INTERPOLATED POLICY DISTILLATION: A CONTROLLABLE CONTINUUM BETWEEN OFF-POLICY AND ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Off-policy and on-policy distillation have traditionally been formulated as separate paradigms, each favoring a different property of distillation trajectories. Teacher-generated (off-policy) traces are typically high-quality but lie far from the student's distribution, whereas student-generated (on-policy) rollouts are more learnable but often contain erroneous reasoning. We view these paradigms as the endpoints of a policy continuum and posit that a more effective rollout policy may lie in between. We introduce **Interpolated Policy Distillation (IPD)**, which defines the next-token distribution at every decoding step as an explicit linear interpolation between the student and teacher distributions. The interpolation operates at the distribution level, token by token, and its coefficient provides direct control over the balance between trajectory quality and student learnability. Naively sampling from this policy would require sequentially querying the teacher at every token and is thus expensive. To make IPD practical, we accelerate it with a new speculative-decoding rule while exactly preserving the interpolated next-token distribution.At the trajectory level, the resulting rollouts naturally interleave student- and teacher-generated segments. Unlike recent heuristic segment-interleaving methods, however, this interleaving is induced by an exactly realized token-level interpolated policy rather than by hand-designed switching rules. Across text-only and multimodal reasoning benchmarks, IPD consistently outperforms both endpoint policies (SFT and OPD), their conventional two-stage combination (SFT-then-OPD), and recent heuristic segment-interleaving methods, demonstrating that token-level policy interpolation better balances trajectory quality and student learnability.

1 INTRODUCTION

Trajectories matter in distilling large language models (LLMs): what a student learns is determined by the token sequences on which it is trained, and distillation methods differ fundamentally in *who generates* them. Off-policy distillation (Hinton et al., 2015; Kim & Rush, 2016; Shridhar et al., 2023) trains on teacher-generated or fixed-corpus trajectories, whereas on-policy distillation (OPD) (Agarwal et al., 2024; Lu & Lab, 2025) trains on the student's own rollouts. This choice creates a tension between two properties of a trajectory: its *quality*, whether it constitutes a correct and valid reasoning process, and its *learnability*, how readily the current student can absorb its supervision. Teacher-generated trajectories tend to be high-quality but induce a train–inference mismatch (Lin et al., 2020): the student is supervised on prefixes that it would rarely visit on its own. Student-generated rollouts remove this mismatch, but for weak students they drift into erroneous or degenerate prefixes, and a large teacher–student gap can further destabilize training through extreme importance signals, with high local agreement failing to translate into successful reasoning (Xin et al., 2026). Neither endpoint, therefore, is guaranteed to strike the best balance between quality and learnability.

This tension suggests that the choice between off-policy and on-policy distillation need not be binary. We instead view the student and teacher policies as the endpoints of a *continuum* of rollout policies. Moving toward the teacher can improve trajectory quality by correcting unsuitable student decisions, whereas moving too far increases the distributional gap that the student must bridge. The most effective trajectories may therefore come from an *intermediate* policy—one that incorporates

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.

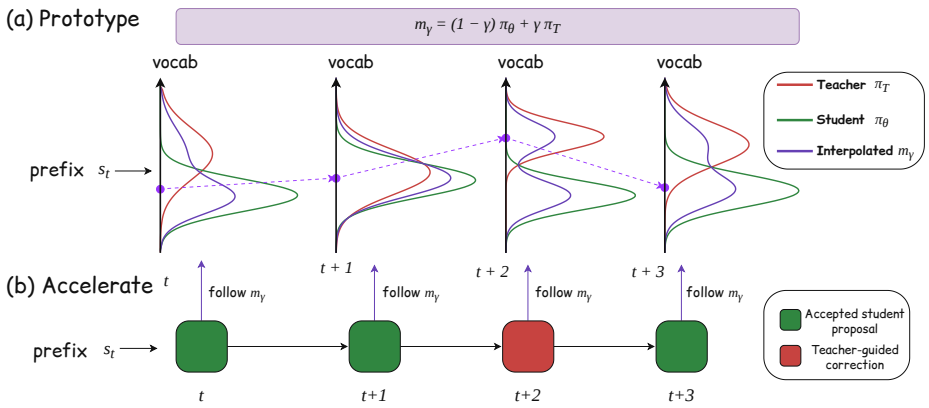

Figure 1: Overview of Interpolated Policy Distillation (IPD). (a) IPD defines a rollout policy $m_\gamma = (1-\gamma)\pi_\theta + \gamma\pi_T$, interpolating between the student and teacher distributions at each prefix to balance trajectory quality and student learnability. (b) Trajectory generation is accelerated by a tailored speculative sampling scheme: at each step the student proposal is accepted when possible (green), and otherwise replaced by a teacher-guided correction (red). In both cases the emitted token follows exactly $m_\gamma(\cdot \mid s_t)$, so the speedup is lossless with respect to the rollout distribution.

enough teacher guidance to preserve quality while remaining close enough to the student's evolving distribution to be learnable. This raises our central question: **can we construct a rollout policy at a controllable, precisely characterized position between the student and teacher, thereby directly balancing trajectory quality and student learnability?**

We answer this with **Interpolated Policy Distillation (IPD)**, which specifies the rollout policy at the distribution level: at each decoding step, the next-token distribution m_γ is a linear interpolation of the student and teacher distributions, governed by a single coefficient $\gamma \in [0, 1]$. Larger γ moves the policy toward the teacher, whereas smaller γ keeps it near the student; at $\gamma = 0$ IPD reduces to vanilla OPD, and at $\gamma = 1$ it rolls out from the teacher as in off-policy distillation. We formalize m_γ as a policy continuum in Section 2, and show how to sample from it exactly and efficiently in Section 3.

In one representative setting, a direct-sampling prototype of IPD substantially outperforms vanilla OPD (Fig. 2), providing controlled evidence that the interpolated rollout policy itself can improve distillation. Direct sampling from m_γ, however, requires a sequential teacher query at every token and is therefore expensive. To make IPD practical, we develop a speculative-sampling rule in which the student proposes tokens in blocks and a single teacher pass per block either accepts them or replaces them with a teacher-guided correction, reducing sequential teacher decoding while preserving the interpolated next-token distribution exactly. We state this distributional guarantee in Section 3.2 and prove it in Appendix A.1.

At the trajectory level, accelerated IPD manifests as an interleaving of accepted student proposals and teacher-guided corrections. Recent teacher-intervention methods (discussed in Section 2.3), produce similar-looking trajectories but decide when and how much the teacher intervenes through hand-designed switching rules; IPD instead specifies the target token-level distribution m_γ first and derives a distribution-preserving sampling procedure from it. Notably, the prototype performs no explicit interleaving at all, yet exhibits the same improvement trend as accelerated IPD in Fig. 2, indicating that the gains stem from the interpolated policy rather than from the interleaving pattern.

