WAYPOINT-OPD: TRANSFERRING RL-INDUCED POLICY SHIFTS VIA GATED ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Policy-shift distillation reuses the difference between a teacher's log probabilities before and after reinforcement learning (RL) as supervision on student-generated trajectories. However, this endpoint policy shift cannot reveal whether local action preferences evolved consistently during teacher training. The same endpoint shift can result from early and late shifts that reinforce or partially cancel each other. We investigate whether agreement between early and late teacher policy shifts can guide selective transfer. We introduce **Waypoint-guided On-Policy Distillation** (Waypoint-OPD), which uses an intermediate teacher checkpoint from the same RL run to compare early and late shifts in relative action preferences at student-visited prefixes. We gate the endpoint shift according to the directional agreement between the early and late shifts, suppressing it when they point in opposing directions. Compared with endpoint-only transfer, our method requires only one additional frozen-teacher scoring pass. Using RL-trained Qwen3-4B teachers, we evaluate our method with Qwen3-8B, Qwen3-14B, and Qwen3-30B-A3B as student models in mathematical and code reasoning. In all six domain–student settings, Waypoint-OPD achieves the highest domain-average score among the compared distillation methods. Ablations with the Qwen3-8B student show that removing or shuffling gates reduces mathematical reasoning performance. These results support the use of directional agreement between early and late teacher policy shifts to weight the endpoint shift during student training. Our code is available at `https://anonymous.4open.science/r/Waypoint-OPD`.

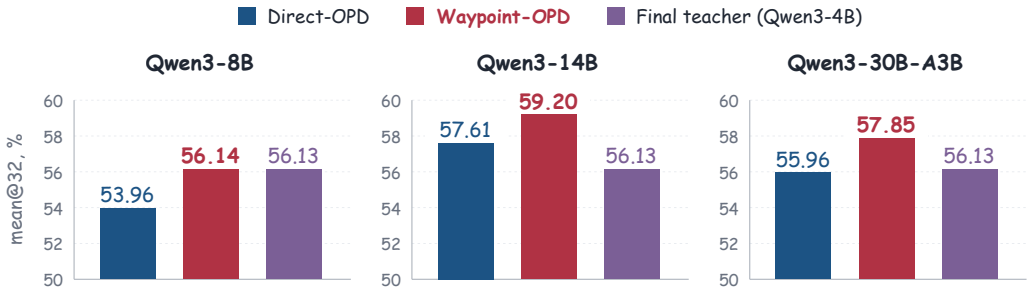


Figure 1: **Mathematical reasoning across three student models.** `mean@32` accuracy (%), averaged over seven benchmarks. Waypoint-OPD improves over Direct-OPD by 2.18, 1.59, and 1.89 percentage points, respectively. Teacher scores are from the Final Qwen3-4B checkpoint.

1 INTRODUCTION

Reinforcement learning can improve language-model instruction following using rewards learned from human preferences (Ouyang et al., 2022), and mathematical and code reasoning using verifiable rewards (Shao et al., 2024; Guo et al., 2025; Yu et al., 2025). The latter requires repeated generation, verification, and optimization of on-policy responses. Reusing a teacher's RL improvements offers a way to share this training effort across target models. This raises the question: which local changes in the teacher's policy should guide a student on its own generated trajectories?

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.

Standard on-policy distillation (OPD) trains the student to match the teacher's token distributions on student-generated prefixes (Agarwal et al., 2024). Its supervision remains tied to the teacher's absolute policy and capability limitations. On-Policy Delta Distillation (OPD^2) uses the change between a reasoning-tuned teacher and its base model, retaining token-level signals when the centered change agrees in sign with the centered standard distillation signal (Heo et al., 2026). Direct On-Policy Distillation (Direct-OPD) focuses on RL-induced improvements. It uses the log-probability ratio between pre-RL and post-RL teacher checkpoints as a dense implicit reward on student-generated states, transferring the teacher's policy change without matching its absolute distribution (Feng et al., 2026). Both methods extract changes from a pair of teacher models, but neither directly measures whether consecutive changes during teacher RL point in compatible directions.

The Initial-to-Final log-probability ratio captures only the change between the teacher's policies before and after RL training. The same endpoint signal can arise from early and late policy changes that point in similar or opposing directions. Endpoint-only transfer cannot distinguish these cases. We hypothesize that agreement between the early and late changes provides evidence for deciding how much weight to assign an endpoint shift at a student-generated prefix.

Building on this hypothesis, we propose Waypoint-OPD, which uses agreement between consecutive teacher shifts to gate the endpoint transfer signal. We retain three checkpoints from the same teacher RL run: Initial before RL, Waypoint at an intermediate RL step, and Final at the end of RL. At each student-generated prefix, we select its top-k most probable next-token candidates. Over these candidates, we compare how the teacher's relative preferences change from Initial to Waypoint and from Waypoint to Final. Their agreement gates the centered Initial-to-Final signal, retaining aligned changes and suppressing reversals. Waypoint thus supplies evidence for weighting the endpoint shift, which remains the transferred signal. Relative to endpoint-only transfer, this requires one additional frozen teacher scoring pass, with no new teacher training or verifier calls during student transfer.

Our contributions are:

- **Temporal agreement as a criterion for selective transfer.** We use the Waypoint checkpoint to assess temporal agreement at student-generated prefixes, revealing directional information unavailable from the endpoint pair alone.
- **Waypoint-guided gating of endpoint signals.** We develop Waypoint-OPD, a dense on-policy objective that uses this agreement to weight the endpoint signal at each student-generated prefix, with one additional frozen teacher scoring pass.
- **Higher average performance across domains and model sizes.** We evaluate Qwen3-8B, Qwen3-14B, and Qwen3-30B-A3B (Yang et al., 2025) on mathematics and code. Waypoint-OPD achieves the highest recorded domain average among the evaluated baselines in all six settings (Figure 1 and Tables 1 and 2). Ablations on Qwen3-8B mathematical reasoning support the value of temporal gating (Table 3).

