TRAINING-FREE ADAPTIVE TIME GRIDS FOR FLOW-BASED VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Flow matching has recently emerged as a powerful action generation paradigm for Vision-Language-Action (VLA) models, where continuous action trajectories are generated by solving a learned probability-flow ordinary differential equation (ODE). Despite their strong modeling capability, existing inference pipelines for flow-based VLA models typically rely on uniform temporal discretization, implicitly assuming that the transport trajectory evolves with approximately constant complexity over time. In this work, we show that this assumption does not hold in practice: the differential entropy of intermediate flow states exhibits highly non-uniform evolution, revealing that different temporal regions contain different levels of uncertainty and trajectory complexity. Motivated by this observation, we propose Entropy- and Geometry-aware Adaptive Time Grids (**EATG**), a training-free inference framework that dynamically reallocates the denoising budget according to the intrinsic dynamics of the learned flow. EATG first performs a lightweight calibration rollout to estimate the entropy evolution along the trajectory and constructs an adaptive timestep schedule based on uncertainty dynamics. Furthermore, recognizing that reinforcement-learning post-training can alter the transport geometry inherited from conditional flow matching, EATG incorporates local velocity-field geometry as a complementary signal to capture regions where entropy alone is insufficient. EATG is fully plug-and-play: it requires neither additional training nor model modification and can be directly applied to flow-based action experts obtained through either supervised fine-tuning or reinforcement-learning post-training. Experiments on LIBERO, ManiSkill, CALVIN, and Meta-World report improvements over uniform discretization across in-distribution and out-of-distribution tasks, with the calibration cost reported separately. These results highlight the importance of uncertainty- and geometry-aware temporal discretization as an effective approach for improving flow-based VLA policies at inference time.

1 INTRODUCTION

Flow matching has recently emerged as a powerful paradigm for learning continuous generative models, achieving strong performance in high-dimensional generation tasks such as image synthesis (Lipman et al., 2022) and, more recently, vision-language-action (VLA) models for robotic control (Black et al., 2024) (Intelligence et al., 2025). In flow-based VLA systems, actions are generated by transporting samples from an initial noise distribution to a target action distribution through a learned velocity field. Given their strong modeling capacity, these policies can achieve impressive imitation performance after supervised fine-tuning (SFT). However, improving robustness, generalization, and long-horizon task performance in complex robotic environments remains challenging. Reinforcement learning (RL) post-training has therefore become an important approach for further adapting flow-based VLA policies through environment interaction (Chen et al., 2025).

Despite the success of flow-based VLA models, their inference efficiency remains constrained by the numerical discretization of the underlying continuous transport process. Since flow matching defines generation through a deterministic ordinary differential equation (ODE), practical inference requires approximating the continuous trajectory with a finite number of solver steps. Existing approaches typically employ fixed time discretization strategies, such as uniform, cosine, or power schedules. However, these manually designed schedules assume that the complexity of the transport process is