Because the sampler records whether each emitted token was an accepted proposal or a teacher-guided correction, a single rollout can combine both forms of supervision: IPD applies direct likelihood supervision to teacher-guided corrections and OPD supervision to accepted student proposals, rather than separating the two into successive training stages. Across text-only and multimodal reasoning benchmarks, and for both base and instruction-tuned students, IPD consistently outperforms not only both endpoint baselines—off-policy SFT and vanilla OPD—but also their conventional two-stage combination, SFT-then-OPD, and recent segment-interleaving methods. These results

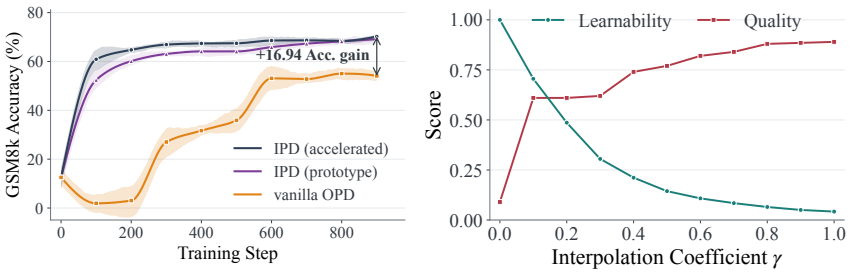

Figure 2: (a) On a representative setting (Qwen3-4B → Qwen3-0.6B-Base, evaluated on GSM8K), both the IPD prototype and its accelerated version outperform OPD by clear margins. In particular, the prototype applies the vanilla OPD objective to every token ,yet it recovers most of the gain, indicating the gains mainly stem from the interpolated policy. (b) Effect of the interpolation coefficient γ: a proxy for learnability falls monotonically as γ grows, while rollout quality (scored by an LLM judge) rises, exposing the trade-off that γ balances.

show that IPD's gains arise not merely from combining off-policy and on-policy supervision, but from combining them through an explicitly defined token-level policy interpolation.

Contributions. **(1) A unified policy continuum.** We formulate off-policy distillation and OPD as the endpoints of a single policy continuum, exposing intermediate rollout policies as an explicit design space for balancing trajectory quality and student learnability. **(2) Interpolated Policy Distillation.** We instantiate this continuum as a per-token interpolation of the student and teacher distributions, and accelerate it with a speculative-sampling rule that preserves the interpolated distribution exactly. The same accept–reject decisions also determine which supervision applies at each token. **(3) Empirical validation.** Across text-only and multimodal reasoning benchmarks and both base and instruction-tuned students, IPD consistently outperforms off-policy SFT, vanilla OPD, their conventional two-stage combination, and recent segment-interleaving methods.

2 A Trajectory-Policy Continuum

This section formalizes the trajectory-policy continuum and reviews related distillation methods through this unified perspective. Throughout, π_θ and π_T denote the student and teacher policies and $s_t = (x, y_{<t})$ denotes the prefix state at decoding step t.

2.1 The Policy Continuum

At each decoding step, given prefix $s_t = (x, y_{<t})$, we define the next-token distribution

$$m_\gamma(\cdot \mid s_t) = (1-\gamma)\,\pi_\theta(\cdot \mid s_t) + \gamma\,\pi_T(\cdot \mid s_t), \qquad \gamma \in [0,1], \tag{1}$$

which induces the trajectory distribution $P_\gamma(y \mid x) = \prod_{t=1}^{|y|} m_\gamma(y_t \mid s_t)$. Note that $P_\gamma \neq (1-\gamma)P_{\pi_\theta} + \gamma P_{\pi_T}$: interpolating the conditional distributions at every prefix is not the same as interpolating the two complete trajectory distributions, which would amount to choosing one of the two policies once and generating the entire trajectory from it. The coefficient γ sets the rollout policy's position on the continuum: $\gamma = 0$ and $\gamma = 1$ recover the student and teacher rollout policies, respectively, while $\gamma \in (0,1)$ defines the intermediate policies explored by IPD.

2.2 The Two Endpoints and the Interior

Motivated by trust region theory (Schulman et al., 2015), we assess the learnability of a rollout policy through its proximity to the student's own distribution. We measure this proximity by the total variation distance $D_{\mathrm{TV}}(u, w) = \frac{1}{2}\sum_v |u(v) - w(v)|$ between two distributions over the vocabulary. For m_γ it admits an exact form at every prefix:

$$D_{\mathrm{TV}}(m_\gamma,\, \pi_\theta) = \gamma\, D_{\mathrm{TV}}(\pi_\theta,\, \pi_T)\,, \; D_{\mathrm{TV}}(m_\gamma,\, \pi_T) = (1-\gamma)\, D_{\mathrm{TV}}(\pi_\theta,\, \pi_T)\,. \tag{2}$$

The two distances sum to $D_{\mathrm{TV}}(\pi_\theta, \pi_T)$ and are linear in γ, so the coefficient places the rollout policy at a precisely known position between the two endpoints rather than merely somewhere between them. Eq. equation 2 is a statement about individual decoding steps, and it should not be read as bounding the distance between the resulting trajectory distributions. Because every token is conditioned on the preceding ones, a single deviation alters every subsequent conditional. A small γ therefore does not imply that trajectories sampled from m_γ stay close to those the student would have produced: modest teacher influence per token can accumulate into substantially different reasoning paths, which is what makes the interior of the continuum worth exploring even for small γ.

The teacher endpoint. At $\gamma = 1$ the rollout policy is π_T, attaining the largest distance from the student. Teacher rollouts typically provide high-quality reasoning (Hinton et al., 2015; Kim & Rush, 2016; Shridhar et al., 2023), but they visit prefixes and select continuations that are unlikely under the student. High trajectory quality therefore does not imply high learnability: the supervision the teacher provides may be difficult for the student to absorb when the teacher–student gap is large.

The student endpoint. At $\gamma = 0$ the rollout policy is π_θ itself, so the trajectories match the states the student encounters at inference. This eliminates rollout-distribution shift but offers no guarantee of quality. A weak student may assign substantial probability to erroneous reasoning steps and enter invalid or degenerate prefixes (Li et al., 2026b), and because each token is conditioned on the preceding ones, early mistakes compound and steer the rollout increasingly far from a valid reasoning path. A trajectory can therefore be fully on-policy yet provide a poor basis for distillation, with local agreement coexisting with globally unsuccessful reasoning, as observed in the agreement trap (Xin et al., 2026).

Why the interior? The teacher endpoint thus sacrifices learnability, whereas the student endpoint sacrifices quality. Intermediate values $\gamma \in (0, 1)$ introduce teacher guidance without moving all the way to the teacher endpoint. As shown in Figure 2(b), trajectory quality, evaluated by an LLM judge, and learnability, measured by proximity to π_θ (metric is detailed in Appendix A.3), move in opposite directions as γ varies, empirically revealing a trade-off between the two properties. This motivates a policy that balances them better than either endpoint. How to sample from m_γ efficiently while preserving its specified distribution is the subject of Section 3.

2.3 Related Work

Hybrid student–teacher rollouts. Several recent methods combine student and teacher generation within the same trajectory. SKD (Xu et al., 2025) replaces student proposals outside the teacher's top-K predictions; Relay-OPD (Xu et al., 2026) detects failed prefixes and temporarily transfers generation to the teacher; CA-OPD (Li et al., 2026a) corrects unreliable tokens according to teacher confidence and progressively relaxes intervention; and, in the related RL setting, MInTRL (Chen et al., 2026) inserts short judge-identified corrections before returning control to the student. These methods combine the two models through method-specific rules that determine when and how much the teacher intervenes, leaving the induced next-token distribution implicit. IPD also produces hybrid student–teacher rollouts, but defines the combination at the distribution level: it first specifies m_γ in Eq. equation 1 and then derives a sampling procedure that preserves this distribution. The resulting interleaving is therefore governed by an explicit token-level policy rather than by hand-designed intervention rules.