2 Preliminaries

2.1 Problem setup and notation

Let $x \sim \mathcal{D}$ be a prompt, $y = (y_1, \ldots, y_T)$ a response, and $s_t = (x, y_{<t})$ the prefix at which a candidate token a is scored. The student π_θ generates on-policy responses $y \sim \pi_\theta(\cdot \mid x)$. Its initialization π_S remains fixed as a reference during transfer.

We denote the policies of the Initial, Waypoint, and Final teachers by π_0, π_m, and π_1, respectively. All three checkpoints come from the same teacher RL run: Initial before RL, Waypoint at an intermediate step, and Final after RL. They remain frozen during transfer. We measure the teacher's RL-induced change relative to π_0 and regularize student updates relative to student initialization π_S. Teachers and student share a tokenizer, allowing direct comparison of probabilities for the same token.

2.2 On-Policy Distillation

On-policy distillation (OPD) samples responses from the student and aligns its next-token distribution with the teacher's at the prefixes the student visits (Agarwal et al., 2024). Using the Final teacher π_1,

OPD minimizes the reverse-KL objective

$$\mathcal{L}_{\text{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)}\left[\sum_{t=1}^{T} D_{\text{KL}}(\pi_\theta(\cdot \mid s_t) \,\|\, \pi_1(\cdot \mid s_t))\right]. \tag{1}$$

2.3 Direct On-Policy Distillation

Direct-OPD uses the teacher's RL-induced policy shift as an implicit reward on student-generated responses (Feng et al., 2026). The Initial-to-Final shift decomposes into token-level log-probability ratios:

$$\Delta^{(0,1)}(y \mid x) = \log \frac{\pi_1(y \mid x)}{\pi_0(y \mid x)} = \sum_{t=1}^{T} \delta_t(y_t), \tag{2}$$

where

$$\delta_t(a) = \log \pi_1(a \mid s_t) - \log \pi_0(a \mid s_t). \tag{3}$$

A positive shift indicates that teacher RL increased a token's probability at s_t; a negative shift indicates a decrease. At an idealized teacher optimum with KL regularization toward π_0, the sequence log-ratio recovers the reward up to positive scale and a prompt-dependent constant (Rafailov et al., 2023) as used by Feng et al. (2026). This motivates the endpoint signal; we do not assume that finite-step teacher RL attains this optimum.

The idealized sequence-level objective maximizes this shift while regularizing the student toward its own initialization:

$$\mathcal{J}_{\text{Direct-OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\Big[\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[\Delta^{(0,1)}(y \mid x)] - \beta D_{\text{KL}}(\pi_\theta(\cdot \mid x) \,\|\, \pi_S(\cdot \mid x))\Big], \tag{4}$$

where $\beta > 0$ controls the student KL penalty. In practice, Direct-OPD uses $\delta_t(a)$ as an immediate token-level reward at student-generated prefixes, without verifier queries.

3 Method

Waypoint-OPD augments the endpoint signal in Section 2.3 with the Waypoint checkpoint π_m from the same teacher RL run. Agreement between the early and late shifts weights the endpoint signal on student-visited prefixes. Waypoint supplies consistency evidence rather than an imitation target. Figure 2 shows the pipeline.

3.1 Student-local trajectory consistency

At a student-visited prefix s_t, $\mathcal{K}_t = \text{TopK}(\pi_\theta(\cdot \mid s_t), k)$ selects the k token candidates with the highest student probabilities. The normalized student distribution on this support is

$$q_t(a) = \frac{\pi_\theta(a \mid s_t)}{\sum_{b\in\mathcal{K}_t} \pi_\theta(b \mid s_t)}, \qquad a \in \mathcal{K}_t. \tag{5}$$

These weights focus the comparison on likely student actions. Consistency is evaluated on the local support that receives the student update, with more probable candidates receiving greater weight. We score these candidates with all three frozen teachers. The two consecutive teacher shifts are

$$\delta_t^{(1)}(a) = \log \pi_m(a \mid s_t) - \log \pi_0(a \mid s_t), \tag{6}$$

$$\delta_t^{(2)}(a) = \log \pi_1(a \mid s_t) - \log \pi_m(a \mid s_t), \tag{7}$$

and telescope to $\delta_t = \delta_t^{(1)} + \delta_t^{(2)}$.

A uniform shift across candidates leaves their relative preferences unchanged, yet can affect the angle between raw shift vectors. We remove this action-independent component with the q_t-centering operator

$$C_{q_t}[z](a) = z(a) - \mathbb{E}_{b\sim q_t}[z(b)] \tag{8}$$

and write $u_t = C_{q_t}[\delta_t^{(1)}]$, $v_t = C_{q_t}[\delta_t^{(2)}]$, and $d_t = C_{q_t}[\delta_t] = u_t + v_t$. We weight candidates by student probability using the inner product $\langle u_t, v_t\rangle_{q_t} = \sum_{a\in\mathcal{K}_t} q_t(a)u_t(a)v_t(a)$. The local temporal agreement and weighted norms are

$$c_t = \frac{\langle u_t, v_t\rangle_{q_t}}{\|u_t\|_{q_t}\|v_t\|_{q_t}}, \quad \|u_t\|_{q_t} = \sqrt{\langle u_t, u_t\rangle_{q_t}}, \quad \|v_t\|_{q_t} = \sqrt{\langle v_t, v_t\rangle_{q_t}}. \tag{9}$$