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

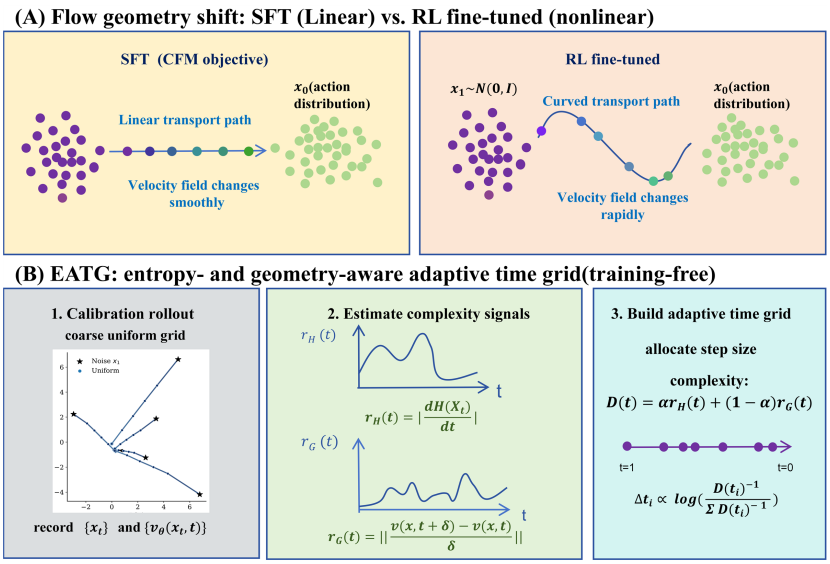


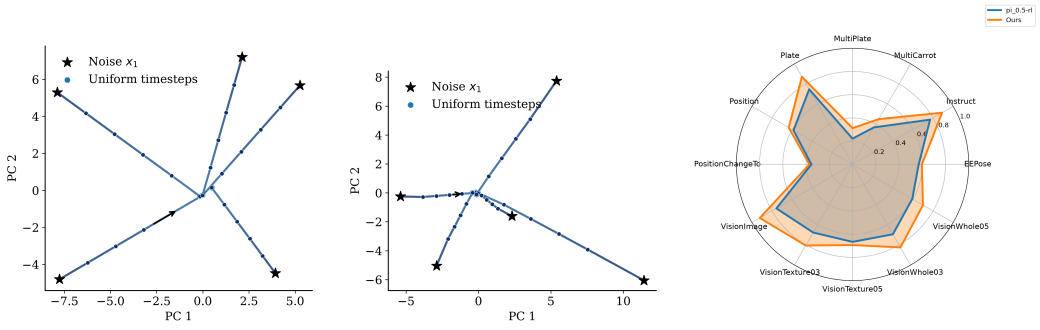
Figure 1: (**A**) Overview of Entropy- and Geometry-aware Adaptive Time Grid (EATG). (**B**) EATG performs a training-free calibration rollout to characterize the complexity of the flow trajectory through entropy dynamics and velocity-field geometry.

relatively uniform across time, ignoring the fact that uncertainty and geometric complexity of the learned velocity field may vary substantially along the trajectory.

Recent studies have explored information-theoretic adaptive schedulers that construct non-uniform time grids according to the evolution of conditional entropy (Stancevic et al., 2026). By allocating inference steps according to information gain, these methods improve sampling efficiency in diffusion and flow-based generative models under low-step regimes. However, such approaches are primarily developed for models whose intermediate transport geometry remains consistent with the original training objective. In contrast, flow-based VLA policies can undergo substantial changes after downstream optimization, especially after RL fine-tuning, where the velocity field is optimized according to task-level rewards rather than solely preserving the conditional flow matching objective (Liu et al., 2026a) (Chen et al., 2025). Consequently, entropy evolution alone may not fully capture the local complexity of the resulting transport trajectory.

In this work, we investigate adaptive discretization for flow-based VLA models from an information-theoretic and geometric perspective. We first analyze the evolution of marginal differential entropy in flow matching trajectories and establish its connection with endpoint uncertainty through conditional entropy decomposition. Under the linear interpolation assumption of conditional flow matching, entropy evolution provides a direct characterization of how uncertainty changes along the transport path. This observation motivates entropy-based timestep allocation for SFT models, whose velocity fields approximately preserve the original conditional transport geometry. However, RL post-training can distort the intermediate trajectory geometry by modifying the learned velocity field. We visualize velocity-fields of SFT models and RL finetuned models in Figure 2 and find that, for RL-adapted VLA policies, regions with similar entropy evolution may exhibit substantially different local velocity variations. Therefore, entropy alone is insufficient to characterize inference difficulty. To address this limitation, we further incorporate velocity-field geometry by measuring local temporal variations of the learned flow. Combining entropy dynamics and geometric properties, we propose a training-free adaptive timestep scheduler that dynamically allocates inference computation according to trajectory complexity.