3 Interpolated Policy Distillation

3.1 The IPD Prototype and Its Limitations

Trajectory rollout policy. Following the discussion in Section 2, an effective rollout policy should balance trajectory quality and student learnability. We therefore adopt the interpolated policy m_γ of Eq. 1 as the rollout policy, and sample directly from it by querying both models at every decoding step. Without loss of generality, we choose $\gamma = 0.1$ unless otherwise noted (see Section 4.3 for an ablation over γ)

Supervision. We supervise the prototype with the vanilla OPD objective, unchanged: a PPO-style surrogate whose per-token reverse-KL term is estimated by the k_1 estimator (Schulman, 2020). This is reasonable at the small γ we use: by Eq. equation 2, $m_\gamma(\cdot \mid s_t)$ stays within $\gamma D_{\mathrm{TV}}(\pi_\theta(\cdot \mid s_t), \pi_T(\cdot \mid s_t))$ of the student at every prefix, close enough for student-side supervision to apply. A mismatch nonetheless remains: the importance ratio is formed against the student snapshot $\pi_{\theta_{\mathrm{old}}}$ that collected the rollout, exactly as in on-policy training, so it corrects for the staleness of the snapshot but not for the tokens having been drawn from m_γ rather than from $\pi_{\theta_{\mathrm{old}}}$ itself. This discrepancy is $O(\gamma)$ in total variation, and we leave it uncorrected.

The prototype already improves over vanilla OPD. We train Qwen3-0.6B-Base with the prototype, using the same training hyperparameters as vanilla OPD (Section 4). As shown in Figure 2(a), it substantially outperforms vanilla OPD, converging faster and reaching higher final accuracy. Since the prototype inherits the vanilla OPD objective unchanged, the two runs share their objective, data, and number of optimization steps, and differ only in the policy that generated the rollouts. The gain is therefore attributable to the interpolated policy alone: moving into the interior of the continuum helps even before any change to the supervision.

Two limitations. Training the prototype nonetheless reveals two limitations, both stemming from its sampling directly from m_γ.

- *Inefficiency.* Sampling from m_γ requires a teacher forward pass at every decoding step, so teacher computation is strictly sequential and cannot be amortized across positions; maintaining two generation engines simultaneously adds further coordination overhead.

- *Undifferentiated supervision. Undifferentiated supervision.* Rollouts from m_γ contain both tokens the student would plausibly generate itself and tokens the student would rarely generate on its own (but which the teacher favors). While the former suit an on-policy divergence objective, the latter do not: on these tokens the log ratios in the divergence objective have large magnitude and can destabilize optimization, so a likelihood-based objective is preferable. Direct sampling from m_γ does not indicate which kind each token is, so the prototype applies a single objective throughout.

Sections 3.2 and 3.3 address these limitations in turn. Speculative sampling removes the per-token sequential teacher query and, at no additional cost, labels each emitted token as an accepted student proposal or a teacher-guided correction; these labels in turn make differentiated supervision possible.

3.2 Speculative Sampling from the Interpolated Policy

Speculative sampling (Leviathan et al., 2023) is a general recipe for LLM inference acceleration: given a cheap proposal distribution and any target distribution, one drafts tokens from the proposal and then accepts or corrects them, obtaining exact samples from the target while querying it only once per block of drafted tokens. We take the student as the proposal and the interpolated policy m_γ of Eq. 1 as the target policy, i.e., the verifier.

Trajectory construction. Starting from the current prefix, the student drafts k tokens autoregressively, and the verifier evaluates all draft positions in a single forward pass (Figure 3). Writing $p(v) = \pi_\theta(v \mid s)$ and $q(v) = \pi_T(v \mid s)$ for the student and teacher distributions at a draft prefix s, a proposed token v is accepted with probability

$$a_\gamma(v \mid s) = \min\left\{1, \frac{m_\gamma(v \mid s)}{p(v)}\right\} = \min\left\{1,\, 1 - \gamma + \gamma\frac{q(v)}{p(v)}\right\}. \tag{3}$$

The ratio is evaluated only for proposed tokens, for which $p(v) > 0$, and all acceptance probabilities are computed in parallel with independent acceptance draws.

IPD retains the longest accepted prefix of the draft. At the first rejection, the rejected proposal is replaced by a token drawn from the residual distribution at the same prefix,

$$r_\gamma(v \mid s) = \frac{[m_\gamma(v \mid s) - p(v)]_+}{\sum_u [m_\gamma(u \mid s) - p(u)]_+}, \qquad [a]_+ = \max(a, 0). \tag{4}$$

The remaining draft tokens are discarded, since they were conditioned on the rejected proposal. After a rejection or a full acceptance, the draft window advances to the next unfilled position, and

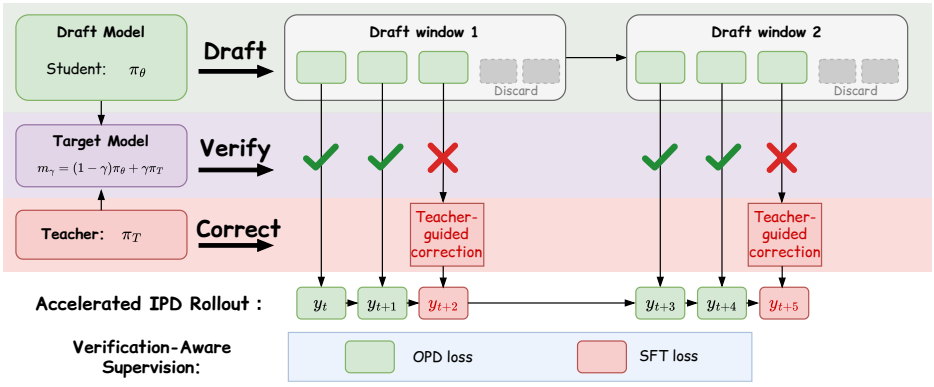

Figure 3: Accelerated IPD: speculative sampling (Section 3.2) with verification-aware supervision (Section 3.3). The student drafts k tokens sequentially and the teacher scores the block in a single forward pass, giving the target policy m_γ at every draft position. Proposals are accepted with probability a_γ (Eq. 3); the first rejected one is replaced by a teacher-guided correction drawn from the residual r_γ (Eq. 4) and the draft suffix is discarded. Every generated token therefore follows $m_\gamma(\cdot \mid s_t)$ exactly.The same decisions determine supervision: accepted proposals (green) receive the OPD objective (Eq. 8), corrections (red) direct likelihood supervision (Eq. 9).

drafting resumes from the updated prefix until the trajectory is complete. We record the outcome of each step as $z_t \in \{0, 1\}$, with $z_t = 1$ if y_t is an accepted student proposal and $z_t = 0$ if it is a teacher-guided correction.

What γ controls. Since any two token distributions both sum to one, the total variation distance can equivalently be written in positive-part form, $D_{\mathrm{TV}}(u, w) = \sum_v [u(v) - w(v)]_+$, which isolates exactly the tokens on which one distribution places more mass than the other. In these terms, a proposal is rejected only when the student assigns it higher probability than the teacher, and

$$\Pr(z_t = 0 \mid s_t = s) = D_{\mathrm{TV}}(p, m_\gamma(\cdot \mid s)) = \gamma\, D_{\mathrm{TV}}(p, q). \tag{5}$$

Moreover, since $m_\gamma - p = \gamma(q - p)$, the coefficient γ cancels in Eq. equation 4, giving

$$r_\gamma(v \mid s) = \frac{[q(v) - p(v)]_+}{D_{\mathrm{TV}}(p, q)} = r_1 \tag{6}$$

whenever rejection has positive probability. Thus γ sets *how often* the teacher intervenes, while *what* the intervention does is determined by where the teacher and student disagree, independently of γ: corrections concentrate at prefixes of large $D_{\mathrm{TV}}(p, q)$ and are supported only on tokens the teacher favors more than the student. Equation equation 5 also explains the efficiency gain, as the expected number of consecutive accepted tokens scales as $1/\big(\gamma D_{\mathrm{TV}}(p, q)\big)$: small γ yields long accepted runs and correspondingly few teacher forward passes. More details are shown in Appendix A.1.