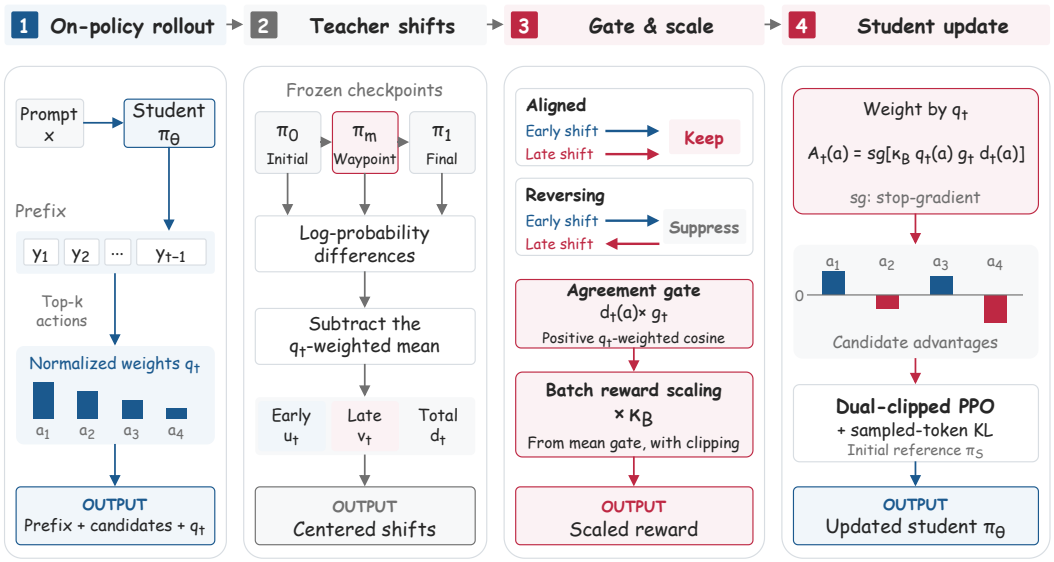


Figure 2: **Training pipeline of Waypoint-OPD.** Initial, Waypoint, and Final score candidates at student-generated prefixes. Their centered segment shifts define an agreement gate on the endpoint shift. Batch scaling and detached candidate advantages drive the student update. Waypoint measures consistency only.

3.2 Agreement Gating and Batch-Scale Normalization

When both centered segment norms exceed $\varepsilon = 10^{-8}$, we use the positive-cosine gate

$$g_t = \max(0, c_t). \tag{10}$$

If either norm is at most ε, we use the low-norm fallback $g_t = 1$. The gate $g_t \in [0, 1]$ is shared by all candidates at prefix s_t. Outside this fallback, it removes the endpoint signal when $c_t \leq 0$ and scales its magnitude by c_t otherwise. Algorithm and implementation conventions are given in Appendix B.

For illustration, consider two decompositions of a nonzero q_t-centered endpoint d, both outside the low-norm fallback. The pair $u = v = d/2$ gives $c = 1$ and $g = 1$, whereas $u = 2d$, $v = -d$ gives $c = -1$ and $g = 0$. Both yield $u + v = d$, so endpoint-only transfer cannot distinguish them. Appendix A.1 derives the endpoint's independence from Waypoint at fixed student prefixes and weights. Directional agreement alone does not guarantee beneficial supervision.

To stabilize the average gate scale across batches, we multiply all gated rewards in batch B by

$$\kappa_B = \mathrm{clip}\left(\frac{\tau}{\max\left(|B_{\mathrm{pref}}|^{-1} \sum_{t \in B_{\mathrm{pref}}} g_t, \varepsilon_{\mathrm{num}} \right)}, \kappa_{\min}, \kappa_{\max} \right), \tag{11}$$

where B_{pref} contains valid response prefixes in B, and $\varepsilon_{\mathrm{num}} > 0$ is a numerical floor preventing division by zero. The ratio targets a mean rescaled gate of $\tau > 0$, while clipping restricts κ_B to $[\kappa_{\min}, \kappa_{\max}]$, with $0 < \kappa_{\min} \leq \kappa_{\max}$. This shared positive multiplier preserves relative gate weights and zero gates. The effective multiplier $\kappa_B g_t$ can exceed one; batch scaling does not fix the reward-vector norm.

3.3 Dense Candidate Objective

The final coefficient for candidate action $a \in \mathcal{K}_t$ is

$$A_t(a) = \mathrm{sg}[\kappa_B\, q_t(a)\, g_t\, d_t(a)]\,. \tag{12}$$

Following the signal-minus-KL form of Eq. 4, the local on-policy surrogate to maximize is

$$\mathcal{J}_{\text{WP}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\Bigg[\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\sum_{t=1}^{T}\sum_{a\in\mathcal{K}_t} A_t(a)\log\pi_\theta(a \mid s_t)\right] - \beta\,\widehat{D}_{\text{KL}}(\pi_\theta(\cdot \mid x)\,\|\,\pi_S(\cdot \mid x))\Bigg], \tag{13}$$

Rollout samples and candidate supports are held fixed when differentiating this surrogate, and the stop-gradient operator sg detaches all factors in A_t. Candidate credit is dense on $\mathcal{K}_t$, while the KL penalty is estimated on sampled response tokens and anchors the student to π_S. The implementation uses dual-clipped PPO, which has the same gradient as this surrogate at the rollout policy (Appendix B).

Relative to Direct-OPD, Waypoint-OPD adds one frozen teacher checkpoint and scoring pass, without additional trainable parameters or student rollouts. Derivations and signal properties appear in Appendix A.

4 Experiments

We examine three questions:

Q1. Does Waypoint-OPD improve transfer across math, code, and model scales?

Q2. Does temporal consistency improve on endpoint transfer and Final-teacher imitation?

Q3. How do gate assignment, mean normalization, and Waypoint position affect performance?

Supplementary training-time and response-length analyses appear in Appendix H.

