WAM-Flow: Parallel Coarse-to-Fine Motion Planning via Discrete Flow Matching for Autonomous Driving

Supplementary Material

This appendix provides additional experimental results and implementation details to complement the main paper. Specifically, Section A presents extended evaluations on the nuScenes [3] datasets, along with additional qualitative experiments on NAVSIM. Section B provides pseudocode for the training and inference stages, respectively. Section C elaborates on the evaluation metrics, and Section D discusses implementation specifics. Finally, Section E discuss the limitation and future work.

A. Additional Experiments

A.1. nuScenes Results

We evaluate our method on the nuScenes dataset [3] following the NAVSIM benchmark perspective [4, 6], which focuses on collision rate as the primary metric. This emphasis stems from the established finding in NAVSIM that open-loop L2 distance exhibits negligible correlation with closed-loop performance. As shown in Table 1, our method achieves an average collision rate of **0.12%** under ST-P3 metrics, matching the performance of the best non-VLA model (UniAD). More notably, under the more comprehensive UniAD metrics, WAM-Flow sets a new state-of-the-art with the lowest average collision rate (**0.23%**) among all evaluated VLA methods. The model also demonstrates superior short-term safety, achieving a perfect **0.00%** collision rate at the 1-second horizon.

A.2. NAVSIM Qualitative Results

Figure 2, 3 and 4 visualizes 1-, 3- and 5-step results on NAVSIM, respectively. For straightforward driving scenarios (Figure 2), WAM-Flow generates acceptable trajectories with only 1-step denoising. For relatively complex scenarios (Figure 4), our method predicts reasonable results through a 5-step parallel coarse-to-fine process.

A.3. Driving VQA Results

To comprehensively evaluate the language understanding ability, we report the performance on driving VQA datasets in Table 3. Experimental results demonstrate that WAM-Flow achieves SOTA performance on nuScenes-QA [16] and DriveBench [22], and outperforms the commercial large model GPT-4o [1].

B. Pseudocode for Training and Inference

Algorithm 1 and 2 respectively describe the training and inference procedure.

Algorithm 1 Training

```
Require: model parameters $\theta$, time schedule $\beta_t$
Ensure: Optimized parameters $\theta^*$
Initialize model parameters $\theta$
while not converged do
    Sample batch $x_1 \sim q(x)$                                        ▷ Trajectory
    Sample $t \sim \mathcal{U}[0,1]$                                    ▷ Continuous time sampling
    $p_t(x|x_1) = \mathrm{softmax}(-\beta_t \cdot d(x, x_1))$           ▷ Compute transition probabilities
    $x_t \sim p_t(x|x_1)$                                               ▷ Sample noisy tokens
    $p^{\theta}_{1|t}(\cdot|x_t) = \mathrm{model}_\theta(x_t, c)$       ▷ Compute conditional distribution
    $\mathcal{L}_{\mathrm{CE}} = -\mathbb{E}\left[\sum_{i=1}^{D} \log p^{\theta,i}_{1|t}(x^i_1|x_t)\right]$   ▷ Compute loss
    Update $\theta$ via gradient descent on $\mathcal{L}_{\mathrm{CE}}$
end while
```

Algorithm 2 Inference

```
Require: Number of inference steps $n$
Ensure: Generated token sequence $x_1$
$h \leftarrow 1/n$                                                      ▷ Step size for Euler discretization
Initialize $x_0$: for each coordinate $i$, sample $x^i_0$ uniformly from vocabulary
for $k = 0, 1, \ldots, n-1$ do
    $t \leftarrow k \cdot h$                                            ▷ Current time in $[0, 1)$
    for $i = 1$ to $D$ in parallel do                                   ▷ Parallel processing of all coordinates
        Compute posterior: $p^{\theta,i}_{1|t}(\cdot|x_t) \leftarrow \mathrm{model}_\theta(x_t, c)$
        Sample target: $x^i_1 \sim p^{\theta,i}_{1|t}(\cdot|x_t)$
        Compute total transition rate: $\lambda_i \leftarrow \sum_{y^i \neq x^i_t} u^i_t(y^i, x^i_t|x^i_1)$
        Sample threshold: $Z_i \sim \mathcal{U}[0,1]$
        if $Z_i \le 1 - e^{-h\lambda_i}$ then                            ▷ Transition occurs with probability $1 - e^{-h\lambda_i}$
            Sample new token: $x^i_{t+h} \sim \frac{u^i_t(\cdot, x^i_t|x^i_1)}{\lambda_i}\left(1 - \delta_{x^i_t}(\cdot)\right)$
        else
            Retain current token: $x^i_{t+h} \leftarrow x^i_t$
        end if
    end for
    Advance time: $x_t \leftarrow x_{t+h}$
end for
return $x_1$                                                            ▷ Final denoised token sequence at $t = 1$
```

C. Detailed Explanation for Metrics

This section provides detailed definitions of the evaluation metrics used in our experiments.