The proposed scheduler is plug-and-play and can be directly integrated into existing flow-based VLA inference pipelines without modifying model parameters or training objectives. It performs a lightweight calibration rollout to estimate trajectory complexity and constructs a non-uniform timestep schedule for subsequent inference using the same initial noise. Extensive experiments on multiple robotic manipulation benchmarks demonstrate that our method improves performance



(a) SFT model (b) RL-finetuned model (c) Maniskill OOD evaluation results

Figure 2: (a)(b)Velocity-field visualization of flow-based VLA policies on Meta-World. (a)The SFT model maintains the nearly linear transport geometry learned from conditional flow matching, with stable velocity magnitude and direction along the trajectory. (b)After RL fine-tuning, the velocity field becomes highly non-uniform and the trajectory exhibits increased curvature, indicating a deviation from the original linear flow assumption. These geometric differences motivate the geometry-aware component of EATG.

for both SFT and RL-finetuned flow-based VLA models, achieving consistent gains in both in-distribution and out-of-distribution generalization.

Our main contributions are summarized as follows:

- We provide an information-theoretic analysis of entropy evolution in flow-based VLA trajectories and establish its relationship with endpoint uncertainty under conditional flow matching.
- We reveal that RL post-training can alter the transport geometry of flow-based VLA policies, making velocity-field geometry an important complement to entropy-based timestep allocation.
- We propose a training-free entropy- and geometry-aware adaptive timestep scheduler that dynamically allocates inference computation for flow-based VLA models without additional training.
- We demonstrate through extensive experiments on robotic manipulation benchmarks that the proposed method consistently improves both SFT and RL-finetuned policies under limited inference budgets.

2 Related Works

Adaptive timestep scheduling has recently emerged as an effective approach for improving the efficiency of continuous generative models under limited inference budgets. Since diffusion and flow-based models both approximate continuous-time generation processes through numerical discretization, a key challenge is how to allocate inference steps according to the intrinsic complexity of the generation trajectory.

Entropy-based scheduling. A recent line of work explores information-theoretic principles for adaptive time allocation. Stancevic et al. (Stancevic et al., 2026) introduce Entropic Time Schedulers for diffusion models in image generation, where the conditional entropy is used as a temporal coordinate such that each discretization step contributes approximately equal information gain. Their method provides tractable entropy estimation and demonstrates improvements under low-NFE sampling regimes.

Beyond diffusion models, entropy-based scheduling has also been extended to continuous transport models. Entropy Across the Bridge (Trentini et al., 2026) proposes conditional–marginal entropy-rate objectives for flow matching and Schrödinger bridge samplers, enabling training-free construction of nonuniform time grids. For discrete diffusion models, Entropic Discrete Schedule (Foresti

et al., 2026) further utilizes entropy production rates to construct schedules with approximately constant information gain.

Although these methods demonstrate the effectiveness of entropy as a measure of sampling difficulty, they mainly assume that entropy evolution sufficiently characterizes the complexity of the underlying generative trajectory. Such an assumption is suitable for models that preserve their original training-time transport geometry, but may become inaccurate when the learned flow field is modified during downstream optimization.

Adaptive inference for VLA policies. ProbeFlow probes the cosine similarity between initial and lookahead velocities to skip redundant flow steps, primarily targeting action-decoding latency (Fang et al., 2026). AdaVLA uses a trajectory-curvature proxy to adapt both flow-step counts and MLP pruning for on-device acceleration (Han et al., 2026). AAC addresses a different control decision: it estimates uncertainty from multiple sampled action chunks and adjusts how long the robot executes a chunk before observing again (Liang et al., 2026). EATG keeps the executed chunk horizon fixed and reallocates the *integration grid* using complementary entropy-evolution and velocity-geometry signals, including for RL-finetuned policies. Unlike ProbeFlow and AdaVLA, its objective is not solely to reduce latency; its calibration pass adds evaluations and must be included in cost comparisons. Geometry-aware inference itself is shared with those methods, so the contribution depends on demonstrating that the entropy–geometry combination improves over strong matched-cost schedules.

3 METHODOLOGY

3.1 ENDPOINT UNCERTAINTY CHARACTERIZATION

We first introduce an information-theoretic perspective to characterize the uncertainty evolution of intermediate states.

Given an endpoint condition

$$\mathcal{P} = (X_1, X_0), \tag{1}$$

where X_1 and X_0 denote the initial noise state and final action state, respectively, the intermediate state X_t provides partial information about the endpoint condition.