4.1 Models, data, and evaluation

Teacher checkpoints and student models. For each domain, we use three checkpoints from the same Qwen3-4B (Yang et al., 2025) RL run: Initial (π_0) before RL, Waypoint (π_m) at RL step 600, and Final (π_1) at RL step 1200. Math and code teachers use non-thinking RL. All three checkpoints remain frozen during transfer. We train Qwen3-8B, Qwen3-14B, and Qwen3-30B-A3B students using the teacher's RL-induced policy shifts. These students span three scales and dense and MoE architectures within one family with compatible tokenizers. The transfer objective does not require the teacher to outperform the student. We report all three teacher checkpoints and the Base students in the result tables to show pre-transfer performance.

Training and evaluation data. Math transfer uses DAPO-Math-17K prompts (Yu et al., 2025). We evaluate on seven benchmarks: AIME 2024, AIME 2025, AIME 2026, HMMT February 2025, HMMT November 2025, HMMT February 2026, and AMC 2023. For each problem, `mean@32` measures average correctness over 32 generated responses. Benchmark scores average these values over problems, and Avg. is the unweighted mean across benchmarks.

Code transfer uses a Codeforces-derived prompt set. Evaluation uses the 378-problem MBPP+ set from EvalPlus (Liu et al., 2023), reporting both the original MBPP tests (Austin et al., 2021) and the original plus additional tests (MBPP+), together with LiveCodeBench v4, v5, and v6 (Jain et al., 2025). We report `mean@8`, which averages correctness over eight generated responses per problem and then over the problems in each benchmark. Code Avg. is the unweighted mean over these five evaluation scores. Within each benchmark, methods share prompt templates, decoding parameters, maximum generation lengths, and sample counts.

Student checkpoint selection. For both domains, we use AIME 2024 as the validation set and select each trained student's checkpoint with the highest validation accuracy. All benchmark results for that student use this selected checkpoint.

Implementation. We build on `verl` (Sheng et al., 2025) and score the student's top-16 token IDs with the three frozen teachers. The transfer objective combines the dense distillation reward

with the student KL penalty. Answer graders are used only for evaluation. Implementation and training/evaluation configurations appear in Appendices B and C.

4.2 Baselines and controls

We compare against the Base student, standard on-policy distillation (OPD) from the Final teacher (Agarwal et al., 2024), OPD^2 with its sampled-token objective (Heo et al., 2026), and endpoint Direct-OPD using $\log \pi_1 - \log \pi_0$ (Feng et al., 2026). All methods share training and evaluation settings, except for method-specific hyperparameters.

Three controls on Qwen3-8B math retain centering and alter one component of the full method. **No gate** sets $g_t = 1$ while retaining the gate-mean normalization rule. **Shuffled gate** permutes gates among valid response prefixes within a batch, retaining each prefix's candidate weights and centered endpoint signal. **No normalization** fixes $\kappa_B = 1$ while retaining the positive-cosine gate. Appendix E gives definitions and invariants of the shuffle control. We separately vary the math Waypoint over teacher RL steps 500, 600, and 700 on Qwen3-8B in Appendix F.

5 Results and Analysis

We compare transfer performance across math and code at three student scales (Section 5.1). We examine learning trajectories and performance relative to the Final teacher on Qwen3-8B math (Section 5.2). We assess the roles of temporal gating, gate assignment, and mean normalization through ablations (Section 5.3).

5.1 Main results across domains and scales

Tables 1 and 2 report math scores and code averages. Full code results appear in Appendix D. Waypoint-OPD achieves the highest domain-average score among the evaluated student methods in all six settings.

Math. Waypoint-OPD improves over Direct-OPD, the strongest average baseline for all three students, by 2.18, 1.59, and 1.89 percentage points for Qwen3-8B, Qwen3-14B, and Qwen3-30B-A3B, respectively. It improves 19 of 21 benchmark comparisons. The exceptions are HMMT25N for 14B (47.60 to 47.08) and HMMT26F for 30B-A3B (33.05 to 32.77). These results demonstrate improvements over endpoint transfer across student scales and both dense and MoE architectures. The 14B and 30B-A3B students also exceed the Final teacher's 56.13 average by 3.07 and 1.72 points, respectively, while 8B essentially matches it at 56.14. The 14B student surpasses the teacher on all seven math benchmarks.

Code. Waypoint-OPD improves over Direct-OPD by 2.66, 1.52, and 1.19 percentage points on average for Qwen3-8B, Qwen3-14B, and Qwen3-30B-A3B, respectively, with gains in all 15 code comparisons. OPD^2 achieves higher scores on 14B LCB v5 and on 30B-A3B MBPP and MBPP+, while Waypoint-OPD retains the highest average for each student (Appendix D). The 14B and 30B-A3B students also surpass the Final teacher's 55.37 average by 2.45 and 2.17 points, respectively, each scoring higher on four of the five code evaluations.

Interpreting the endpoint comparison. The gains over Direct-OPD show that Waypoint-OPD improves transfer performance across both domains and all three student scales. OPD and OPD^2 provide complementary comparisons with Final-teacher imitation and selective distillation based on advantage sign agreement, respectively.

5.2 Learning dynamics

Figure 3 compares Waypoint-OPD and Direct-OPD at matched evaluation steps during Qwen3-8B math training. Unlike the selected-checkpoint results in Table 1, these trajectories show the performance gap over the same 100 updates.

Table 1: **Mathematical reasoning across three student sizes.** `mean@32` scores (%) on seven benchmarks and their average. Bold marks the best score per student. Teacher checkpoints are references.

