DIFFERENT FACTORIZATIONS, SAME PATH: ON-POLICY DISTILLATION FROM AUTOREGRESSIVE TO DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion language models (DLMs) have drawn increasing attention for their parallel decoding capability, but still lag behind strong autoregressive language models (ARLMs) in generation quality. This motivates the use of on-policy distillation to transfer the stronger generation capability of ARLMs to DLMs. Yet, such cross-architecture distillation is challenging because ARLMs and DLMs employ fundamentally different conditioning structures for token prediction. To bridge this mismatch, we augment the ARLM response distribution with a prescribed forward masking process to construct a teacher distribution over the same path space as the DLM decoding process. We then introduce *Path KL* to align the student and teacher path distributions, and prove that it upper-bounds the reverse KL between their final response distributions, enabling teacher evaluation of complete-response to guide the student denoising trajectories. Experiments demonstrate consistent improvement across model scales with average benchmark gains of up to 3.15 points, which become more pronounced under highly parallel decoding.

1 INTRODUCTION

Diffusion language models (DLMs) offer a promising paradigm for text generation through iterative denoising (Sahoo et al., 2024; Nie et al., 2026), enabling multiple tokens to be predicted in parallel. Despite this efficiency advantage, DLMs still lag behind strong autoregressive language models (ARLMs) in generation quality (Ye et al., 2025; Cheng et al., 2026; Bie et al., 2025). Benefiting from large-scale pretraining and extensive post-training (Yang et al., 2025; Guo et al., 2025), leading ARLMs have developed strong language modeling and reasoning capabilities, making them a natural source of supervision for improving DLMs.

To transfer these capabilities while retaining the parallel decoding mechanism of DLMs, we adopt on-policy distillation (OPD) (Agarwal et al., 2024; Gu et al., 2023), which applies teacher supervision to responses sampled from the current DLM under its existing decoding policy. Training on these student-generated responses focuses supervision on the contexts and errors encountered during actual inference, without altering the student's underlying parallel decoding policy.

However, the fundamentally different conditioning structures of ARLM and DLM make it difficult to define a suitable distillation objective as presented in Fig.1(a). At the **token** level, ARLM predictions condition on causal prefixes, whereas DLM denoising predictions condition on partially masked sequences with bidirectional context. Thus, token-level KL under these mismatched conditioning contexts does not generally recover the KL divergence between their response distributions. At the **step** level, an ARLM teacher does not define transition probabilities over the partially masked states traversed by the DLM denoising process. These mismatches motivate a common probabilistic representation that connects the ARLM response distribution with the DLM denoising process.

To construct such a representation, we turn to distributions over complete denoising trajectories, lifting the ARLM response distribution into the *path space* of the DLM decoding process, as shown in Fig. 1(b). Specifically, we augment it with an auxiliary forward masking process that starts from a complete response and progressively produces partially masked states. The masking follows a prescribed schedule compatible with the DLM decoding process, with transition probabilities determined entirely by the chosen masking rule. This augmented teacher distribution preserves the

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.

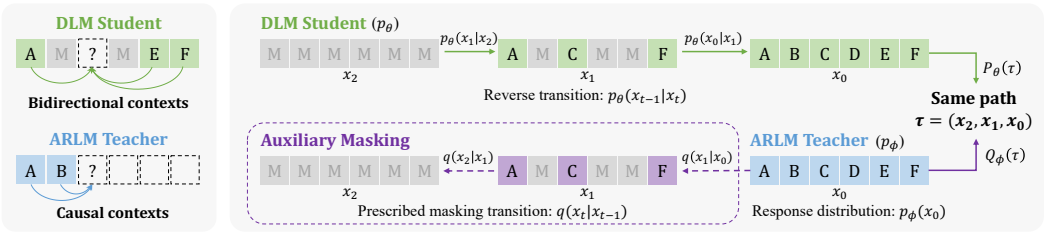


Figure 1: **The construction of a common path space for cross-architecture OPD. (a) Mismatched conditioning contexts.** The DLM student predicts masked tokens using bidirectional context, whereas the ARLM teacher conditions on causal prefixes. **(b) A common path space.** We extend the ARLM response distribution to the student's denoising trajectory space through an auxiliary forward masking process.

ARLM response distribution as its marginal, while defining probabilities over the same trajectory variables as the DLM student.

Within this common path space, we propose *Path KL*, a principled objective for on-policy distillation from ARLMs to DLMs. Defined as the reverse KL from the DLM trajectory distribution to the augmented teacher distribution, Path KL upper-bounds the corresponding divergence between their response distributions. Path KL uses ARLM evaluation of completed responses to optimize the entire denoising process. By training on trajectories sampled from the current student, our approach directly optimizes the DLM under its actual decoding policy while retaining parallel decoding.

Our contributions are threefold:

- We construct a common path-space representation for distillation from ARLMs to DLMs by augmenting the teacher response distribution with an auxiliary forward masking process, bridging their different conditioning structures.
- We introduce Path KL, an on-policy distillation objective defined on this shared path space, and prove that it upper-bounds the reverse KL between student and teacher response distributions, providing a principled objective for distillation.
- We demonstrate consistent improvements over the original models and competing baselines across student scales on math and code tasks, with average benchmark gains of up to 3.15 points over the original models. The gains become more pronounced under highly parallel decoding, reaching 7.6 points on MATH500.

2 Preliminaries

2.1 On-Policy Distillation

On-policy distillation (OPD) transfers capabilities from the teacher model p_ϕ to the student model p_θ through providing dense supervision on the responses generated by the student. For AR models conditioned on a context c, the student minimizes the divergence between the student and teacher next-token distributions using AR factorization. The objective can be written as

$$\mathcal{L}_{\text{OPD}}(\theta) = \mathbb{E}_{x\sim p_\theta(\cdot|c)}\left[\sum_{i=1}^{L} D_{\text{KL}}\big(p_\theta(\cdot \mid x^{<i}, c) \| p_\phi(\cdot \mid x^{<i}, c)\big)\right], \tag{1}$$

where $x^{<i} = (x^1, \ldots, x^{i-1})$ denotes the student-generated prefix preceding position i. Here, we present the reverse-KL formulation of OPD, as its mode-seeking behavior can help a student with limited capacity focus on the high-probability modes of the teacher.