We measure the remaining uncertainty of the endpoint condition given X_t using conditional Shannon entropy:

$$H(\mathcal{P}|X_t). \tag{2}$$

A larger value indicates that the current intermediate state contains less information about the final action, corresponding to a more ambiguous region of the transport trajectory. Conversely, a smaller value indicates that the flow has progressively refined toward a deterministic solution.

Theorem 1 (Conditional entropy decomposition)**.** *For an endpoint condition $\mathcal{P} = (X_1, X_0)$ and an intermediate state X_t, the conditional entropy satisfies:*

$$H(\mathcal{P}|X_t) = H(X_t|\mathcal{P}) + H(\mathcal{P}) - H(X_t). \tag{3}$$

Proof. The mutual information between $\mathcal{P}$ and X_t can be expressed as:

$$I(\mathcal{P}; X_t) = H(\mathcal{P}) - H(\mathcal{P}|X_t), \tag{4}$$

and equivalently:

$$I(\mathcal{P}; X_t) = H(X_t) - H(X_t|\mathcal{P}). \tag{5}$$

Equating the two expressions yields Eq. (3).

□

3.2 ENTROPY EVOLUTION OF FLOW TRAJECTORIES

Conditional flow matching (CFM) learns a velocity field by matching a prescribed conditional transport path. A commonly used path is the linear interpolation:

$$X_t = (1 - t)X_0 + tX_1. \tag{6}$$

Under this deterministic interpolation, X_t is fully determined by the endpoint condition $\mathcal{P}$:

$$H(X_t|\mathcal{P}) = 0. \tag{7}$$

Therefore, according to Theorem 1, we have:

Corollary 1 (Conditional entropy under linear interpolation)**.** *For deterministic linear interpolation,*

$$H(\mathcal{P}|X_t) = H(\mathcal{P}) - H(X_t). \tag{8}$$

The uncertainty evolution satisfies:

$$\frac{d}{dt}H(\mathcal{P}|X_t) = -\frac{d}{dt}H(X_t). \tag{9}$$

The above result indicates that under the CFM assumption, the endpoint uncertainty evolution can be characterized through the marginal entropy variation of intermediate states.

Let $p_t(x)$ denote the marginal distribution of X_t. The probability flow satisfies the continuity equation:

$$\partial_t p_t(x) = -\nabla_x \cdot (p_t(x)v_\theta(x,t)). \tag{10}$$

The entropy evolution of the probability flow is given by:

Theorem 2 (Entropy evolution of probability flow)**.** *For a deterministic flow:*

$$\frac{dX_t}{dt} = v_\theta(X_t, t), \tag{11}$$

the marginal entropy satisfies:

$$\frac{d}{dt}H(X_t) = \mathbb{E}_{x_t\sim p_t}[\nabla_x \cdot v_\theta(x_t,t)]. \tag{12}$$

For CFM-trained models, the linear interpolation assumption further enables an analytical velocity-based expression:

Corollary 2 (Velocity-based entropy evolution)**.** *Under the linear interpolation path, the entropy evolution can be written as:*

$$\frac{dH(t)}{dt} = \frac{1}{t}\mathbb{E}_t[v_t(X_t)\cdot X_t] + \frac{1-t}{t}\mathbb{E}_t[\|v_t(X_t)\|^2]. \tag{13}$$

This result provides a closed-form entropy estimator for models that preserve the original CFM geometry.

3.3 Geometry-aware Complexity Estimation

Although CFM provides an analytical entropy evolution, post-training optimization such as reinforcement learning may modify the learned velocity field. The endpoint condition remains unchanged, but the intermediate transport geometry may deviate from the ideal linear interpolation.

Therefore, the velocity-based closed-form entropy estimator may become inaccurate. To handle general flow models, we additionally consider the local geometry of the learned velocity field.

The temporal variation of the velocity field is defined as:

$$G(t,\delta) = \frac{\|v_\theta(x_t, t+\delta) - v_\theta(x_t,t)\|_2}{|\delta|}. \tag{14}$$

A large value indicates rapid variation of the vector field, corresponding to a high-curvature region where smaller integration steps are required.