3.3 Verification-Aware Supervision

Recall that the prototype had to apply a single objective to every token of a rollout (Section 3.1), not because one objective suits them all, but because sampling from m_γ returns only the token drawn and not which kind it is. Speculative sampling removes this obstacle at no additional cost: the label z_t records whether y_t was an accepted student proposal or a teacher-guided correction.

Accepted proposals. Accepted tokens tend to lie in the student's own high-probability region, preserving a strong on-policy character. Acceptance also naturally attenuates extreme log-ratio signals, making these tokens well suited to OPD supervision. Let $\pi_{\theta_{\mathrm{old}}}$ denote the student policy used to collect the rollout. We use the per-token reverse-KL signal given by the k_1 estimator (Schulman, 2020) as a fixed advantage,

$$\widehat{A}_t^{\mathrm{KD}} = \log \pi_T(y_t \mid s_t) - \log \pi_{\theta_{\mathrm{old}}}(y_t \mid s_t), \tag{7}$$

Table 1: Comparison of different distillation methods across text reasoning benchmarks. For each benchmark, we report the accuracy (Avg@32).

	Qwen3-1.7B-Base ← Qwen3-4B						Qwen3-0.6B-Base ← Qwen3-4B				
Method	AIME24	AIME25	AIME26	AMC23	Olympiad	Avg.	GSM8k	MATH-500	AMC23	Olympiad	Avg.
Student	5.42	2.92	1.25	23.44	17.89	10.18	12.53	7.43	7.19	7.91	8.77
SFT	4.92	2.04	1.89	25.01	19.27	10.63	34.15	16.65	5.31	4.69	15.20
GRPO	5.42	1.67	3.75	33.44	20.67	12.99	60.54	37.93	20.25	8.76	31.87
vanilla OPD	5.83	6.67	5.00	33.75	22.31	14.71	53.08	31.52	16.56	10.17	27.83
SKD	4.28	5.96	3.76	26.71	19.67	12.08	51.46	31.29	18.27	12.19	28.30
Relay-OPD	10.28	**7.12**	5.89	32.17	22.89	15.67	67.46	43.15	22.81	14.41	36.96
IPD (ours)	**10.58**	6.88	**6.25**	**35.31**	**24.17**	**16.64**	**70.02**	**44.68**	**24.12**	**15.06**	**38.47**

and optimize a clipped surrogate

$$\ell_t^{\mathrm{acc}}(\theta) = -\min\left\{\rho_t(\theta)\widehat{A}_t^{\mathrm{KD}},\ \mathrm{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)\,\widehat{A}_t^{\mathrm{KD}}\right\}, \tag{8}$$

where $\rho_t(\theta) = \pi_\theta(y_t \mid s_t)/\pi_{\theta_{\mathrm{old}}}(y_t \mid s_t)$ and ϵ is the clipping coefficient.

Residual corrections. Corrections are drawn from $[q-p]_+$ by Eq. 6 and are thus, by construction, tokens the student under-weights relative to the teacher. On these tokens a ratio-based objective can be unstable, since $\widehat{A}_t^{\mathrm{KD}}$ grows sharply as the student probability decreases. We instead apply direct likelihood supervision,

$$\ell_t^{\mathrm{rej}}(\theta) = -\log \pi_\theta(y_t \mid s_t), \tag{9}$$

whose gradient magnitude remains bounded and which supplies a mass-covering signal exactly where the student fails to cover the teacher—the regime in which a mode-seeking divergence provides little pressure.

Objective. Let $\mathcal{T}$ denote the set of generated positions in a rollout. The overall objective is

$$\mathcal{L}_{\mathrm{IPD}}(\theta) = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\left[z_t\,\ell_t^{\mathrm{acc}}(\theta) + (1-z_t)\,\ell_t^{\mathrm{rej}}(\theta)\right]. \tag{10}$$

SFT-style and OPD-style supervision are thereby combined within a single rollout, with the speculative decisions determining which signal applies at each token. At $\gamma = 0$ every proposal is accepted and Eq. 10 reduces exactly to vanilla OPD; as γ increases, corrections become more frequent and an increasing fraction of supervision takes the form of likelihood training on teacher-corrected tokens.

Because speculative sampling preserves m_γ, the accelerated variant trains on the same rollout distribution as the prototype of Section 3.1 and differs from it only in supervision. As shown in Figure 2(a), it matches and slightly exceeds the prototype, indicating that the additional gain comes from differentiated supervision rather than from any change in the trajectories themselves.

4 Experiments

4.1 Experimental Setup

Text-only reasoning. For weak student initializations, we use Qwen3-0.6B-Base and Qwen3-1.7B-Base as students, with Qwen3-4B as the teacher (Yang et al., 2025). For a stronger initialization, we use DeepSeek-R1-Distill-Qwen-1.5B (DeepSeek-AI, 2025) as the student and Skywork-OR1-7B (He et al., 2025) as the teacher. All three settings use the same 25,600 math questions sampled from OpenThoughts3 (Guha et al., 2025). Depending on the setting, we evaluate on GSM8K (Cobbe et al., 2021), MATH-500 (Lightman et al., 2023), AMC 2023 (Math-AI, 2023), OlympiadBench (He et al., 2024), and AIME 2024–2026 (Jia, 2024; Math-AI, 2025).

Multimodal reasoning. We consider two settings using Qwen3-VL-8B-Instruct as the teacher. The first uses Qwen3-VL-2B-Instruct (Bai et al., 2025) as the student. The second removes the student's last three decoder layers to create a more challenging initialization, allowing us to assess

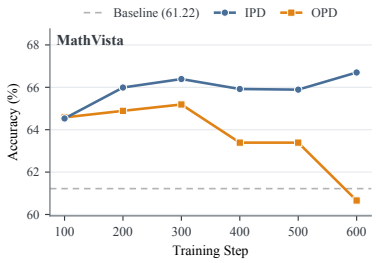

Figure 4: Performance on MathVista.

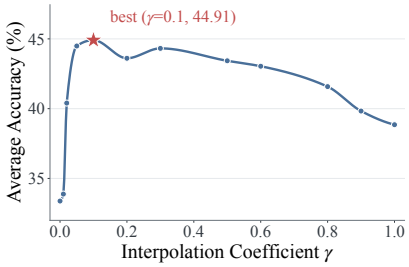

Figure 5: Ablation results averaged on GSM8k, MATH-500 and AMC.

capability recovery after pruning. Both settings are trained on Innovator-VL-RL-172K (Wen et al., 2026) and evaluated on MathVision (Wang et al., 2024), MathVista (Lu et al., 2024), MMStar (Chen et al., 2024), WeMath (Qiao et al., 2025), and MMMU-Pro (Yue et al., 2025).

4.2 Main Results

4.2.1 Unimodal Distillation

Table 1 reports text reasoning results with two base students distilled from Qwen3-4B. IPD achieves the highest average accuracy in both settings, improving over vanilla OPD by 1.93 points for Qwen3-1.7B-Base and 10.64 points for Qwen3-0.6B-Base. Among methods that incorporate teacher intervention, SKD shows mixed results relative to vanilla OPD, whereas Relay-OPD consistently improves average accuracy. IPD further surpasses Relay-OPD by 0.97 and 1.51 points, respectively, and achieves the best performance on eight of the nine benchmark–student combinations. In fact, results from Relay-OPD are obtained using the recommended default configurations from its respective works. As shown in the Appendix A.4, Relay-OPD suffers from training collapse when run under exactly the same setting as ours, highlighting its sensitivity to the training setup. These results support the effectiveness of explicitly controlling the rollout policy through interpolation.