Method	AIME24	AIME25	AIME26	HMMT25 Feb	HMMT25 Nov	HMMT26 Feb	AMC23	Avg.
Teacher: Qwen3-4B								
Initial	21.67	21.67	21.67	12.50	10.00	15.15	71.25	24.84
Waypoint	58.96	54.79	56.77	31.35	35.73	31.72	93.44	51.82
Final	65.83	60.00	60.83	35.83	40.83	35.13	94.45	56.13
Student: Qwen3-8B								
Base	29.17	20.83	18.33	11.67	13.33	18.94	70.00	26.04
OPD	63.65	49.69	54.90	29.06	38.65	32.58	88.75	51.04
OPD^2	64.48	48.33	53.96	28.44	40.52	31.53	**92.97**	51.46
Direct-OPD	67.29	55.31	57.40	31.77	40.83	33.71	91.41	53.96
Waypoint-OPD	**68.85**	**57.92**	**60.21**	**34.38**	**42.60**	**36.36**	92.66	**56.14**
Student: Qwen3-14B								
Base	30.00	27.50	20.00	10.00	11.67	18.94	73.13	27.32
OPD	69.17	55.52	58.44	36.77	45.21	31.82	**96.25**	56.17
OPD^2	69.48	46.98	54.38	33.13	41.98	31.91	91.95	52.83
Direct-OPD	70.10	56.46	61.98	37.92	**47.60**	35.13	94.06	57.61
Waypoint-OPD	**70.21**	**60.52**	**63.75**	**40.21**	47.08	**36.55**	96.09	**59.20**
Student: Qwen3-30B-A3B								
Base	31.67	22.50	22.50	10.00	13.33	21.21	75.00	28.03
OPD	63.54	49.06	57.50	30.83	37.71	28.03	91.88	51.22
OPD^2	64.90	51.77	60.42	31.56	38.54	30.21	91.33	52.68
Direct-OPD	69.17	54.69	61.98	38.96	41.56	**33.05**	92.34	55.96
Waypoint-OPD	**70.94**	**58.44**	**63.65**	**39.90**	**44.69**	32.77	**94.53**	**57.85**

Table 2: **Average code reasoning performance.** Average `mean@8` scores (%) over five evaluations. Qwen3-4B teacher checkpoints are shared references. Bold marks the best score per student. Full results: Appendix D.

Teacher reference: Qwen3-4B			
	Initial: 37.64	**Waypoint: 52.22**	**Final: 55.37**
Method	Qwen3-8B	Qwen3-14B	Qwen3-30B-A3B
Base	41.03	47.72	50.33
OPD	43.91	51.66	51.91
OPD^2	51.21	55.60	55.96
Direct-OPD	50.59	56.30	56.35
Waypoint-OPD	**53.25**	**57.82**	**57.54**

Consistent gains across training checkpoints. Waypoint-OPD delivers consistent gains over Direct-OPD across both benchmarks and training checkpoints. Excluding the shared initialization, it scores higher in 13 of the 15 benchmark–checkpoint comparisons, with two exceptions early in training. From step 60 onward, Waypoint-OPD leads on all three benchmarks at every recorded evaluation, showing that its advantage is sustained through the end of training. The step-100 gains are 4.17, 3.13, and 0.10 percentage points on AIME 2025, AIME 2026, and HMMT November 2025, respectively. These results show that gains extend beyond the selected checkpoint to sustained advantages during training.

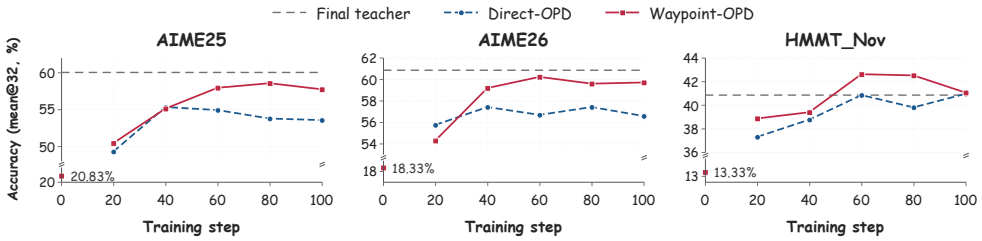


Figure 3: **Evaluation accuracy during training.** Qwen3-8B results on AIME 2025, AIME 2026, and HMMT November 2025 at steps 0, 20, 40, 60, 80, and 100. Scores are `mean@32` percentages. Solid red curves denote Waypoint-OPD and dashed blue curves denote Direct-OPD. Gray dashed lines show the Final Qwen3-4B teacher scores.

Comparison with the Final teacher. The gray reference lines distinguish gains over endpoint transfer from performance relative to the teacher. On HMMT November 2025, Waypoint-OPD exceeds the Final teacher's 40.83 at steps 60, 80, and 100, reaching 42.60 at step 60. On both AIME benchmarks, its recorded scores remain below the corresponding teacher reference. Improvements over Direct-OPD therefore coexist with different teacher–student orderings across benchmarks. The sustained HMMT advantage demonstrates that Waypoint-OPD can use the teacher's policy shifts to achieve performance beyond the Final teacher's score.

5.3 Gate and normalization ablations

Table 3: **Gate and normalization ablations on Qwen3-8B mathematical reasoning.** Bold marks the best score in each column.

Method	AIME24	AIME25	AIME26	HMMT25 Feb	HMMT25 Nov	HMMT26 Feb	AMC23	Avg.
No gate	66.25	54.27	56.67	32.60	39.79	34.66	90.23	53.50
Shuffled gate	63.33	52.92	54.58	30.00	37.08	31.06	89.45	51.20
No normalization	68.13	56.46	58.33	**34.79**	41.56	35.23	92.03	55.22
Direct-OPD	67.29	55.31	57.40	31.77	40.83	33.71	91.41	53.96
Waypoint-OPD	**68.85**	**57.92**	**60.21**	34.38	**42.60**	**36.36**	**92.66**	**56.14**

Removing the gate. On Qwen3-8B math, removing the gate lowers Avg. from 56.14 to 53.50, a decrease of 2.64 points, with lower scores on all seven benchmarks (Table 3). Retaining centering and normalization without consistency-dependent weighting therefore does not recover the full method's gain. This comparison does not isolate the effect of centering.