For general flow models, entropy evolution can be estimated through the divergence term:

$$E_H(t) = |\nabla_x \cdot v_\theta(x_t,t)|\,. \tag{15}$$

Since directly computing divergence is expensive in high-dimensional action spaces, we use Hutchinson's trace estimator:

$$\nabla_x \cdot v_\theta(x_t,t) \approx \epsilon^T \nabla_x v_\theta(x_t,t)\epsilon, \tag{16}$$

where

$$\epsilon \sim \mathcal{N}(0, I). \tag{17}$$

The entropy and geometry terms characterize complementary aspects of the transport dynamics. Since their numerical scales are different, both terms are normalized over the calibration trajectory:

$$\hat{E}_H(t) = \mathrm{Norm}(E_H(t)) \quad \hat{E}_G(t) = \mathrm{Norm}(E_G(t)). \tag{18}$$

The final local complexity score is defined as:

$$D(t) = \alpha \hat{E}_H(t) + (1 - \alpha)\hat{E}_G(t), \tag{19}$$

where α controls the relative contribution of entropy evolution and velocity geometry. Rather than using a fixed coefficient, we adapt α according to the global curvature of the sampled trajectory.

Specifically, we estimate the trajectory curvature as:

$$K = \frac{1}{M} \sum_i \frac{\|v_\theta(x_{t_i}, t_i) - v_\theta(x_{t_{i+1}}, t_{i+1})\|}{\|v_\theta(x_{t_i}, t_i)\| + \epsilon}, \tag{20}$$

and compute

$$\alpha = \exp(-\lambda K). \tag{21}$$

This design assigns larger weights to entropy evolution for nearly linear CFM trajectories, while automatically increasing the contribution of velocity geometry for highly curved RL-adapted flows.

3.4 Calibration-based Entropy and Geometry-aware Scheduling

Based on the above complexity analysis, we propose a training-free Entropy- and Geometry-aware Timestep Scheduling (EATG), which adaptively redistributes the inference budget according to the uncertainty evolution of each trajectory.

The key intuition is that slow entropy contraction indicates that the current state remains uncertain about the endpoint. In such regions, multiple plausible action trajectories may coexist, and finer temporal discretization is required. Conversely, rapid entropy contraction indicates that the trajectory has entered a deterministic refinement stage where larger integration steps are sufficient.

Given an initial noise realization x_1, EATG first performs a calibration rollout using a uniformly discretized temporal grid

$$\mathcal{T} = \{t_0, t_1, \ldots, t_M\}. \tag{22}$$

During this pass, the intermediate states and velocity fields are recorded, from which the local complexity profile $\{r_i\}_{i=1}^M$ is estimated using Eq. (19). The adaptive timestep weights are then computed as

$$\Delta t \propto w_i = \log(1 + \frac{r_i^{-1}}{\sum_j r_j^{-1}}). \tag{23}$$

where intervals with larger complexity receive smaller temporal spacing and therefore more inference steps.

The final timestep sequence $\{\tau_k\}_{k=0}^N$ is constructed according to the cumulative distribution of $\{w_i\}$. The policy is subsequently evaluated using a second rollout with the adaptive schedule while keeping the same initial noise realization. This two-pass design eliminates randomness introduced by different initial conditions and isolates the effect of

4 Experimental Results

4.1 Experimental Setup

Benchmarks We evaluate EATG on both simulation benchmarks and real-world robotic manipulation to demonstrate its effectiveness across different training paradigms and deployment settings. Specifically, simulation experiments are conducted on four widely-used manipulation benchmarks: LIBERO (Liu et al., 2023), ManiSkill (Mu et al., 2021), Meta-World (Yu et al., 2020), and CALVIN (Mees et al., 2022). In addition, we perform real-world deployment experiments on a a Franka Research3 robot.

Model	Spatial	Object	Goal	Long	Avg.
$\pi_{0.5}$(SFT)	98.8	98.2	98.0	92.4	96.9
EATG(Ours)	**99.2**	**99.0**	**98.4**	**94.2**	**97.7**

Table 1: Evaluation results on the LIBERO benchmark, evaluated based on the success rate (%).