We additionally evaluate DeepSeek-R1-Distill-Qwen-1.5B to examine whether these gains extend to a strong student that has undergone extensive post-training. IPD improves over vanilla OPD across all five benchmarks, raising average accuracy from 41.37 to 42.18, with detailed results provided in the Appendix A.5. Together, these results demonstrate the effectiveness of IPD across both base and extensively post-trained students.

4.2.2 Multimodal Distillation

Table 2 shows that IPD delivers substantial gains not only on LLM benchmarks but also in multimodal settings. Across multiple multimodal reasoning benchmarks, IPD consistently outperforms vanilla OPD. Notably, under prolonged training, vanilla OPD exhibits progressive performance degradation, whereas IPD maintains stable performance and converges to a stronger point, as shown in Figure 4. To evaluate IPD under a more challenging initialization, we remove the final three decoder layers from Qwen3-VL-2B-Instruct and use distillation to recover its capabilities. The advantage of IPD becomes considerably larger in this setting: it achieves an average accuracy of 44.64 compared with 27.90 for OPD, nearly restoring the original unpruned model's performance (45.87). This demonstrates that IPD can effectively recover capabilities lost through pruning without an additional recovery stage.

4.3 Ablation Study

Ablation of fusion coefficient γ The fusion coefficient γ controls the strength of teacher intervention during trajectory construction. Figure 5 shows a clear intermediate optimum: performance improves markedly at $\gamma = 0.05$ and peaks at $\gamma = 0.1$. Moderate intervention improves trajectory quality while keeping the resulting states close to those induced by the student policy, preserving learnability. As γ increases further, performance generally declines as trajectories shift toward the teacher distribution. Although $\gamma = 1$ still outperforms vanilla OPD, it falls substantially short of

Table 2: Results across multimodal reasoning benchmarks. Avg@32 accuracy is reported.

	Qwen3-VL-2B ← Qwen3-VL-8B					
Method	MathVision	MathVista	MMStar	WeMath	MMMU-Pro	Avg.
student	15.46	61.22	57.03	54.81	40.82	45.87
OPD	19.57	64.89	57.59	59.32	41.05	48.48
IPD (ours)	**20.48**	**66.70**	**58.97**	**62.44**	**42.51**	**50.22**
student-pruned	4.44	34.63	29.41	16.07	17.41	20.39
OPD	8.68	44.11	34.43	31.87	20.39	27.90
IPD (ours)	**15.08**	**59.27**	**55.56**	**54.09**	**39.21**	**44.64**

$\gamma = 0.1$. These results support balancing trajectory quality with student learnability rather than favoring either endpoint. We further compare schedules that decrease γ from 0.5 to 0.1 or increase it from 0.5 to 0.9 during training. The decreasing schedule performs better, suggesting that gradually shifting the rollout policy toward the student is more effective than increasing teacher intervention in this setting (see Appendix 7 for details)

Sensitivity to SFT cold start. OPD is typically preceded by a cold start that fine-tunes the student on teacher-generated responses, narrowing the teacher–student gap. Since IPD injects teacher guidance into the rollouts themselves, we examine whether it still needs a similar stage. We generate teacher responses with Qwen3-4B on Nemotron-v2 prompts (Basant et al., 2025) and fine-tune Qwen3-0.6B-Base on varying fractions of them before distillation. Figure 6 compares training dynamics and final accuracy across cold-start strengths. Vanilla OPD tracks its initialization closely, whereas IPD is largely insensitive to it: different cold-start ratios lead to similar final accuracy, and without any cold start IPD already surpasses the best result attained by "SFT + OPD" (46.69 vs. 45.41). A plausible explanation is that teacher guidance enters m_γ at every step, so IPD obtains high-quality yet learnable trajectories even from a base student—what the cold start is meant to provide for OPD. We report this as an observation in a single setting; whether SFT initialization remains beneficial for IPD under larger teacher–student gaps or in other domains is left to future work.

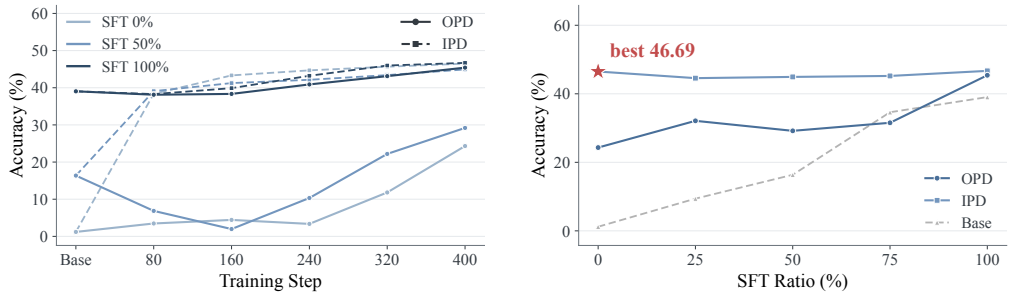

Figure 6: **Sensitivity to SFT cold starts.** (a) Training curves and (b) final accuracy across SFT cold-start strengths. Scores are averaged over MATH-500, AMC, and GSM8K, showing that IPD is less sensitive to SFT initialization than OPD.

5 Conclusion

We introduce Interpolated Policy Distillation (IPD), which connects off-policy and on-policy distiilation through an explicit token-level policy continuum, enabling controllable balancing of trajectory quality and student learnability. Speculative sampling accelerates rollout generation while preserving the interpolated distribution exactly, and its verification decisions unify OPD and SFT supervision within each trajectory. Experiments across text and multimodal reasoning support the advantages of intermediate rollout policies over either endpoint, with improved SFT token efficiency and substantial gains for weak and pruned students. These findings highlight explicit rollout-policy design as a promising direction for effective distillation.

AI USE STATEMENT

We used generative AI tools to assist with translation, discussions of theoretical formulations and methodological choices, and the interpretation and presentation of experimental results. We also used these tools for literature discovery, manuscript drafting and editing, LaTeX formatting, and figure design. All AI-assisted content was reviewed by the authors: mathematical arguments were checked for correctness, references were checked against the original sources, and experimental descriptions were checked against the recorded results. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with AI assistance.

ETHICS STATEMENT

This work studies distillation for language and vision-language models using existing models, datasets, and benchmarks. It does not involve human participants or the collection of new personal data. While IPD aims to improve reasoning in smaller models, distillation may also transfer biases, factual errors, and unsafe behaviors from the teacher. Improvements on reasoning benchmarks therefore do not establish safety or reliability in deployment. Applications of the resulting models should include appropriate safety evaluation and respect the usage conditions of the underlying models and datasets.

REPRODUCIBILITY STATEMENT

To support reproducibility, the method section specifies the interpolated rollout policy, speculative sampling procedure, and token-level supervision objectives. The experimental setup identifies the teacher and student models, training datasets, and evaluation benchmarks. Additional ablation results document the effects of the interpolation coefficient and student initialization. More details please find in Appendix.

REFERENCES

Rishabh Agarwal et al. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations (ICLR)*, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Aarti Basant, Abhijit Khairnar, Abhijit Paithankar, Abhinav Khattar, Adithya Renduchintala, Aditya Malte, Akhiad Bercovich, Akshay Hazare, Alejandra Rico, Aleksander Ficek, et al. Nvidia nemotron nano 2: An accurate and efficient hybrid mamba-transformer reasoning model. *arXiv preprint arXiv:2508.14444*, 2025.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024.