Disrupting the gate–prefix association. Shuffling gates lowers Avg. to 51.20, which is 4.94 points below the full method and 2.30 below the ungated variant, with the same ordering on every benchmark. Random reassignment is thus more detrimental than removing the selective weighting in this setting, supporting the value of matching each gate to the prefix whose temporal agreement produced it.

Removing mean normalization. Fixing $\kappa_B = 1$ while retaining the positive-cosine gate lowers Avg. by 0.92 points to 55.22. Normalization improves six of seven benchmarks. The exception is HMMT February 2025 (34.79 without normalization versus 34.38 with it). The unnormalized variant still exceeds Direct-OPD by 1.26 points on average and on all seven benchmarks. Thus, the advantage persists without batch normalization, which provides an additional average benefit here. The remaining gain cannot be assigned entirely to gating because centering and other differences from Direct-OPD are also retained.

Ablation insights. These ablations support temporal weighting, its prefix assignment, and an incremental benefit from normalization in Qwen3-8B math. Their gaps are conditional on the retained components, not independent additive contributions. For a fixed rollout batch, shuffling preserves the gate distribution and κ_B, while advantage and gradient norms depend additionally on the gate–signal pairing (Appendix E). Supplementary diagnostics show stronger average gate attenuation in later batches, while mean-gate normalization does not fix the reward norm (Appendix G).

6 Related Work

On-policy distillation and policy-shift transfer. Knowledge distillation transfers teacher distributions or generated sequences (Hinton et al., 2015; Kim & Rush, 2016), while on-policy variants supervise student-generated trajectories (Agarwal et al., 2024). Complementary work refines divergence objectives and the use of teacher- and student-generated data (Wen et al., 2023; Gu et al., 2024; Ko et al., 2024; 2025). Weak-to-strong generalization asks whether weaker supervision can elicit stronger capabilities (Burns et al., 2024). WSPO and proxy-tuning reuse a weak model's adaptation contrast for training and decoding, respectively (Zhu et al., 2025; Liu et al., 2024). Within on-policy training, Direct-OPD transfers an RL-induced policy change through an endpoint log-ratio on student rollouts (Feng et al., 2026). W2S-OPD constructs a proxy teacher by adding a weak-model logit contrast to student base logits, then minimizes reverse KL on student rollouts. Its contrasts span RL checkpoints, model scales, and correct versus incorrect hints (Yu et al., 2026). OPRD uses a centered teacher–reference shift to amplify the component of a verifier-driven gradient along that direction (Park et al., 2026). Waypoint-OPD builds on Direct-OPD, weighting local endpoint signals by agreement between shifts before and after a Waypoint checkpoint.

Selective supervision in distillation. Teacher–student compatibility affects the utility of on-policy supervision (Li et al., 2026). Prune-OPD uses support overlap for suffix attenuation (Yang et al., 2026). EOPD uses teacher uncertainty in its entropy-dependent forward KL (Jin et al., 2026), while PW-OPSD uses token position in its reference-conditioned self-distillation (Liu et al., 2026). OPD^2 gates sampled-token updates by sign agreement between centered teacher–base and teacher–student advantages (Heo et al., 2026), whereas RG-OPD gates trajectories using agreement between verifier-derived advantages and teacher–student sequence log-likelihood gaps (Akhondzadeh et al., 2026). Waypoint-OPD uses different evidence: consecutive shifts in the teacher's own RL trajectory over student-supported candidate actions. Their weighted directional agreement produces a prefix-level gate for the dense endpoint update, without verifier feedback during transfer. This agreement is a weighting criterion, not a guarantee of correctness.

Checkpoint selection and progressive distillation. A teacher's final performance need not determine its teaching utility: early stopping and intermediate checkpoint selection can improve distillation (Cho & Hariharan, 2019; Wang et al., 2022). Pro-KD and progressive distillation use successive teacher checkpoints as targets or a curriculum (Rezagholizadeh et al., 2022; Panigrahi et al., 2025). Direct-OPD also studies RL checkpoint choice for transfer (Feng et al., 2026). For fixed Initial and Final endpoints, Waypoint measures directional consistency between the two shifts. It supplies neither an imitation target nor a curriculum. The centered Initial-to-Final shift remains the transferred object.

7 Conclusion

We presented Waypoint-OPD, which uses agreement between consecutive teacher shifts to transfer RL-induced policy changes. An intermediate Waypoint checkpoint reveals whether early and late changes in relative action preferences reinforce or oppose each other at student-generated prefixes. This information, unavailable from endpoint differences alone, determines the weight of the centered Initial-to-Final shift in the student's on-policy objective. The method requires one additional frozen-teacher scoring pass and no verifier calls during transfer. Experiments on mathematical and code reasoning show gains over endpoint transfer across three Qwen3 student scales. Ablations on Qwen3-8B math support prefix-specific gate assignment and batch-scale normalization. These findings suggest that intermediate teacher checkpoints provide evidence for weighting local policy changes during student learning.

AI USE STATEMENT

We use generative AI tools for language polishing, assistance in clarifying mathematical derivations, and consistency checks of notation and exposition. These tools were not used for experimental design, implementation, or execution. We take full responsibility for the final content of this work.

ETHICS STATEMENT

Use of Human Annotations Human annotations are only used in methodological research at the beginning of the work, to assist in analyzing the feasibility of the proposed solution. Annotators consented to the use of data for research purposes. We ensure that the privacy of all annotators is protected throughout the annotation process, and all of them are adequately paid according to local standards. Human annotations are not applied during the evaluation of our method.