Method	ManiSkill (IND)	MetaWorld (IND)	ManiSkill (OOD)	CALVIN (OOD)
Flow-SDE (π_{RL})	90.9	70.7	49.3	79.1
EATG (Ours)	**92.2**	**72.7**	**60.4**	**81.8**

Table 2: Comprehensive IND and OOD performance comparison across there benchmarks.

Flow-based VLAs Since EATG is a training-free inference method that does not depend on policy optimization, we evaluate it on both SFT and RL fine-tuned flow-based VLA policies. For simulation benchmarks, we use the publicly released $\pi_{0.5}$ (Intelligence et al., 2025)checkpoints, including the full-dataset SFT checkpoint from OpenPI and the RL-finetuned checkpoints provided by RLinf. For real-world deployment, we adopt the official π_0 (Black et al., 2024) policy supported by the OpenPI. In all experiments, EATG is applied directly at inference time without any additional training or policy modification.

4.2 Results on different training-paramdam models

We evaluate our proposed EATG on both supervised fine-tuned (SFT) and reinforcement learning (RL) adapted flow-based VLA models. Unlike existing approaches that modify model parameters or require additional training, EATG performs training-free inference-time optimization by dynamically allocating denoising steps according to the local complexity of the learned flow trajectory.

4.2.1 SFT models

We first evaluate EATG on SFT models, where the learned velocity field is optimized under the conditional flow matching objective and therefore approximately follows the prescribed linear interpolation trajectory.

As shown in Table 1, EATG consistently improves the performance of $\pi_{0.5}$ across all categories of the LIBERO benchmark. Specifically, EATG improves the overall success rate from 96.9% to 97.7%, with gains on Spatial (+0.4%), Object (+0.8%), Goal (+0.4%), and Long (+1.8%) tasks.

These improvements demonstrate that adaptive timestep allocation can further improve the numerical integration accuracy even when the underlying flow model already follows the CFM trajectory. By assigning more inference steps to regions with higher entropy variation, EATG better resolves the uncertain stages of the transport process while avoiding unnecessary evaluations in relatively deterministic regions.

4.2.2 RL finetuned models

We further evaluate EATG on RL-finetuned flow-based VLA models. In this work, we use flow-SDE (Chen et al., 2025) checkpoints trained by RLinf (Yu et al., 2026) as baseline. Different from SFT models, RL optimization directly targets task-level rewards and does not explicitly preserve the intermediate transport trajectory imposed by CFM. Therefore, the learned velocity field may exhibit non-uniform geometric complexity, making fixed timestep discretization suboptimal.

Table 2 compares EATG with the original RL-finetuned model using Flow-SDE across both in-distribution (IND) and out-of-distribution (OOD) benchmarks. EATG consistently improves performance on all evaluated environments.

Environment	Variation-Version-Type	Flow-SDE (π_{RL})	EATG (Ours)
In distribution	Main-v3-train	90.9	**92.2**
Visual -Language Variations	Instruct-v1-test	77.0	**89.0**
	VisionImage-v1-test	78.8	**92.2**
	VisionTexture03-v1-test	69.6	**80.6**
	VisionTexture05-v1-test	58.0	**69.4**
	VisionWhole03-v1-test	69.6	**82.5**
	VisionWhole05-v1-test	55.0	**70.0**
Semantic Reasoning	MultiCarrot-v1-test	36.8	**45.0**
	MultiCarrot-v1-train	49.5	**61.5**
	MultiPlate-v1-test	29.4	**31.3**
	MultiPlate-v1-train	22.3	**26.3**
Action Execution	PositionChangeTo-v1-test	36.2	**37.2**
	Position-v1-tes	54.5	**63.4**

Table 3: Specific generalization evaluation results in the ManiSkill OOD setting.

Benchmark	Method	NFE	Time (s)
Meta-World	Uniform	5	28
	EATG(twice rollout)	10	30

Table 4: Measured inference cost on Meta-World. EATG includes a calibration rollout and uses more velocity evaluations than uniform sampling.