2.2 Diffusion Language Models

In this work, we focus on masked DLMs, which generate text through iterative denoising. Given a conditioning context c, let $x_0 = (x_0^1, \ldots, x_0^L) \in \mathcal{V}^L$ denote a clean sequence of length L, where $\mathcal{V}$

is the token vocabulary. We consider an absorbing-state diffusion process with a special mask token $\mathbf{m} \notin \mathcal{V}$ and states $x_t \in (\mathcal{V} \cup \{\mathbf{m}\})^L$ for $t \in \{0, \dots, T\}$.

The forward process progressively replaces tokens in the clean sequence with $\mathbf{m}$, independently across positions, with transition probabilities:

$$q(x_t^i \mid x_{t-1}^i) = \frac{\alpha_t}{\alpha_{t-1}} \mathbb{I}\{x_t^i = x_{t-1}^i\} + \big(1 - \frac{\alpha_t}{\alpha_{t-1}}\big) \mathbb{I}\{x_t^i = \mathbf{m}\}, \tag{2}$$

where $\mathbb{I}\{\cdot\}$ is the indicator function. Here, α_t denotes the probability that a token remains unmasked at step t, following a non-increasing schedule satisfying

$$1 = \alpha_0 \geq \alpha_1 \geq \cdots \geq \alpha_{T-1} > \alpha_T = 0.$$

Once masked, a position remains masked, and all positions are masked at step T.

The reverse process starts from x_T and uses a denoiser p_θ parameterized by θ to progressively recover x_0. The denoiser is trained to reconstruct masked tokens, with $p_\theta(\cdot \mid x_t, c)$ denoting its predicted distribution over $\mathcal{V}$ at position i. We consider monotone reverse transitions, where revealed tokens remain unchanged. The masked positions at step t and the positions revealed in the transition from x_t to x_{t-1} are defined as

$$\mathcal{M}_t = \{i : x_t^i = \mathbf{m}\}, \quad \mathcal{S}_t = \mathcal{M}_t \setminus \mathcal{M}_{t-1} = \{i : x_t^i = \mathbf{m},\ x_{t-1}^i \neq \mathbf{m}\}. \tag{3}$$

Given a reveal set $\mathcal{S}_t$ deterministically selected before token sampling, the revealed tokens are sampled conditionally independently. Thus, the reverse transition probability is represented as

$$p_\theta(x_{t-1} \mid x_t, c) = \prod_{i \in \mathcal{S}_t} p_\theta(x_{t-1}^i \mid x_t, c). \tag{4}$$

3 Path-Space On Policy Distillation

We develop a path-space formulation for on-policy distillation from ARLMs to DLMs. To connect response-level distillation with student denoising trajectories, we extend the ARLM teacher distribution to a shared path space through a normalized auxiliary masking process that preserves its response marginal, as illustrated in Fig. 2. We then introduce Path KL, establish its upper-bound relation to response-level KL with an explicit gap, and present a gradient estimator using on-policy student rollouts.

3.1 From Response-Level Distillation to Path Space

Following the notation in Sec. 2, the response-level distillation objective can be defined as

$$\mathcal{L}_{\text{resp}}(\theta; c) = D_{\text{KL}}(p_\theta(x_0 \mid c) \,\|\, p_\phi(x_0 \mid c)), \tag{5}$$

where $p_\theta(x_0 \mid c)$ denotes the response distribution induced by the DLM student and $p_\phi(x_0 \mid c)$ is the ARLM teacher distribution. To connect this objective with the student's denoising process, we represent both models through distributions over complete trajectories.

Student path distribution. Let $\tau = (x_T, \dots, x_0)$ denote a denoising trajectory, starting from the fully masked state x_T and ending at the completed response x_0. Using the reverse transitions defined in Sec. 2.2, the student decoding process induces the path distribution

$$P_\theta(\tau \mid c) = \prod_{t=1}^{T} p_\theta(x_{t-1} \mid x_t, c), \tag{6}$$

and the response distribution is its marginal: $p_\theta(x_0 \mid c) = \sum_{x_{1:T}} P_\theta(\tau \mid c)$. Using the token-level factorization in Eq. (4), the log-probability of a sampled trajectory can be evaluated as

$$\log P_\theta(\tau \mid c) = \sum_{t=1}^{T} \sum_{i \in S_t} \log p_\theta(x_{t-1}^i \mid x_t, c). \tag{7}$$

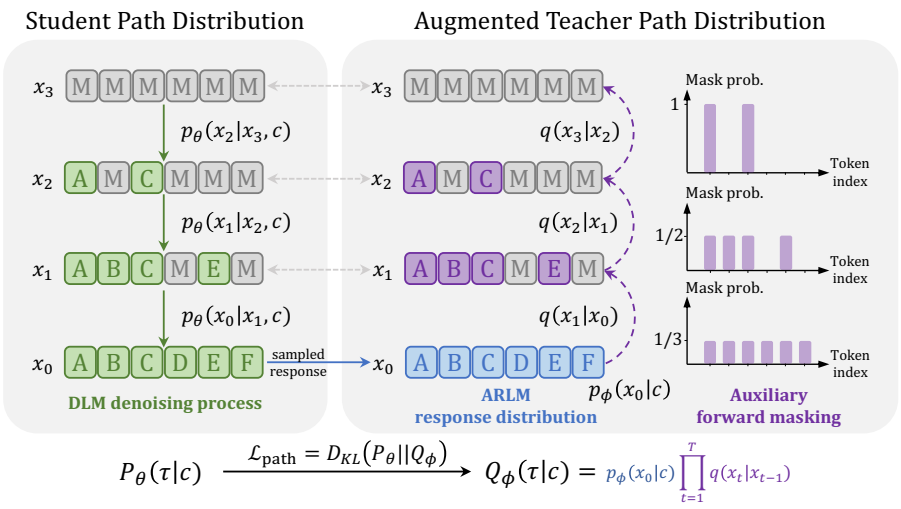


Figure 2: **Path distributions for Path KL.** The DLM student generates a denoising trajectory, whose probability under the student factorizes over its denoising transitions (left). The augmented teacher assigns a probability to the **same** trajectory by combining the ARLM response likelihood with auxiliary forward masking probabilities (right). The bars show the per-step masking probabilities for currently visible tokens under a linear schedule with $T = 3$.