C.1. NAVSIM-v1 Metrics

For NAVSIM-v1 [6], the primary evaluation metric is the Predictive Driver Model Score (PDMS), which integrates five key performance indicators:

$$\mathrm{PDMS} = \mathrm{NC} \times \mathrm{DAC} \times \frac{(5 \times \mathrm{TTC} + 2 \times \mathrm{C} + 5 \times \mathrm{EP})}{12} \tag{1}$$

Method	**Paradigm**	**Backbone**	**Collision (%) ↓**							
			ST-P3 metrics				**UniAD metrics**			
			1s	2s	3s	Avg.	1s	2s	3s	Avg.
End-to-End										
PreWorld [10]	-	-	-	-	-	-	0.19	0.57	2.65	1.14
ST-P3 [8]	-	-	0.23	0.62	1.27	0.71	-	-	-	-
Ego-MLP [12]	-	-	0.21	0.35	0.58	0.38	-	-	-	-
InsightDrive [17]	-	-	0.09	0.10	0.27	0.15	0.08	0.15	0.84	0.36
VAD-v2 [5]	-	-	0.07	0.10	0.24	0.14	-	-	-	-
UniAD [9]	-	-	**0.04**	**0.08**	**0.23**	**0.12**	0.05	0.17	0.71	0.31
End-to-End VLA										
Epona [23]	AR + Diff.	DiT-2.5B [15]	0.05	0.22	0.85	0.96	-	-	-	-
OmniDrive [20]	AR	LLaVA-7B [13]	**0.04**	0.46	2.32	0.94	-	-	-	-
DriveVLM [18]	AR	Qwen2-VL-7B [19]	0.10	0.22	0.45	0.27	-	-	-	-
GPT-Driver [14]	AR	GPT-4 [1]	0.04	0.12	0.36	0.17	0.07	0.15	1.10	0.44
AutoVLA [24]	AR	Qwen2.5-3B [2]	0.13	0.18	0.28	0.20	0.14	0.25	**0.53**	0.31
DME-Driver [7]	AR	LLaVA-7B [13]	-	-	-	-	0.05	0.28	0.55	0.29
Ours	DFM	Janus-1.5B [21]	**0.04**	0.10	**0.23**	**0.12**	**0.00**	**0.10**	0.60	**0.23**

Table 1. End-to-end motion planning performance on the nuScenes [3] dataset. We sort previous methods according to the average collision rate. Abbreviation: Diff.(Diffusion), AR (autoregressive), DFM (discrete flow matching).

Hyperparameter	***Stage 1***	***Stage 2***	***Stage 3***	***Stage 4***
	Embedding Training	**Pre-training**	**Supervised Fine-tuning**	**Reinforcement Learning**
Training Modules	Numerical Tokenizer	VLA	VLA	VLA
Training Parameters	0.4B	1.5B	1.5B	1.5B
Training Data	nuPlan (668K)	VQA (6.5M)	nuPlan (668K)	NAVSIM (103K)
Loss	$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{num}}$	$\mathcal{L}_{\text{CE}}$	$\mathcal{L}_{\text{CE}}$	$\mathcal{L}_{\text{GRPO}}$
Training Epochs	4	3	2	0.5
Batch Size	80	256	64	32
Optimizer	Adam	Adam	Adam	Adam
Learning Rate	1×10^{-5}	1×10^{-5}	5×10^{-6}	1×10^{-6}
Learning Rate Scheduler	constant	constant	cosine annealing	cosine annealing
Warm-up Steps	0	0	500	500
Gradient Accumulation Steps	1	1	1	1

Table 2. Key hyperparameters for different training stages.

Method	**Size**	**Type**	**nuScenes-QA**	**DriveBench**	
			Acc.↑	Percept. Clean↑	Percept. Corr.↑
Human	-	-	-	47.6	38.3
GPT-4o	-	Commercial	37.1	35.3	35.2
LLaVA-1.5	13 B	Open	26.2	23.4	23.3
Qwen2-VL	72B	Open	-	30.1	26.9
DriveLM	7B	Specialist	34.5	36.9	36.0
Ours	1.5B	Specialist	**49.2**	**44.1**	**44.8**

Table 3. Comparison on driving VQA datasets.

- **No at-fault Collision (NC)**: Penalizes collisions based on fault assignment. NC=1 indicates no at-fault collisions, NC=0.5 indicates one fault collision with static objects, and NC=0 indicates multiple fault collisions.
- **Drivable Area Compliance (DAC)**: Measures adherence to drivable areas (lanes, parking areas). DAC=1 when the ego bounding box remains entirely within drivable areas, and DAC=0 when any corner exits designated areas.
- **Ego Progress (EP)**: Quantifies navigation goal achievement as the ratio of actual progress to a search-based safe upper bound derived from PDM-Closed trajectories. The ratio is clipped to [0,1], with low or negative values discarded.
- **Time-to-Collision (TTC)**: Encourages maintenance of safe distances from other vehicles. TTC=1 when the minimum time-to-collision exceeds 0.9 seconds, and 0 other-

wise.
- **Comfort (C)**: Assesses kinematic constraints including acceleration and jerk. C=1 when all predefined thresholds are satisfied, and 0 upon any violation.