On IND tasks, EATG improves the success rate from 90.9% to 92.2% on ManiSkill and from 70.7% to 72.7% on MetaWorld. More importantly, EATG achieves substantial improvements on OOD generalization, increasing performance from 49.3% to 60.4% on ManiSkill-OOD and from 79.1% to 81.8% on CALVIN-OOD. Maniskill OOD benchmark consists three kinds of out of distribution tasks: **Vision**, **Semantic** and **Execution** (Liu et al., 2026b). Detailed performance is included in Figure 2c.

The larger improvements on OOD benchmarks indicate that adaptive temporal allocation is particularly beneficial when the learned flow trajectory deviates from the original training distribution. In these cases, local geometric variations of the velocity field become more significant, and EATG dynamically allocates additional integration steps to challenging regions. These results validate the effectiveness of combining entropy evolution and velocity geometry for inference-time adaptation of flow-based VLA models.

4.2.3 Inference Efficency

Since EATG performs a calibration rollout before adaptive inference, an additional computational overhead is introduced compared with standard uniform discretization. However, this overhead is marginal in practice due to the lightweight nature of ODE-based flow sampling. Specifically, the calibration phase only requires forward evaluations of the velocity field without gradient computation, and the generated trajectory is discarded after estimating the entropy and geometry profiles.

To quantify inference cost, we compare wall-clock time for the same number of Meta-World tasks. Table 4 reports 10 NFE for uniform sampling and 10 NFE for EATG, including its calibration rollout($5\text{NFE} \times 2$). The measured times are 28 and 30 seconds, respectively, under this protocol. This is a cost measurement, not a matched-NFE comparison.

4.3 Ablation: Geometry-Probe Radius

We examine the sensitivity of an inference-time geometry probe to its temporal radius r on the released ManiSkill FlowSDE checkpoint. Each configuration is evaluated with the same checkpoint and reset protocol at random seeds 0 and 1, with 320 episodes per seed. The mean in Table 5 is the

Time grid	Total NFE	Seed 0	Seed 1	Mean (%)
Uniform	4	269/320	287/320	86.88
Geometry probe, $r = 0.20$	16	274/320	283/320	87.03
Geometry probe, $r = 0.02$	16	272/320	274/320	85.31

Table 5: Exploratory ManiSkill grid-radius ablation over two seeds. The geometry probe uses strength 0.5 and four final integration steps. The table does not isolate the entropy and geometry components of EATG.

arithmetic mean of the two seed-level success-once rates. The adaptive variants use a 12-evaluation probe followed by four Euler updates; thus their total 16 NFE is not matched to the 4-NFE uniform baseline.

The smaller radius reduces the two-seed mean by 1.72 percentage points relative to $r = 0.20$. At $r = 0.20$, the mean differs from uniform by only 0.16 points despite four times as many velocity evaluations. These results motivate matched-cost component ablations before attributing any gain specifically to the combination of entropy and geometry.

4.4 Real-World Deployment

To evaluate the practical applicability of EATG beyond simulation, we conduct real-world deployment experiments on a Franka Research3 robot arm using the official π_0 policy.

We evaluate four representative manipulation tasks, including Cap Alignment, Pen Insertion, Weight Insertion, and Object Placement, covering alignment, insertion, and pick-and-place skills. For each task, we collect 40 teleoperated demonstrations using a SpaceMouse to train the π_0 policy. During evaluation, both the baseline policy and EATG use the identical trained checkpoint. EATG is applied only during inference without any additional training, fine-tuning, or policy modification.

As shown in Table 6, EATG consistently improves the success rate across all four real-world manipulation tasks without any additional training or policy modification. The overall success rate increases from 50% to 85%, corresponding to an absolute improvement of 35 percentage points over the original π_0 policy. In particular, EATG achieves substantial gains on the Cap Alignment (+45 pp), Pen Insertion (+45 pp), and Weight Insertion (+40 pp) tasks, while also improving the already strong performance on Object Placement from 90% to 100%.