Teacher path distribution. The ARLM teacher assigns probabilities to complete responses through its autoregressive factorization as follows

$$p_\phi(x_0 \mid c) = \prod_{i=1}^{L} p_\phi(x_0^i \mid x_0^{<i}, c). \tag{8}$$

To extend this ARLM response distribution to the denoising trajectory space of the DLM student, we introduce a normalized auxiliary forward masking process

$$q_{\text{fwd}}(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}). \tag{9}$$

We use the linear schedule $\alpha_t = 1 - t/T$: at step t, each currently visible token is independently masked with probability $1/(T - t + 1)$. This process ends at the fully masked state x_T and assigns non-zero probability to the masking path corresponding to each monotone student trajectory.

We then define the augmented teacher distribution as

$$Q_\phi(\tau \mid c) = p_\phi(x_0 \mid c)\, q_{\text{fwd}}(x_{1:T} \mid x_0). \tag{10}$$

This construction defines a normalized distribution over trajectories that preserves the teacher's response marginal (proof provided in Appendix B.1)

$$\sum_{x_{1:T}} Q_\phi(\tau \mid c) = p_\phi(x_0 \mid c). \tag{11}$$

Its path log-probability is the sum of the teacher's autoregressive log-likelihood of x_0 and the known auxiliary masking log-probability

$$\log Q_\phi(\tau \mid c) = \sum_{i=1}^{L} \log p_\phi(x_0^i \mid x_0^{<i}, c) + \sum_{t=1}^{T} \log q(x_t \mid x_{t-1}). \tag{12}$$

The resulting P_θ and Q_ϕ assign probabilities to the same trajectory variables, with response marginals p_θ and p_ϕ, respectively.

Algorithm 1 Path-Space On-Policy Distillation

Require: DLM student parameters θ; frozen ARLM teacher parameters ϕ; prompt dataset $\mathcal{D}$; rollout count B; learning rate η

while not converged **do**
  Sample a prompt $c \sim \mathcal{D}$
  // Generate and evaluate complete denoising paths
  for $b = 1, \dots, B$ **do**
    $\tau^{(b)} = (x_T^{(b)}, \dots, x_0^{(b)}) \sim P_\theta(\cdot \mid c)$ ▷ *Record reveal set $\mathcal{S}_t$ per reverse transition.*
    Compute $\log P_\theta(\tau^{(b)} \mid c)$ using Eq. (7)
    Compute $\log Q_\phi(\tau^{(b)} \mid c)$ using Eq. (12)
  end for
  // Estimate the Path KL gradient ▷ $\mathrm{sg}[\cdot]$ *denotes the stop-gradient operator.*
  $\nabla_\theta \widehat{\mathcal{L}}_{\text{path}} \leftarrow \frac{1}{B} \sum_{b=1}^{B} \mathrm{sg}\Big[\log P_\theta(\tau^{(b)} \mid c) - \log Q_\phi(\tau^{(b)} \mid c)\Big] \cdot \nabla_\theta \log P_\theta(\tau^{(b)} \mid c)$
  Update $\theta \leftarrow \theta - \eta \nabla_\theta \widehat{\mathcal{L}}_{\text{path}}$
end while
return θ

3.2 KL Divergence in the Path Space

Using the path distributions constructed in the previous subsection, we define Path KL as the reverse KL divergence from the student path distribution to the augmented teacher distribution

$$\mathcal{L}_{\text{path}}(\theta; c) = D_{\text{KL}}(P_\theta(\tau \mid c) \,\|\, Q_\phi(\tau \mid c)) \,. \tag{13}$$

To relate this objective to response-level distillation (Eq. (5)), we factorize the student path distribution into its response marginal and its conditional trajectory distribution

$$P_\theta(\tau \mid c) = p_\theta(x_0 \mid c)\, P_\theta(x_{1:T} \mid x_0, c). \tag{14}$$

Together with the teacher construction in Eq. (10), the definition of KL Divergence gives $\mathcal{L}_{\text{path}}(\theta; c)$ the following decomposition

$$\begin{aligned} \mathcal{L}_{\text{path}}(\theta; c) = {} & D_{\text{KL}}\left(p_\theta(x_0 \mid c) \,\|\, p_\phi(x_0 \mid c)\right) \\ & + \mathbb{E}_{x_0 \sim p_\theta(\cdot \mid c)}\left[D_{\text{KL}}\left(P_\theta(x_{1:T} \mid x_0, c) \,\|\, q_{\text{fwd}}(x_{1:T} \mid x_0)\right)\right] \end{aligned} \tag{15}$$

Since the second term is nonnegative, we have

$$D_{\text{KL}}(p_\theta(x_0 \mid c) \,\|\, p_\phi(x_0 \mid c)) = \mathcal{L}_{\text{resp}}(\theta; c) \le \mathcal{L}_{\text{path}}(\theta; c), \tag{16}$$

with the full derivation provided in Appendix B.2.

It is clear that Path KL provides an upper bound on the response-level distillation objective, and the gap is given by the expected KL divergence between the student and auxiliary masking distributions over trajectories, conditioned on the same completed response. Note that this does not require explicit computation of the student response marginal $p_\theta(x_0 \mid c)$.