Risks In this paper, all datasets are obtained from official sources. The datasets adopted have been anonymized and do not contain offensive information. However, we cannot guarantee that the datasets do not contain socially harmful or toxic language.

REPRODUCIBILITY STATEMENT

The transfer objective is specified in Section 3, with supporting derivations and signal properties in Appendix A and pseudocode and optimization details in Appendix B. Section 4.1 describes the teacher checkpoints, student models, training and evaluation data, evaluation metrics, and student checkpoint selection. Appendix C reports hyperparameters, hardware, and evaluation settings.

Baselines and controls are described in Section 4.2, with precise definitions of the gate and normalization ablations in Appendix E and the Waypoint-position ablation setup in Appendix F. Per-benchmark results and domain averages appear in Tables 1 and 7, and gate and normalization ablation results appear in Table 3.

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 *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Mohammad Sadegh Akhondzadeh, Vijay Lingam, Atula Tejaswi, Chanakya Ekbote, Sujay Sanghavi, and Aleksandar Bojchevski. Reward-gated on-policy distillation. *arXiv preprint arXiv:2607.04037*, 2026.

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.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeffrey Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4971–5012. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/burns24b.html.

Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 4793–4801. IEEE, 2019. doi: 10.1109/ICCV.2019.00489. URL https://doi.org/10.1109/ICCV.2019.00489.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards. *Trans. Mach. Learn. Res.*, 2026, 2026. URL https://openreview.net/forum?id=9SkkifLopZ.

Shiyuan Feng, Huan-ang Gao, Haohan Chi, Hanlin Wu, Zhilong Zhang, Zheng Jiang, Bingxiang He, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Weak-to-strong generalization via direct on-policy distillation. *arXiv preprint arXiv:2607.05394*, 2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=5h0qf7IBZZ.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, Hao Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, Tao Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nat.*, 645(8081):633–638, 2025. doi: 10.1038/S41586-025-09422-Z. URL https://doi.org/10.1038/s41586-025-09422-z.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 138306–138322, 2026.

Byeongho Heo, Jaehui Hwang, Sangdoo Yun, and Dongyoon Han. On-policy delta distillation. *arXiv preprint arXiv:2607.15161*, 2026.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=chfJJYC3iL.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In Jian Su, Xavier Carreras, and Kevin Duh (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1317–1327. The Association for Computational Linguistics, 2016. doi: 10.18653/V1/D16-1139. URL https://doi.org/10.18653/v1/d16-1139.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 24872–24895. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/ko24c.html.

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. Distillm-2: A contrastive approach boosts the distillation of llms. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/ko25a.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In Jason Flinn, Margo I. Seltzer, Peter Druschel, Antoine Kaufmann, and Jonathan Mace (eds.), *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pp. 611–626. ACM, 2023. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

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*, 2026.

Alisa Liu, Xiaochuang Han, Yizhong Wang, Yulia Tsvetkov, Yejin Choi, and Noah A Smith. Tuning language models by proxy. *arXiv preprint arXiv:2401.08565*, 2024.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html.

Xiaogeng Liu, Xinyan Wang, Yingzi Ma, Yechao Zhang, and Chaowei Xiao. When are teacher tokens reliable? position-weighted on-policy self-distillation for reasoning. *arXiv preprint arXiv:2605.21606*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

Abhishek Panigrahi, Bingbin Liu, Sadhika Malladi, Andrej Risteski, and Surbhi Goel. Progressive distillation induces an implicit curriculum. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=wPMRwmytZe.

Youngrok Park, Sangmin Bae, Hojung Jung, Jongwoo Ko, Yunseon Choi, Young Jin Kim, Pashmina Cameron, Aaron Courville, and Se-Young Yun. Eliciting weak-to-strong generalization with on-policy reverse distillation. *arXiv preprint arXiv:2609.08798*, 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

Mehdi Rezagholizadeh, Aref Jafari, Puneeth S. M. Saladi, Pranav Sharma, Ali Saheb Pasand, and Ali Ghodsi. Pro-kd: Progressive distillation by following the footsteps of the teacher. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (eds.), *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pp. 4714–4727. International Committee on Computational Linguistics, 2022. URL https://aclanthology.org/2022.coling-1.418.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems, EuroSys 2025, Rotterdam, The Netherlands, 30 March 2025 - 3 April 2025*, pp. 1279–1297. ACM, 2025. doi: 10.1145/3689031.3696075. URL https://doi.org/10.1145/3689031.3696075.

Chaofei Wang, Qisen Yang, Rui Huang, Shiji Song, and Gao Huang. Efficient knowledge distillation from model checkpoints. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/03e0712bf85ebe7cec4f1a7fc53216c9-Abstract-Conference.html.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. f-divergence minimization for sequence-level knowledge distillation. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pp. 10817–10834. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.ACL-LONG.605. URL https://doi.org/10.18653/v1/2023.acl-long.605.

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.

Zhicheng Yang, Zhijiang Guo, Yifan Song, Minrui Xu, Yongxin Wang, Yiwei Wang, Xiaodan Liang, and Jing Tang. Prune-opd: Efficient and reliable on-policy distillation for long-horizon reasoning. *arXiv preprint arXiv:2605.07804*, 2026.

Deheng Ye, Zhao Liu, Mingfei Sun, Bei Shi, Peilin Zhao, Hao Wu, Hongsheng Yu, Shaojie Yang, Xipeng Wu, Qingwei Guo, Qiaobo Chen, Yinyuting Yin, Hao Zhang, Tengfei Shi, Liang Wang, Qiang Fu, Wei Yang, and Lanxiao Huang. Mastering complex control in MOBA games with deep reinforcement learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 6672–6679. AAAI Press, 2020. doi: 10.1609/AAAI.V34I04.6144. URL https://doi.org/10.1609/aaai.v34i04.6144.