Manipulation Task	**π_0 Baseline**	**EATG (Ours)**	**Improvement**
Cap Alignment	30 (12/40)	**75** (30/40)	+45
Pen Insertion	50 (20/40)	**95** (38/40)	+45
Weight Insertion	30 (12/40)	**70** (28/40)	+40
Object Placement	90 (36/40)	**100** (40/40)	+10
Overall	50	**85**	+35

Table 6: Success rates (%) results on four real-world manipulation tasks. Improvement denotes the absolute gain over the π_0 baseline in percentage points.

5 Conclusion

In this work, we study the problem of inference-time timestep allocation for flow-based vision-language-action (VLA) models. Unlike conventional uniform discretization strategies that assume homogeneous transport dynamics, we show that the uncertainty and geometric complexity of flow trajectories vary significantly across different temporal regions. By analyzing the entropy evolution of the probability flow, we establish a connection between intermediate-state uncertainty and adaptive temporal discretization.

Future work will explore more accurate estimators of trajectory complexity and extend adaptive scheduling to broader classes of generative policies beyond flow matching, including stochastic transport models and large-scale embodied foundation models.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Kang Chen, Zhihao Liu, Tonghe Zhang, Zhen Guo, Si Xu, Hao Lin, Hongzhi Zang, Quanlu Zhang, Zhaofei Yu, Guoliang Fan, et al. πrl: Online rl fine-tuning for flow-based vision-language-action models. *arXiv preprint arXiv:2510.25889*, 2025.

Zhou Fang, Jiaqi Wang, Yi Zhou, and Qiongfeng Shi. Probeflow: Training-free adaptive flow matching for vision-language-action models. arXiv:2603.17850, 2026.

Alberto Foresti, Mustapha Bounoua, Giulio Franzese, Luca Ambrogioni, and Pietro Michiardi. Improved sampling schedules for discrete diffusion models. *arXiv preprint arXiv:2602.06849*, 2026.

Sunghwan Han, Youngtae Han, and Youngmin Yi. Adavla: Adaptive step flow matching for training-free acceleration of vision-language-action models. arXiv:2608.29208, 2026.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. : a vision-language-action model with open-world generalization. *0.5*: $avision - language - actionmodelwithopen - worldgeneralization$, 2025.

Yuanchang Liang, Xiaobo Wang, Kai Wang, Shuo Wang, Xiaojiang Peng, Haoyu Chen, David Kim Huat Chua, and Prahlad Vadakkepat. Adaptive action chunking at inference-time for vision-language-action models. arXiv:2604.04161, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The eleventh international conference on learning representations*, 2022.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026a.

Jijia Liu, Feng Gao, Bingwen Wei, Xinlei Chen, Qingmin Liao, Yi Wu, Chao Yu, and Yu Wang. What can rl bring to vla generalization? an empirical study. *Advances in Neural Information Processing Systems*, 38:97121–97151, 2026b.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

Tongzhou Mu, Zhan Ling, Fanbo Xiang, Derek Yang, Xuanlin Li, Stone Tao, Zhiao Huang, Zhiwei Jia, and Hao Su. Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations. *arXiv preprint arXiv:2107.14483*, 2021.

Dejan Stancevic, Florian Handke, and Luca Ambrogioni. Entropic time schedulers for generative diffusion models. *Advances in Neural Information Processing Systems*, 38:44222–44252, 2026.

Bruno Trentini, Dejan Stancevic, Michael M Bronstein, Alexander Tong, and Luca Ambrogioni. Entropy across the bridge: Conditional-marginal discretization for flow and schr\" odinger samplers. *arXiv preprint arXiv:2605.16126*, 2026.

Chao Yu, Yuanqing Wang, Zhen Guo, Hao Lin, Si Xu, Hongzhi Zang, Quanlu Zhang, Yongji Wu, Chunyang Zhu, Junhao Hu, et al. {RLinf}: Flexible and efficient {Large-Scale} reinforcement learning via {Macro-to-Micro} flow transformation. In *20th USENIX Symposium on Operating Systems Design and Implementation (OSDI 26)*, pp. 829–846, 2026.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020.