3.3 Optimizing Path KL

We optimize Path KL using denoising trajectories sampled from the current student. For a context c, the objective can be written as

$$\mathcal{L}_{\text{path}}(\theta; c) = \mathbb{E}_{\tau \sim P_\theta(\cdot \mid c)}\left[\log P_\theta(\tau \mid c) - \log Q_\phi(\tau \mid c)\right]. \tag{17}$$

Substituting the student and teacher path log-probabilities from Eq. (7) and Eq. (12) gives

$$\mathcal{L}_{\text{path}} = \mathbb{E}_{\tau \sim P_\theta(\cdot \mid c)}\Big[\sum_{t=1}^{T} \sum_{i \in \mathcal{S}_t} \log p_\theta(x_{t-1}^i \mid x_t, c) - \sum_{i=1}^{L} \log p_\phi(x_0^i \mid x_0^{<i}, c) - \sum_{t=1}^{T} \log q(x_t \mid x_{t-1})\Big]. \tag{18}$$

Thus, each sampled trajectory can be evaluated using the student's denoising log-probabilities, the teacher's autoregressive log-likelihood of the completed response, and the known auxiliary masking probabilities. Note that under the linear masking schedule, the auxiliary masking term is constant across trajectories for fixed L and T and can be omitted during optimization. The derivation is provided in Appendix B.3.

We estimate Path KL by Monte Carlo averaging over on-policy student rollouts and optimize it using a score-function gradient estimator, as summarized in Algo. 1. The resulting trajectory log-ratio weights the gradient of the student path log-probability, allowing response-level teacher supervision to guide updates throughout the student's denoising process.

4 EXPERIMENTS

We evaluate the effectiveness of Path KL in transferring knowledge from autoregressive teachers to diffusion students. Our main experiments compare Path KL with baseline methods across student sizes on math and code tasks. We then examine whether the gains persist under increased inference-time parallelism, measuring both accuracy and the average number of tokens generated per step. Finally, ablation studies investigate how the representation of the student distribution affects teacher–student alignment and downstream performance, and whether Path KL remains effective when fewer denoising steps are used during training.

Table 1: Task performance across SDAR student sizes. MATH500 and GSM8K report accuracy; HumanEval and MBPP report pass@1. The OPDLM results are taken from the original paper Su et al. (2026). The teacher is included as a reference.

Model	Training method	MATH500	GSM8K	HumanEval	MBPP
Teacher: Qwen3-4B-Instruct		87.60	94.47	84.15	80.54
SDAR 1.7B	Base	62.40	80.89	62.20	57.20
	SFT	61.60	80.44	62.80	59.14
	OPDLM	53.00	71.00	35.40	45.00
	Path KL (ours)	**66.60**	**81.43**	**63.41**	**61.48**
SDAR 4B	Base	72.40	90.22	73.17	66.54
	SFT	71.40	91.13	75.61	67.32
	OPDLM	72.80	87.60	56.10	57.70
	Path KL (ours)	**75.60**	**91.58**	**79.27**	**68.48**
SDAR 8B	Base	77.20	90.67	79.88	71.98
	SFT	76.80	92.72	79.27	72.37
	OPDLM	71.20	87.10	59.80	48.70
	Path KL (ours)	**79.80**	**93.48**	**80.49**	**74.71**

4.1 EXPERIMENTAL SETUP

Models. We use Qwen3-4B-Instruct (Yang et al., 2025) as the autoregressive teacher and SDAR models (Cheng et al., 2026) with 1.7B, 4B, and 8B parameters as diffusion students. The students adopt block diffusion architecture (Arriola et al., 2025), using iterative denoising within each block.

Implementation details. We train on a mixed corpus of mathematics and code, using only the prompts for on-policy distillation and generating responses with the current student policy. We evaluate mathematical reasoning on GSM8K(Cobbe et al., 2021) and MATH500 (Hendrycks et al., 2021), and code generation on HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021), reporting final-answer accuracy and pass@1, respectively. Details of the training data and implementation are provided in Appendix C.1.

Baselines. We compare Path KL with three baselines: **Base**, **SFT**, and **OPDLM**. Base denotes the original student checkpoint before distillation. SFT fine-tunes the diffusion student on teacher-generated responses using its supervised fine-tuning objective. OPDLM Su et al. (2026) converts pretrained autoregressive models into diffusion models through on-policy distillation. It matches

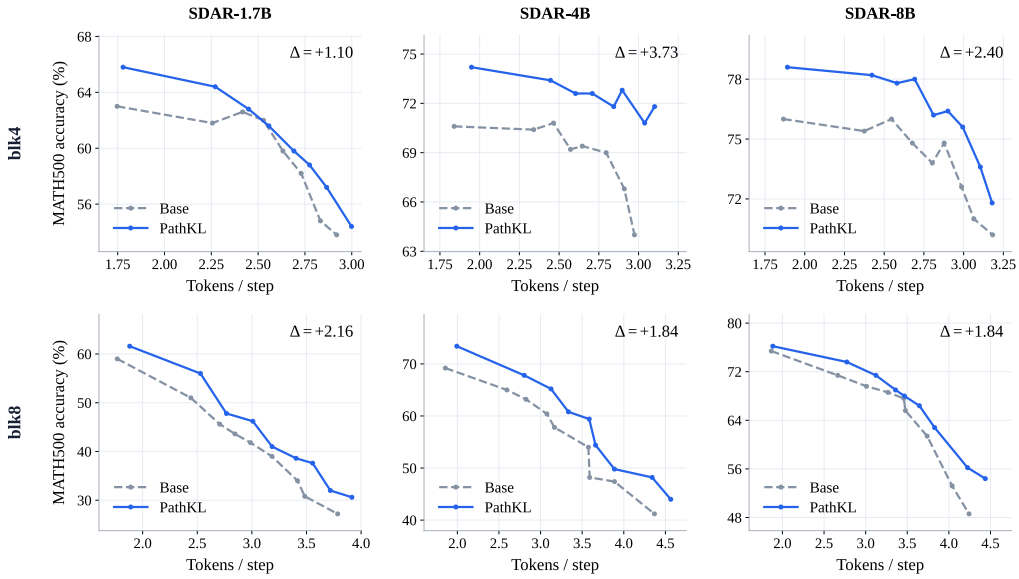


Figure 3: Accuracy–parallelism trade-off on MATH500 under confidence-based dynamic decoding. The top and bottom rows use block lengths of 4 and 8, respectively. Δ denotes the average accuracy gain over base model at matched confidence thresholds.

token distributions at intermediate states sampled from the student's decoding trajectories. We report its results directly from the original paper at matching parameter scales. For each student size, all training methods use the same evaluation configuration unless otherwise specified.