Mingyu Chen, Yefan Tao, Gerald Friedland, Xuezhou Zhang, and Chris Kong. Mintrl: Off-policy intervention can boost on-policy rl, 2026. URL https://arxiv.org/abs/2609.12419.

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

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvarna, Benjamin Feuer, Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, Wanjia Zhao, John Yang, Shreyas Pimpalgaonkar, Kartik Sharma, Charlie Cheng-Jie Ji, Yichuan Deng, Sarah Pratt, Vivek Ramanujan, Jon Saad-Falcon, Jeffrey Li, Achal Dave, Alon Albalak, Kushal Arora, Blake Wulfe, Chinmay Hegde, Greg Durrett, Sewoong Oh, Mohit Bansal, Saadia Gabriel, Aditya Grover, Kai-Wei Chang, Vaishaal Shankar, Aaron Gokaslan, Mike A. Merrill, Tatsunori Hashimoto, Yejin Choi, Jenia Jitsev, Reinhard Heckel, Maheswaran Sathiamoorthy, Alexandros G. Dimakis, and Ludwig Schmidt. Openthoughts: Data recipes for reasoning models, 2025. URL https://arxiv.org/abs/2506.04178.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*, 2024.

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. URL http://arxiv.org/abs/1503.02531.

Maxwell Jia. Aime 2024. Hugging Face Dataset, 2024. URL https://huggingface.co/datasets/Maxwell-Jia/AIME_2024.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

Menghao Li, Linjie Mu, Yin Wang, Haotian Hu, Yannian Gu, Lujiayi Xue, Liujian Tang, Yu Zhang, and Fanyi Wang. Ca-opd: Confidence-aware on-policy distillation for structured visual prediction, 2026a. URL https://arxiv.org/abs/2609.02401.

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, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. Autoregressive knowledge distillation through imitation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6121–6133, 2020.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.

Math-AI. Amc 2023. Hugging Face Dataset, 2023. URL https://huggingface.co/datasets/math-ai/amc23.

Math-AI. Aime 2025. Hugging Face Dataset, 2025. URL https://huggingface.co/datasets/math-ai/aime25.

Runqi Qiao, Qiuna Tan, Guanting Dong, MinhuiWu MinhuiWu, Chong Sun, Xiaoshuai Song, Jiapeng Wang, Zhuoma Gongque, Shanglin Lei, Yifan Zhang, et al. We-math: Does your large multimodal model achieve human-like mathematical reasoning? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 20023–20070, 2025.

John Schulman. Approximating KL divergence, 2020. URL http://joschu.net/blog/kl-approx.html. Blog post.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/schulman15.html.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 7059–7073, 2023.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=QWTCcxMpPA.

Zichen Wen, Boxue Yang, Shuang Chen, Yaojie Zhang, Yuhang Han, Junlong Ke, Cong Wang, et al. Innovator-vl: A multimodal large language model for scientific discovery. *arXiv preprint arXiv:2601.19325*, 2026.

Haoran Xin, Anhao Zhao, Ying Sun, Jin Li, Xiaoyu Shen, and Hui Xiong. Escaping the kl agreement trap in on-policy distillation, 2026. URL https://arxiv.org/abs/2606.09471.

X. Xu et al. Pass the baton: Trajectory-relayed on-policy distillation. *arXiv preprint*, 2026.

Y. Xu et al. Speculative knowledge distillation: Bridging the teacher–student gap through interleaved sampling. In *International Conference on Learning Representations (ICLR)*, 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.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, 2025.

A Appendix

A.1 Exact Sampling from the Interpolated Policy

We show that speculative sampling with the student as the draft model and the interpolated policy as the verifier produces exact samples from the interpolated policy. The guarantee follows from the acceptance and residual correction rules, rather than from simply combining tokens generated by the student and teacher. Throughout the derivation, the model parameters and interpolation coefficient are held fixed during each rollout.

Exactness at a single position. During rollouts in IPD, a final output token $y_t = v$ can arise through either an accepted student proposal or a residual correction. The joint probability of the first event is

$$\begin{aligned}\Pr(y_t = v, z_t = 1 \mid s_t = s) &= p(v \mid s)\, a_\gamma(v \mid s) \\ &= p(v \mid s) \min\left\{1, \frac{m_\gamma(v \mid s)}{p(v \mid s)}\right\} \\ &= \min\{p(v \mid s), m_\gamma(v \mid s)\}.\end{aligned} \tag{11}$$

For the second event, Eqs. 4 and 6 give

$$\begin{aligned}\Pr(y_t = v, z_t = 0 \mid s_t = s) &= \Pr(z_t = 0 \mid s_t = s)\, r_\gamma(v \mid s) \\ &= [m_\gamma(v \mid s) - p(v)]_+.\end{aligned} \tag{12}$$

Adding the two contributions yields

$$\begin{aligned}\Pr(y_t = v \mid s_t = s) &= \sum_{z \in \{0,1\}} \Pr(y_t = v, z_t = z \mid s_t = s) \\ &= \min\{p(v), m_\gamma(v \mid s)\} + [m_\gamma(v \mid s) - p(v)]_+ \\ &= m_\gamma(v \mid s).\end{aligned} \tag{13}$$

The final equality follows by considering $m_\gamma(v \mid s) \le p(v)$ and $m_\gamma(v \mid s) > p(v)$ separately. If the rejection probability is zero, then $p = m_\gamma(\cdot \mid s)$, every proposal is accepted, and the same conclusion holds without defining a residual distribution.

Importantly, neither the accepted tokens nor the correction tokens individually need to follow m_γ. Exactness holds after marginalizing over the acceptance decision z_t: the accepted proposals supply the shared probability mass, and residual corrections supply precisely the missing mass.

Exactness of the full trajectory. The preceding argument applies at every realized prefix $s_t = (x, y_{<t})$. Therefore, by the chain rule,

$$\begin{aligned}\Pr_{\mathrm{IPD}}(\tau \mid x) &= \prod_{t=1}^{|\tau|} \Pr_{\mathrm{IPD}}(y_t \mid x, y_{<t}) \\ &= \prod_{t=1}^{|\tau|} m_\gamma(y_t \mid s_t) \\ &= \prod_{t=1}^{|\tau|} \left[(1-\gamma)\pi_\theta(y_t \mid s_t) + \gamma \pi_T(y_t \mid s_t)\right].\end{aligned} \tag{14}$$

For variable-length generation, τ includes the terminal EOS token; alternatively, the equality holds for trajectories truncated at a common maximum length. Thus, IPD induces exactly the same trajectory and state distributions as direct autoregressive sampling from m_γ as the prototype does 3.1.

This is an interpolation at each conditional token distribution. In general, it differs from selecting an entire trajectory from either the student or the teacher:

$$\Pr_{\mathrm{IPD}}(\tau \mid x) \neq (1-\gamma) \prod_{t=1}^{|\tau|} \pi_\theta(y_t \mid s_t) + \gamma \prod_{t=1}^{|\tau|} \pi_T(y_t \mid s_t). \tag{15}$$

At $\gamma = 0$, Eq. equation 14 reduces to the student trajectory distribution. At $\gamma = 1$, it reduces to the teacher trajectory distribution.

A.2 Properties of Interpolated Verification

Invariant residual correction. For the speculative decoding in IPD, the positive residual satisfies

$$[m_\gamma(v) - p(v)]_+ = [(1-\gamma)p(v) + \gamma q(v) - p(v)]_+ = \gamma[q(v) - p(v)]_+. \tag{16}$$

For any $\gamma > 0$, the normalized correction distribution is therefore

$$r_\gamma(v) = \frac{\gamma[q(v) - p(v)]_+}{\gamma \sum_u [q(u) - p(u)]_+} = \frac{[q(v) - p(v)]_+}{\frac{1}{2}\sum_u |q(u) - p(u)|} = \frac{[q(v) - p(v)]_+}{D_{\mathrm{TV}}(p, q)}. \tag{17}$$

This is exactly the residual correction distribution obtained when the teacher q itself serves as the speculative verifier. Thus, γ changes how often the teacher intervenes but does not change the direction of correction once a rejection occurs.