C.2. NAVSIM-v2 Metrics

For NAVSIM-v2 [4], the Extended Predictive Driver Model Score (EPDMS) incorporates additional safety and compliance measures:

$$\begin{aligned}\mathrm{EPDMS} = \mathrm{NC} \times \mathrm{DAC} \times \mathrm{DDC} \times \mathrm{TL} \times \\ \frac{(5 \times \mathrm{TTC} + 2 \times \mathrm{C} + 5 \times \mathrm{EP} + 5 \times \mathrm{LK} + 5 \times \mathrm{EC})}{22}\end{aligned} \quad (2)$$

- **Driving Direction Compliance (DDC)**: Penalizes reverse driving behavior. DDC=1 for reverse distance $<$ 2m, DDC=0.5 for $2-6$m, and DDC$=0$ for >6m.
- **Traffic Light Compliance (TLC)**: Measures obedience to traffic signals. TLC$=1$ when traffic rules are followed, and 0 upon violations.
- **Lane Keeping (LK)**: Evaluates lateral positioning relative to lane centerlines, scored continuously from 0 to 1.
- **History Comfort (HC)**: Assesses trajectory consistency with historical motion patterns, ranging from 0 to 1.
- **Extended Comfort (EC)**: Compares planned trajectories across consecutive frames for dynamic consistency, scored from 0 to 1.

C.3. nuScenes Metrics

For nuScenes, we follow the NAVSIM [4, 6] perspective, focusing only on the collision rate.

D. Implementation Details

In Table 2, we show the key hyperparameters for different training steps, including training modules, parameters, data, loss, epochs, batch sizes, optimizer, learning rate, learning rate scheduler, warm-up and gradient accumulation steps.

E. Limitation and Future Work

While WAM-Flow demonstrates promising results, several limitations warrant attention. First, our evaluation is conducted primarily in simulation environments (NAVSIM, nuScenes), which may not fully capture the complexities of real-world driving scenarios. Second, the GRPO reward is designed for and evaluated in simulation; its safety and performance terms require careful redesign to bridge the sim-to-real gap. Third, the model is trained and validated on existing benchmarks, which may not encompass the full long-tail distribution of real-world driving scenarios.

Future work will explore several directions. We plan to extend the framework to support variable-horizon planning and incorporate multi-modal sensor inputs (e.g., LiDAR, radar) for enhanced robustness. We also plan to investigate learning a world model as a more generalizable alternative to simulator-based rewards. Finally, real-world deployment and testing will be essential to validate the model's performance under actual driving conditions.

Figure 2. For straightforward driving scenarios on NAVSIM, our method achieves acceptable outcomes with just 1-step denoising.

Figure 3. Visualization of the 3-step refinement results on NAVSIM.

Figure 4. For relatively complex scenarios on NAVSIM, our model generates reasonable results through a 5-step coarse-to-fine trajectory prediction process.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 3

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 3

[3] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2, 3

[4] Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Pseudo-simulation for autonomous driving. In *Conference on Robot Learning (CoRL)*, 2025. 2, 4

[5] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024. 3

[6] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2, 4

[7] Wencheng Han, Dongqian Guo, Cheng-Zhong Xu, and Jianbing Shen. Dme-driver: Integrating human decision logic and 3d scene perception in autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3347–3355, 2025. 3

[8] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022. 3

[9] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023. 3

[10] Xiang Li, Pengfei Li, Yupeng Zheng, Wei Sun, Yan Wang, and Yilun Chen. Semi-supervised vision-centric 3d occupancy world model for autonomous driving. *arXiv preprint arXiv:2502.07309*, 2025. 3

[11] Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. *arXiv preprint arXiv:2506.08052*, 2025. 1

[12] Zhiqi Li, Zhiding Yu, Shiyi Lan, Jiahan Li, Jan Kautz, Tong Lu, and Jose M Alvarez. Is ego status all you need for open-loop end-to-end autonomous driving? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14864–14873, 2024. 3

[13] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 3

[14] Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023. 3

[15] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 3

[16] Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4542–4550, 2024. 2

[17] Ruiqi Song, Xianda Guo, Hangbin Wu, Qinggong Wei, and Long Chen. Insightdrive: Insight scene representation for end-to-end autonomous driving. *arXiv preprint arXiv:2503.13047*, 2025. 3

[18] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*, 2024. 3

[19] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 3

[20] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22442–22452, 2025. 3

[21] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12966–12977, 2025. 3

[22] Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are vlms ready for autonomous driving? an empirical study from the reliability, data and metric perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6585–6597, 2025. 2

[23] Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, Xun Cao, and Wei Yin. Epona: Autoregressive diffusion world model for autonomous driving.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 3

[24] Zewei Zhou, Tianhui Cai, Seth Z Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. *arXiv preprint arXiv:2506.13757*, 2025. 3