann, G., eds., *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, 718–728. PMLR.

Song, K.; Chen, B.; Simchowitz, M.; Du, Y.; Tedrake, R.; and Sitzmann, V. 2025. History-Guided Video Diffusion. arXiv:2502.06764.

Song, Y.; Durkan, C.; Murray, I.; and Ermon, S. 2021. Maximum likelihood training of score-based diffusion models. *Advances in neural information processing systems*, 34: 1415–1428.

Stilman, M.; and Kuffner, J. 2008. Planning among movable obstacles with artificial constraints. *The International Journal of Robotics Research*, 27(11-12): 1295–1307.

Treiber, M.; Hennecke, A.; and Helbing, D. 2000. Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev. E*, 62: 1805–1824.

Weng, X.; Ivanovic, B.; Wang, Y.; Wang, Y.; and Pavone, M. 2024. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15449–15458.

Yang, B.; Su, H.; Gkanatsios, N.; Ke, T.-W.; Jain, A.; Schneider, J.; and Fragkiadaki, K. 2024. 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*, 15342–15353.

Yin, T.; Zhang, Q.; Zhang, R.; Freeman, W. T.; Durand, F.; Shechtman, E.; and Huang, X. 2025. From Slow Bidirectional to Fast Autoregressive Video Diffusion Models. In *CVPR*.

Zheng, Y.; Liang, R.; ZHENG, K.; Zheng, J.; Mao, L.; Li, J.; Gu, W.; Ai, R.; Li, S. E.; Zhan, X.; and Liu, J. 2025. Diffusion-Based Planning for Autonomous Driving with Flexible Guidance. In *The Thirteenth International Conference on Learning Representations*.