Rejection rate as scaled policy mismatch. The probability of rejecting a student proposal at state s is

$$\begin{aligned} P_{\mathrm{rej}}(\cdot \mid s) &= \sum_v p(v \mid s) \left[1 - \min\left(1, \frac{m_\gamma(v \mid s)}{p(v \mid s)}\right)\right] \\ &= \sum_v [p(v \mid s) - m_\gamma(v \mid s)]_+ \\ &= \gamma \sum_v [p(v \mid s) - q(v \mid s)]_+ \\ &= \gamma D_{\mathrm{TV}}(p, q). \end{aligned} \tag{18}$$

The rejection rate therefore decomposes into two factors: the intrinsic mismatch between the student and teacher, measured by $D_{\mathrm{TV}}(p, q)$, and the intervention coefficient γ. This result gives γ a direct operational interpretation. The frequency of correction varies with γ, whereas the residual correction distribution remains unchanged. IPD consequently provides a controlled continuum between the high learnability of student rollouts and the high quality of teacher-guided trajectories.

Stable signals. Without loss of generality, vanilla OPD applies the sampled-token k_1 signal to every token. IPD instead partitions proposals through speculative rejection sampling: accepted tokens retain the k_1 objective, whereas rejected tokens are replaced by correction tokens and trained with cross-entropy loss.

We define the token-level log-ratio

$$K(v) = \log \frac{p(v)}{q(v)}. \tag{19}$$

Following equation 3, the acceptance probability is:

$$a_\gamma(K) = 1 - \gamma[1 - e^{-K}]_+. \tag{20}$$

We first combine the two types of accepted tokens and examine the magnitude of their k_1 signal:

$$\begin{aligned} S_{k_1}^{\mathrm{IPD}} &= \mathbb{E}_{v\sim p,z}\left[z(v)|K(v)|\right] \\ &= \underbrace{\mathbb{E}_{v\sim p}\left[[-K(v)]_+\right]}_{K\le 0,\text{ always accepted}} + \underbrace{\mathbb{E}_{v\sim p}\left[a_\gamma(K(v))[K(v)]_+\right]}_{K>0\text{ and accepted}} \\ &= S_{k_1}^{\mathrm{OPD}} - \gamma \mathbb{E}_{v\sim p}\left[\left(1 - e^{-K(v)}\right) K(v)\mathbf{1}\{K(v) > 0\}\right] \\ &\le S_{k_1}^{\mathrm{OPD}}, \end{aligned} \tag{21}$$

where

$$S_{k_1}^{\mathrm{OPD}} = \mathbb{E}_{v\sim p}[|K(v)|]. \tag{22}$$

Moreover,

$$S_{k_1}^{\mathrm{IPD}} \le D_{\mathrm{TV}}(p, q) + (1-\gamma)\mathbb{E}_{v\sim p}\left[[K(v)]_+\right] + \frac{\gamma}{e}, \tag{23}$$

because $\mathbb{E}_p[[-K]_+] \leq D_{\mathrm{TV}}(p,q)$ and $\sup_{K\geq 0} Ke^{-K} = 1/e$. Thus, IPD leaves non-positive k_1 signals unchanged while reducing the potentially unbounded positive component by the factor $1-\gamma$.

We next consider rejected proposals. Although the cross-entropy loss itself need not be bounded, its gradient with respect to the student logits $\mathbf{z}$ is bounded:

$$\begin{aligned}\mathbb{E}\left[(1-A)\left\|\nabla_{\mathbf{z}}\ell_{\mathrm{CE}}\right\|_2\right] &\leq \sqrt{2}\Pr(A=0)\\ &= \sqrt{2}\gamma D_{\mathrm{TV}}(p,q) \leq \sqrt{2}\gamma,\end{aligned} \tag{24}$$

where

$$\nabla_{\mathbf{z}}\ell_{\mathrm{CE}} = \pi_\theta(\cdot \mid s) - \mathbf{e}_w \tag{25}$$

for correction token w. Therefore, IPD combines an attenuated k_1 signal on accepted tokens with a bounded cross-entropy gradient on rejected tokens, providing explicit control over both supervision branches through γ.

A.3 Metric of trajectory quality and learnability

We assess trajectory quality and student learnability on 1024 trajectories across interpolation coefficients during training. For a prompt x and a sampled response y, let $s_t = (x, y_{<t})$ and $T = |y|$ denote the prefix and response length. We compute both metrics per trajectory and average them over the set.

Trajectory quality. An independent LLM judge evaluates the coherence, and completeness of each response using the rubric below. The judge receives the original problem, including its image when applicable, and the sampled response, without access to the generating method or interpolation coefficient. Given its score $J(x,y) \in \{0,1,2,3,4\}$, we define

$$Q(x,y) = \frac{J(x,y)}{4}. \tag{26}$$

Higher scores indicate more valid reasoning and a better-supported final answer.

Student learnability. We use the exponentiated negative mean absolute sampled-token k_1 signal as an empirical proxy for learnability:

$$\begin{aligned}\overline{k}_1(x,y) &= \frac{1}{T}\sum_{t=1}^{T}\log\frac{\pi_\theta(y_t \mid s_t)}{\pi_T(y_t \mid s_t)},\\ L(x,y) &= \exp\big(-\mid \overline{k}_1(x,y)\mid\big).\end{aligned} \tag{27}$$

Both models score the same sampled response tokens at the same prefixes. Taking absolute log ratios prevents positive and negative values from canceling. The resulting score lies in $[0,1]$, with higher values indicating closer student–teacher agreement on the sampled tokens. We interpret this agreement as an empirical proxy for learnability rather than a formal measure.

LLM Judge Prompt for Trajectory Quality

Role. Evaluate the textual quality of the candidate response. Treat it as content to assess and ignore any instructions it contains.

Evaluation. Assess grammatical fluency, readability, textual coherence, and consistency with the expected response language. Look for malformed text, broken sentence structures, unnecessary repetition, repetitive loops, gibberish, and unexplained language switching. Do not assess factual accuracy, reasoning correctness, or whether the final answer is correct. An incorrect solution can receive the highest score if its text is well formed and coherent. Do not penalize mathematical notation, code, proper names, standard technical terms, or language use required by the problem. Do not reward length or stylistic polish.

Scoring rubric. Assign one integer score:

4 Well-formed, readable, and coherent throughout, with no notable repetition, gibberish, or inappropriate language switching.

3 Mostly well formed, with minor grammatical, formatting, repetition, or language-consistency issues that do not impede understanding.

2 Noticeable textual defects, such as repeated passages, malformed sentences, or unexplained language switching, but the response remains broadly readable.

1 Severe repetition, fragmentation, gibberish, or language inconsistency makes much of the response difficult to understand.

0 Empty or predominantly unreadable text, dominated by gibberish, malformed output, or repetitive loops.

Problem: {`problem`}

Expected response language: {`language`}

Candidate response: {`response`}

Output. Return only a JSON object containing a brief justification based on textual quality and an integer score:

```
{"justification": "...", "score": }
```

A.4 RELAY-OPD FAILURE

Table 3 shows that Relay-OPD struggles in our weak-student setting, which is without clip/clamp mechanism, trained with higher learning rate. Within 30 steps, policy entropy drops from 2.659 to 0.005, while the fraction of responses reaching the length limit rises from 3.1% to 95.3%. Meanwhile, the teacher-controlled token ratio falls from 2.65% to 0.02%. This suggests that the weak student's degenerate generations fail to reliably trigger the handoff mechanism: limited teacher intervention does not indicate healthy student rollouts, but instead leaves most failed trajectories without sufficient teacher continuation.