4.2 Main Results

Table 1 shows that Path KL improves diffusion students across all three model sizes, with the largest average gain observed for SDAR-4B. At this scale, Path KL improves over Base by an average of 3.15 percentage points across the four benchmarks, including a 6.10-point gain on HumanEval. The stronger gains for SDAR-4B suggest that matching the teacher parameter scale facilitates effective knowledge transfer. SDAR-1.7B also benefits from Path KL, with a slightly smaller average gain. This difference may stem from its limited model capacity. SDAR-8B, meanwhile, starts from a stronger baseline but still improves on all four benchmarks under supervision from the 4B teacher.

The comparison with SFT shows that the benefit of Path KL extends beyond access to teacher-generated responses. Although SFT provides direct examples of the teacher behavior, it does not consistently improve the initial students. This contrast supports the value of applying teacher supervision to the student's own denoising process, which is directly relevant to the student generation behavior with parallel decoding policy.

OPDLM uses on-policy supervision of intermediate decoding states for AR-to-diffusion conversion. Our SDAR models trained with Path KL outperform the published OPDLM results at matching parameter scales across all four benchmarks. These results demonstrate the effectiveness of Path KL, providing a way to further improve diffusion language models directly.

4.3 Parallel Decoding after Distillation

We examine whether the benefits of Path KL persist when inference uses greater parallelism. After distillation, we apply confidence-based dynamic decoding, unmasking all tokens whose confidence exceeds a specified threshold at each step. Varying this threshold allows us to trace the relationship between accuracy and the average number of tokens decoded per denoising step. We compare Base Model and Path KL across all three model sizes, using inference block lengths of 4 and 8.

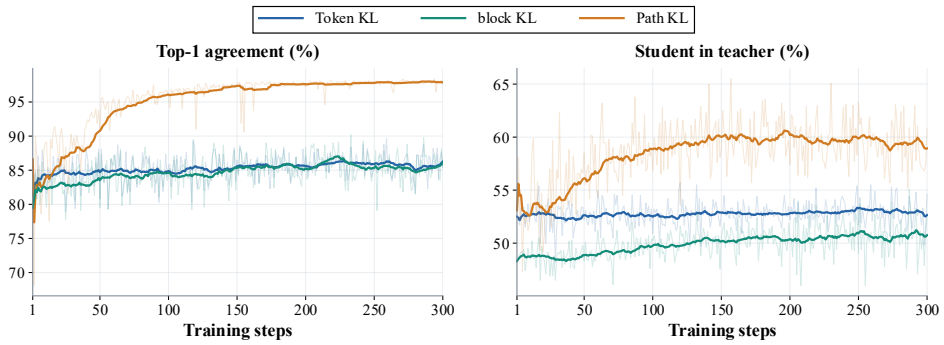


Figure 4: Teacher–student alignment during SDAR-4B training. Left: agreement between the teacher's and student's top-1 predictions. Right: fraction of student-generated tokens within the teacher's top-16 predictions.

Figure 3 shows that Path KL improves the trade-off between accuracy and decoding efficiency. The benefit is particularly clear for SDAR-4B with a block length of 4: Path KL consistently outperforms Base Model across all evaluated confidence thresholds, with an average accuracy gain of 3.73 points. The advantage becomes more pronounced at higher levels of parallelism, highlighting the robustness of Path KL as more tokens are decoded per denoising step.

Path KL also retains its advantage across all model sizes when the inference block length is increased to 8 using the same distilled models. Under highly parallel decoding, SDAR-8B achieves 56.2% accuracy with Path KL, compared with 48.6% for Base Model at approximate number of tokens per step. Overall, the retained advantage under both threshold variation and block-length expansion highlights the effectiveness of applying teacher feedback to the student's own generation process.

4.4 Ablation Studies

We conduct ablation studies on SDAR-4B to examine the choice of distillation objective and the effect of training-time parallelism. We first examine how different representations of the student distribution affect teacher–student alignment and downstream performance. We then vary the number of denoising steps during training while keeping the evaluation configuration fixed, testing whether Path KL remains effective as within-block parallelism increases.

Effect of the distillation objective. We compare three distillation objectives: Token KL, Block KL and Path KL that differ in how the student distribution is represented or approximated. The Token KL randomly masks complete responses and computes the KL divergence between the token distributions of student at masked positions and the corresponding causal token distributions of AR teacher. The Block KL compares student block log-probabilities estimated via the EBPO (Bie et al., 2026) approximation with teacher block log-probabilities computed by autoregressive factorization.

Following prior work (Li et al., 2026b), we analyze the training dynamics in Figure 4 using two metrics: top-1 agreement and the fraction of student-generated tokens within the teacher's top-16 predictions. Path KL steadily increases top-1 agreement to approximately 98%, compared with 85–86% for Token KL and Block KL. It also increases teacher top-16 coverage of generated tokens to approximately 60%, compared with 53% for Token KL and 51% for Block KL. These trends indicate stronger alignment in both the student's preferred predictions and the tokens it actually generates.

The results in Table 2 follow the same pattern: Path KL improves over Base on all four benchmarks. By comparison, the limited alignment progress under Token KL and Block KL is accompanied by no benchmark gains for the former and only modest improvements on two tasks for the latter. Together, these results support defining the student distribution over its actual decoding trajectories, allowing teacher supervision to target the generation decisions that determine its responses.

Effect of denoising steps during training. We investigate the effect of the denoising schedule during training by varying the number of denoising steps used for rollouts and student path log-

Table 2: Effect of the distillation objective on SDAR-4B. The comparison evaluates different representations of the student distribution, with Base denoting the initial student.

Objective	GSM8K	MATH500	HumanEval	MBPP
Base	90.22	72.40	73.17	66.54
Token KL	88.17	72.20	73.17	64.98
Block KL	90.90	71.60	73.78	65.37
Path KL	**91.58**	**75.60**	**79.27**	**68.48**

Table 3: Effect of training denoising steps on SDAR-4B. T_{train} denotes the denoising steps used for training rollouts and student path log-probability computation.

T_{train}	GSM8K	MATH500	HumanEval	MBPP
Base	90.22	72.40	73.17	66.54
1	90.22	74.60	77.44	67.70
2	90.83	73.20	75.00	**69.65**
4	**91.58**	**75.60**	**79.27**	68.48

probability computation among $\{1, 2, 4\}$ for SDAR-4B, while fixing the training block length to 4. During training, reducing the number of denoising steps increases within-block parallelism. As shown in Table 3, single-step training improves over Base on MATH500, HumanEval, and MBPP while maintaining the same accuracy on GSM8K. These results show that Path KL remains effective under increased training-time parallelism, even when all four tokens in a block are generated in a single denoising step.

5 Related Work

5.1 On-Policy Distillation

On-policy distillation trains students on self-generated responses using teacher supervision (Agarwal et al., 2024; Gu et al., 2023; Zhao et al., 2026b). Recent work extends this principle to continuous diffusion and flow models (Li et al., 2026a; Fang et al., 2026), as well as self-distillation in discrete diffusion language models (Dat et al., 2026; Luo et al., 2026; Zhang et al., 2026b; Zhou et al., 2026a; Ren et al., 2026). For AR-to-DLM transfer, OPDLM matches teacher token distributions conditioned on causal prefixes of completed student responses to student predictions on partially masked states (Su et al., 2026). However, matching these token distributions does not directly define a teacher distribution over the student's denoising trajectories. This motivates a common probabilistic formulation that connects the ARLM response distribution with the DLM denoising process.

5.2 Masked Diffusion Language Models

Masked Diffusion language models (DLMs) generate text by iteratively denoising a masked sequence (Li et al., 2022; Sahoo et al., 2024; Yang et al., 2026; Zhou et al., 2026b; Wu et al., 2026). To avoid the cost of pretraining from scratch, several methods transform a pretrained ARLM into a DLM by replacing causal attention with bidirectional attention and continuing training under a masked-diffusion objective (Ye et al., 2025; Cheng et al., 2026; Fu et al., 2025). Bringing OPD to this setting, however, is non-trivial, because the conditioning structures of the two models do not align (Zhao et al., 2026a; Wang et al., 2026; Ma et al., 2026; Zhang et al., 2026a). Adapting OPD to distill from an *independently trained* AR teacher into an MDLM student—thereby decoupling the teacher from the student's initialization—remains unexplored.

6 Conclusion

We presented Path KL, an on-policy distillation framework that transfers autoregressive supervision to diffusion language models through a shared path space. By augmenting the ARLM response distribution with an auxiliary forward masking process, we construct a teacher distribution over complete denoising trajectories while preserving its response marginal. The objective provides a upper bound on the reverse KL between student and teacher response distributions, allowing teacher evaluations of complete responses to supervise the student's entire denoising process. Experiments across model sizes demonstrate consistent improvements on math and code tasks, together with a better accuracy–efficiency trade-off. These findings establish Path KL as a principled and effective approach to improving DLM generation through ARLM supervision while retaining the student's parallel decoding mechanism.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, volume 2025, pp. 50726–50753, 2025.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, et al. Llada2. 0: Scaling up diffusion language models to 100b. *arXiv preprint arXiv:2512.15745*, 2025.

Tiwei Bie, Maosong Cao, Xiang Cao, Bingsen Chen, Fuyuan Chen, Kun Chen, Lun Du, Daozhuo Feng, Haibo Feng, Mingliang Gong, et al. Llada2. 1: Speeding up text diffusion via token editing. *arXiv preprint arXiv:2602.08676*, 2026.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Shuang Cheng, Yihan Bian, Dawei Liu, Yuhua Jiang, Yihao Liu, Linfeng Zhang, Qian Yao, Zhongbo Tian, Wenhai Wang, Qipeng Guo, et al. Sdar: A synergistic diffusion-autoregression paradigm for scalable sequence generation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 22058–22075, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Phuong Tuan Dat, Qi Li, and Xinchao Wang. dopsd: On-policy self-distillation for diffusion language models. *arXiv preprint arXiv:2607.04428*, 2026.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, et al. Flow-opd: On-policy distillation for flow matching models. *arXiv preprint arXiv:2605.08063*, 2026.

Yonggan Fu, Lexington Whalen, Zhifan Ye, Xin Dong, Shizhe Diao, Jingyu Liu, Chengyue Wu, Hao Zhang, Enze Xie, Song Han, et al. Efficient-dlm: From autoregressive to diffusion language models, and beyond in speed. *arXiv preprint arXiv:2512.14067*, 2025.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. Diffusionopd: A unified perspective of on-policy distillation in diffusion models. *SIGGRAPH Asia 2026 Conference Papers*, 2026a.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35: 4328–4343, 2022.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026b.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Yifu Luo, Zeyu Chen, Haoyu Wang, Xinhao Hu, Yuxuan Zhang, Zhizhou Sha, and Shiwei Liu. Learning from the self-future: On-policy self-distillation for dllms. *arXiv preprint arXiv:2606.18195*, 2026.

Tianren Ma, Mu Zhang, Yibing Wang, and Qixiang Ye. Consolidating reinforcement learning for multimodal discrete diffusion models. In *International Conference on Learning Representations*, volume 2026, pp. 147950–147971, 2026.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *Advances in Neural Information Processing Systems*, 38:50608–50646, 2026.

Haolin Ren, Ziyang Huang, Chenhao Yuan, Jun Zhao, and Kang Liu. Trace-based on-policy distillation for masked diffusion language models. *arXiv preprint arXiv:2607.16872*, 2026.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Xingyu Su, Jacob Helwig, Shubham Parashar, Atharv Chagi, Lakshmi Jotsna, Degui Zhi, James Caverlee, Dileep Kalathil, and Shuiwang Ji. Data-efficient autoregressive-to-diffusion language models via on-policy distillation, 2026.