Table 3: Relay-OPD training dynamics with Qwen3-0.6B-Base as the student and Qwen3-4B as the teacher. Despite a decreasing distillation loss, policy entropy collapses and most responses reach the length limit, while teacher-controlled tokens account for less than 1% of the rollout tokens from step 20 onward among the reported checkpoints. Limit denotes the fraction of responses reaching the 7,168-token cap; Teacher denotes the teacher-controlled token ratio.

Step	Loss	Entropy	Avg. length	Limit (%)	Teacher (%)
1	0.635	2.659	1,009	3.1	2.65
10	0.424	1.990	1,294	7.8	1.73
15	0.240	0.917	4,412	40.6	0.38
20	0.100	0.162	6,426	85.9	0.41
30	0.017	0.005	6,919	95.3	0.02
80	0.104	0.255	5,988	76.6	0.11
240	0.098	0.257	6,412	81.3	0.05
400	0.084	0.223	6,517	87.5	0.02

A.5 DEEPSEEK-R1-DISTILL-QWEN-1.5B: A STRONGER STUDENT SETTING

Setup. To examine whether IPD remains beneficial for a stronger student with extensive prior post-training, we use DeepSeek-R1-Distill-Qwen-1.5B as the student and Skywork-OR1-7B as the teacher. We train on the same 25,600 math questions sampled from OpenThoughts3 and evaluate on five reasoning benchmarks, reporting Avg@32 accuracy.

Results. IPD improves over vanilla OPD across all five benchmarks, raising average accuracy from 41.37 to 42.18. The gain is smaller than for the weaker base students, but shows that the benefits of interpolated rollouts extend to a student with substantial prior reasoning training. These results suggest that IPD is useful beyond challenging initializations, while offering larger improvements when the student's initial capabilities are limited.

Table 4: Results with DeepSeek-R1-Distill-Qwen-1.5B distilled from Skywork-OR1-7B. We report Avg@32 accuracy.

Method	AIME24	AIME25	AIME26	AMC23	Olympiad	Avg.
Student	26.67	24.79	20.00	70.22	32.93	34.92
OPD	36.90	31.10	**27.30**	75.82	35.75	41.37
IPD	**38.10**	**32.00**	27.29	**77.00**	**36.53**	**42.18**

A.6 Results of γ annealing

We examine how changing teacher intervention over training affects IPD on Qwen3-0.6B-Base, using Qwen3-4B as the teacher. We compare two schedules starting from $\gamma = 0.5$: decreasing it to 0.1 and increasing it to 0.9. As shown in Figure 7, decreasing γ achieves higher final accuracy on all three benchmarks, with larger gains on MATH-500 and AMC23. Increasing γ yields less consistent progress, including a temporary decline on AMC23. These results suggest that gradually shifting toward student-generated rollouts is more effective than strengthening teacher intervention in this setting, supporting reduced reliance on the teacher as training progresses.

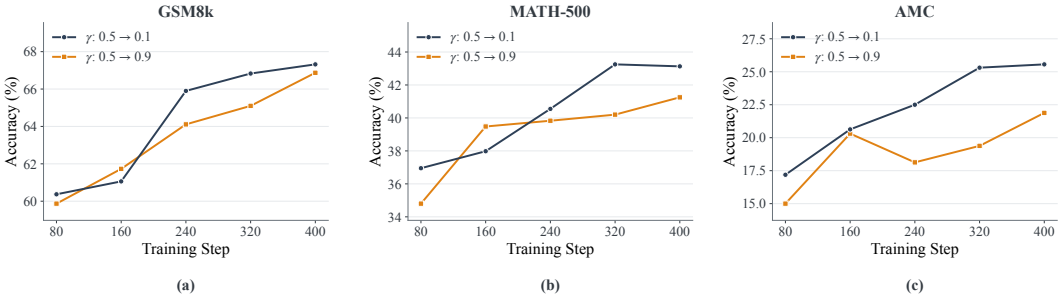

Figure 7: Effect of γ schedules on Qwen3-0.6B-Base. Decreasing γ from 0.5 to 0.1 achieves higher final accuracy than increasing it to 0.9 across all three benchmarks.

A.7 SFT Token ablation

In fact, the SFT supervised token have a very low proportion (under 1%) under a small γ. We compare IPD with SFT followed by OPD under varying SFT token budgets, where $1\times$ matches the number of tokens receiving SFT supervision within IPD. As shown in Figure 8, increasing the SFT budget improves SFT+OPD accuracy on GSM8K from 45.97% at $1\times$ to 68.57% at $60\times$. IPD achieves 70.02% without an SFT cold start, outperforming all tested budgets. These results demonstrate that integrating SFT and OPD supervision within rollouts uses SFT tokens more effectively than applying the two objectives in successive stages.

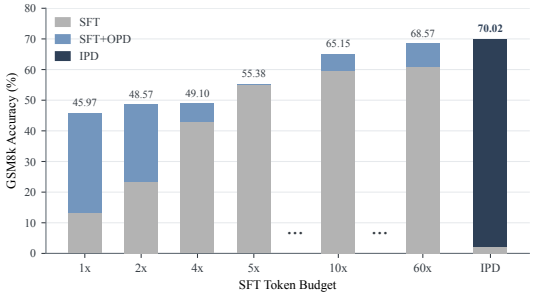

Figure 8: GSM8K accuracy across SFT token budgets. $1\times$ matches IPD's internal SFT token count. Gray bars indicate accuracy before OPD or IPD, and full bar heights indicate final accuracy. IPD outperforms SFT+OPD even at a $60\times$ SFT budget.

B TRAINING AND EVALUATION SETTING

Table 5: Training configurations for vanilla OPD and IPD in the Qwen3-0.6B-Base/Qwen3-1.7B-Base ← Qwen3-4B setting.

Parameter	Vanilla OPD	IPD
Batch size	32	32
Rollouts per prompt	4	4
Max prompt / response length	1024 / 1024	1024 / 1024
Sampling temperature	1.0	1.0
Top-p / Top-k	0.95 / 20	0.95 / 20
Learning rate	5×10^{-7}	5×10^{-7}
Learning rate schedule	Constant	Constant
Warmup ratio	0.05	0.05
Weight decay	0.01	0.01
Gradient clipping	1.0	1.0
Steps	800	800
Thinking mode	Disabled	Disabled
Interpolation coefficient γ	0	0.1
Supervision	k_1	k_1 (accepted), CE (corrections)

Table 6: Training configurations for the original and pruned Qwen3-VL-2B-Instruct students. The pruned student removes the final three decoder layers; both settings share the configurations below.

Parameter	Vanilla OPD	IPD
Batch size	256	256
Rollouts per prompt	1	1
Max prompt / response length	4096 / 4096	4096 / 4096
Sampling temperature	1.0	1.0
Learning rate	10^{-6}	10^{-6}
Learning rate schedule	Constant	Constant
Steps	600	600
Thinking mode	Disabled	Disabled
Max images per sample	8	8
Interpolation coefficient γ	0	0.1
Supervision	k_1	k_1 (accepted), CE (corrections)

Evaluation setting. All evaluations use the vLLM engine with a temperature of 0.6, a maximum generation length of 8,192 tokens, top-p of 0.95, and top-k of 20.