Yinjie Wang, Ling Yang, Bowen Li, Ye Tian, Ke Shen, and Mengdi Wang. Revolutionizing reinforcement learning framework for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pp. 101476–101501, 2026.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. In *International Conference on Learning Representations*, volume 2026, pp. 57027–57051, 2026.

Zhangchen Xu, Yang Liu, Yueqin Yin, Mingyuan Zhou, and Radha Poovendran. Kodcode: A diverse, challenging, and verifiable synthetic dataset for coding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 6980–7008, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. Mmada: Multimodal large diffusion language models. *Advances in Neural Information Processing Systems*, 38:138867–138907, 2026.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

Huaye Zeng, Dongfu Jiang, Haozhe Wang, Ping Nie, Xiaotong Chen, and Wenhu Chen. Acecoder: Acing coder rl via automated test-case synthesis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12023–12040, 2025.

Mu Zhang, Tianren Ma, Yunfan Liu, Kun Hu, and Qixiang Ye. Rebrl: Reinforcing discrete visual diffusion models with rebalanced timestep credits. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 43135–43144, 2026a.

Tunyu Zhang, Xinxi Zhang, Ligong Han, Haizhou Shi, Xiaoxiao He, Zhuowei Li, Hao Wang, Kai Xu, Akash Srivastava, Chengzhi Mao, et al. Few-step diffusion language models via trajectory self-distillation. *arXiv preprint arXiv:2602.12262*, 2026b.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *Advances in Neural Information Processing Systems*, 38:56729–56762, 2026a.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026b.

Haoyang Zhou, Li Kong, Shijie Ren, Xiting Wang, Shuang Liang, Guowei Wang, and Zhenxuan Pan. Tad: Temporal-aware trajectory self-distillation for fast and accurate diffusion llm. *arXiv preprint arXiv:2605.09536*, 2026a.

Zhanhui Zhou, Lingjie Chen, Hanghang Tong, and Dawn Song. dllm: Simple diffusion language modeling. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 78–88, 2026b.

A LLM AND AI USE STATEMENT

We used generative AI tools to improve language clarity and grammar, assist revising the manuscript. We have not used generative AI tools for experimental design, or drafting scientific arguments, and figure generation is not applicable to this work. All AI-assisted content was reviewed, revised, and verified by the authors, who take full responsibility for the final content of this paper.

B DERIVATION

B.1 VALIDITY OF THE AUGMENTED TEACHER DISTRIBUTION

We show that $Q_\phi(\tau \mid c) = p_\phi(x_0 \mid c) q_{\text{fwd}}(x_{1:T} \mid x_0)$ is a normalized trajectory distribution that preserves the teacher's response marginal.

Each forward transition is nonnegative and satisfies $\sum_{x_t} q(x_t \mid x_{t-1}) = 1$. Summing successively over $x_T, \ldots, x_1$ therefore gives

$$\sum_{x_{1:T}} q_{\text{fwd}}(x_{1:T} \mid x_0) = \sum_{x_1} \cdots \sum_{x_T} \prod_{t=1}^{T} q(x_t \mid x_{t-1}) = 1. \tag{19}$$

Consequently, marginalizing the augmented teacher distribution over the intermediate and terminal states yields

$$\sum_{x_{1:T}} Q_\phi(\tau \mid c) = p_\phi(x_0 \mid c) \sum_{x_{1:T}} q_{\text{fwd}}(x_{1:T} \mid x_0) = p_\phi(x_0 \mid c), \tag{20}$$

which establishes preservation of the teacher's response marginal. Summing additionally over x_0 gives

$$\sum_{\tau} Q_\phi(\tau \mid c) = \sum_{x_0} p_\phi(x_0 \mid c) = 1. \tag{21}$$

Together with non-negativity, this proves that Q_ϕ is a valid probability distribution over trajectories.

B.2 DERIVATION OF THE PATH KL UPPER BOUND

For a fixed prompt c, let $\tau = (x_0, x_{1:T})$ denote a complete generation path. The student and augmented teacher path distributions can be factorized as

$$\begin{aligned} P_\theta(\tau \mid c) &= p_\theta(x_0 \mid c)\, P_\theta(x_{1:T} \mid x_0, c), \\ Q_\phi(\tau \mid c) &= p_\phi(x_0 \mid c)\, q_{\text{fwd}}(x_{1:T} \mid x_0). \end{aligned} \tag{22}$$

Since $q_{\text{fwd}}(\cdot \mid x_0)$ is normalized, marginalizing Q_ϕ over $x_{1:T}$ recovers the teacher's response distribution $p_\phi(x_0 \mid c)$. Assuming that $Q_\phi(\tau \mid c) > 0$ whenever $P_\theta(\tau \mid c) > 0$, substituting these factorizations into the definition of Path KL gives

$$\begin{aligned} \mathcal{L}_{\text{path}} &= \sum_{x_0, x_{1:T}} P_\theta(\tau \mid c) \log \frac{P_\theta(\tau \mid c)}{Q_\phi(\tau \mid c)} \\ &= \sum_{x_0, x_{1:T}} P_\theta(\tau \mid c) \left[\log \frac{p_\theta(x_0 \mid c)}{p_\phi(x_0 \mid c)} + \log \frac{P_\theta(x_{1:T} \mid x_0, c)}{q_{\text{fwd}}(x_{1:T} \mid x_0)} \right] \\ &= \sum_{x_0} p_\theta(x_0 \mid c) \log \frac{p_\theta(x_0 \mid c)}{p_\phi(x_0 \mid c)} \\ &\quad + \sum_{x_0} p_\theta(x_0 \mid c) \sum_{x_{1:T}} P_\theta(x_{1:T} \mid x_0, c) \log \frac{P_\theta(x_{1:T} \mid x_0, c)}{q_{\text{fwd}}(x_{1:T} \mid x_0)} \\ &= D_{\text{KL}}(p_\theta(x_0 \mid c) \,\|\, p_\phi(x_0 \mid c)) \\ &\quad + \mathbb{E}_{x_0 \sim p_\theta(\cdot \mid c)} \left[D_{\text{KL}}(P_\theta(x_{1:T} \mid x_0, c) \,\|\, q_{\text{fwd}}(x_{1:T} \mid x_0)) \right]. \end{aligned} \tag{23}$$

The third equality follows by marginalizing the first term over $x_{1:T}$ and applying the student factorization to the second term. Since the expected conditional KL divergence is nonnegative, we obtain

$$D_{\text{KL}}(p_\theta(\cdot \mid c) \,\|\, p_\phi(\cdot \mid c)) \le \mathcal{L}_{\text{path}}, \tag{24}$$

which establishes Eq. 16. The expected conditional KL divergence in Eq. 23 is exactly the gap between Path KL and the response KL. It measures the discrepancy between the student's distribution over paths leading to a given response and the prescribed forward masking distribution. For a fixed student, a closer match between these conditional distributions yields a tighter bound. For Path KL to be finite, the forward process must assign nonzero probability to every path that the student can generate, including those produced by its parallel reveal rules. This support compatibility allows the path-space objective to evaluate the student's actual parallel generation process, while the expected conditional KL quantifies the additional constraint introduced by aligning distributions over complete paths.

B.3 Constant Auxiliary Masking Term

We show that the auxiliary masking term in Eq. 18 is constant for fixed response length L and number of steps T. As defined in Sec. 3.1, the auxiliary forward process acts independently across positions under the linear retention schedule $\alpha_t = 1 - t/T$. A visible token is retained at step t with probability α_t/α_{t-1}, while a masked token remains masked.

Let $K_i \in \{1, \dots, T\}$ denote the step at which position i is first masked. Under the auxiliary forward process, the probability that this occurs at step k is

$$\begin{aligned} \Pr_q(K_i = k \mid x_0) &= \left(\prod_{s=1}^{k-1} \frac{\alpha_s}{\alpha_{s-1}}\right)\left(1 - \frac{\alpha_k}{\alpha_{k-1}}\right) \\ &= \alpha_{k-1} - \alpha_k = \frac{1}{T}. \end{aligned} \tag{25}$$

Consequently, each position is first masked at a uniformly distributed step, independently of the other positions.

Given a complete response x_0, a valid forward trajectory progressively replaces visible tokens with masks and ends at the fully masked state x_T. Each such trajectory is uniquely determined by the masking times of its L positions. Consequently, every valid trajectory has the same conditional probability:

$$q_{\text{fwd}}(x_{1:T} \mid x_0) = \prod_{i=1}^{L} \frac{1}{T} = T^{-L}. \tag{26}$$

The student denoising trajectories, read in the forward masking direction, satisfy these conditions. Their auxiliary masking contribution therefore equals

$$-\sum_{t=1}^{T} \log q(x_t \mid x_{t-1}) = -\log q_{\text{fwd}}(x_{1:T} \mid x_0) = L \log T. \tag{27}$$

Substituting this result into Eq. 18 gives

$$\mathcal{L}_{\text{path}} = \mathbb{E}_{\tau \sim P_\theta(\cdot|c)}\Big[\log P_\theta(\tau \mid c) - \log p_\phi(x_0 \mid c)\Big] + L \log T. \tag{28}$$

For fixed L and T, the auxiliary masking term is identical across all student rollouts and independent of θ. Omitting this additive constant therefore leaves the objective gradient unchanged.

Table 4: Composition of the training corpus.

Domain	Sources	Examples	Share
Mathematics	GSM8K, MATH	14,947	56.32%
Code	KodCode Light RL, AceCode	11,594	43.68%
Total		26,541	100.00%

Table 5: Default training configuration. The block length and number of denoising steps apply to both rollout generation and student path log-probability computation.

Setting	Value
Hardware	$8\times$ NVIDIA H100 GPUs
Rollout engine	LMDeploy
Optimizer	AdamW
Learning rate	5×10^{-7}
Training steps	300
Training block length	4
Training denoising steps	4
Rollout decoding	Confidence-based
Log-probability candidate-set size	$k = 16$

C Experiment Details

C.1 Training Data and Implementation Details

Training data. Our training corpus contains 26,541 examples spanning mathematics and code. As summarized in Table 4, mathematical examples are drawn from GSM8K and MATH (Lightman et al., 2024), while coding examples come from KodCode Light RL (Xu et al., 2025) and AceCode (Zeng et al., 2025). For on-policy distillation, we retain only the prompts from these datasets and generate responses using the current student policy.

Training configuration. Table 5 summarizes the default training configuration, which is used unless otherwise specified. Student path log-probabilities are computed using the same block length and denoising schedule as rollout generation. For log-probability computation, we retain the top-16 highest-scoring token candidates at each position.

C.2 Comparison with OPDLM Variants

We reproduce the forward-KL (FKL) and reverse-KL (RKL) variants of OPDLM on SDAR-4B. We follow its distillation procedure, using the existing SDAR-4B diffusion checkpoint as the student instead of initializing it from autoregressive weights.

Table 6: Comparison with OPDLM variants on SDAR-4B. GSM8K and MATH500 report accuracy; HumanEval and MBPP report pass@1. All scores are percentages (higher is better).

Method	GSM8K	MATH500	HumanEval	MBPP
Base	90.22	72.40	73.17	66.54
OPDLM (FKL)	85.31	68.40	65.85	60.13
OPDLM (RKL)	90.86	72.80	74.39	64.41
Path KL	**91.58**	**75.60**	**79.27**	**68.48**

Table 6 shows that forward KL underperforms Base on all four benchmarks, while reverse KL yields modest improvements on three tasks but reduces performance on MBPP. Path KL achieves the highest scores across all four benchmarks, exceeding the stronger reverse-KL variant by 4.88 percentage points on HumanEval and 4.07 points on MBPP. These results demonstrate the effectiveness of Path KL in further improving existing diffusion students through autoregressive teacher supervision.