Fangxu Yu, Weijia Xu, Michael Xu, Tianyi Zhou, and Zinan Lin. Weak-to-strong on-policy distillation. *arXiv preprint arXiv:2607.26246*, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch FSDP: experiences on scaling fully sharded data parallel. *Proc. VLDB Endow.*, 16(12):3848–3860, 2023. doi: 10.14778/3611540.3611569. URL https://www.vldb.org/pvldb/vol16/p3848-huang.pdf.

Wenhong Zhu, Zhiwei He, Xiaofeng Wang, Pengfei Liu, and Rui Wang. Weak-to-strong preference optimization: Stealing reward from weak aligned model. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=f7KxfUrRSb.

A Derivations and Signal Properties

A.1 Endpoint preservation and gate properties

At fixed s_t, $\mathcal{K}_t$, and q_t, with finite teacher log scores, linearity of centering gives

$$u_t + v_t = C_{q_t}[\delta_t^{(1)} + \delta_t^{(2)}] = C_{q_t}[\log \pi_1 - \log \pi_0] = d_t. \tag{14}$$

Hence Waypoint affects g_t and κ_B, while the transferred endpoint d_t remains fixed. This comparison holds at fixed student prefixes and weights; subsequent training trajectories can differ.

The remaining properties follow directly from the definitions. First, $C_{q_t}[z + b] = C_{q_t}[z]$ for any action-independent constant b, so such offsets in teacher scores leave the transfer signal unchanged. Second, outside the low-norm fallback, the gate is one for positively collinear shifts and zero for nonpositive weighted cosine. Independent positive rescaling of u_t and v_t leaves this gate unchanged provided neither segment enters the fallback; their sum d_t can change. Finally, $g_t \in [0, 1]$ and $\kappa_B > 0$ ensure that $A_t(a)$ cannot reverse the sign of $d_t(a)$. The effective multiplier $\kappa_B g_t$ can exceed one: batch normalization targets the mean gate, not a fixed reward norm. These are properties of the construction; the usefulness of temporal weighting is evaluated empirically.

A.2 Zero-sum coefficients and the local gradient

Centering under q_t immediately gives

$$\sum_{a \in \mathcal{K}_t} A_t(a) = \kappa_B g_t \sum_{a \in \mathcal{K}_t} q_t(a) d_t(a) = 0. \tag{15}$$

With $\mathcal{K}_t$ fixed and A_t detached, this cancels the common softmax term. Writing z_b for the student logit of vocabulary token b at s_t,

$$\frac{\partial}{\partial z_b}\left[-\sum_{a \in \mathcal{K}_t} A_t(a) \log \pi_\theta(a \mid s_t)\right] = \begin{cases} -A_t(b), & b \in \mathcal{K}_t, \\ 0, & b \notin \mathcal{K}_t. \end{cases} \tag{16}$$

The PPO surrogate has this gradient at the rollout policy (Appendix B.1). This local result excludes the KL term and does not imply that a shared-parameter update preserves candidate probability mass or leaves noncandidate probabilities unchanged.

B Algorithm and Implementation Details

Algorithm 1 Agreement-gated on-policy distillation of RL-induced policy shifts

Require: Prompts $\mathcal{D}$, student and reference policies π_θ, π_S, frozen Initial, Waypoint, and Final teachers π_0, π_m, π_1, hyperparameters k, τ, ε, scale bounds, and KL coefficient β.

for each update **do**
  Sample prompts x and responses $y \sim \pi_\theta(\cdot \mid x)$.
  for each valid prefix $s_t = (x, y_{<t})$ **do**
    Obtain $\mathcal{K}_t = \mathrm{TopK}(\pi_\theta(\cdot \mid s_t), k)$ and detached normalized weights q_t.
    Score $\mathcal{K}_t$ with π_0, π_m, and π_1.
    Form $u_t = C_{q_t}[\log \pi_m - \log \pi_0]$ and $v_t = C_{q_t}[\log \pi_1 - \log \pi_m]$.
    Compute c_t with Eq. 9 and set $g_t = \max(0, c_t)$. Use $g_t = 1$ if either centered norm is at most ε.
    Set $d_t = u_t + v_t$.
  end for
  Compute κ_B with Eq. 11.
  Set $A_t(a) = \mathrm{sg}[\kappa_B \, q_t(a) \, g_t \, d_t(a)]$.
  Update π_θ using the PPO surrogate in Eq. 17 and the KL penalty in Eq. 13.
end for

B.1 CANDIDATE-WISE DUAL-CLIPPED PPO

We use the candidate-wise dual-clipped PPO surrogate (Schulman et al., 2017; Ye et al., 2020). Let π_{old} denote the student policy at rollout time, with $r_t(a) = \pi_\theta(a \mid s_t)/\pi_{\text{old}}(a \mid s_t)$, $\bar{r}_t(a) = \text{clip}(r_t(a), 1-\epsilon, 1+\epsilon)$, and let $c > 1$ be the dual-clip constant, distinct from c_t. For each candidate, the maximized surrogate is

$$J_t(a) = \begin{cases} \min\{r_t(a)A_t(a), \bar{r}_t(a)A_t(a)\}, & A_t(a) \geq 0, \\ \max\{\min[r_t(a)A_t(a), \bar{r}_t(a)A_t(a)], cA_t(a)\}, & A_t(a) < 0. \end{cases} \tag{17}$$

The loss sums $-J_t(a)$ over candidates and uses token-mean aggregation over valid response tokens, together with the sampled-token KL penalty to π_S in Eq. 13. At the rollout policy, $r_t(a) = 1$ and clipping is inactive, so maximizing $\sum_{a \in \mathcal{K}_t} J_t(a)$ gives the same local distillation update as Eq. 13 (Appendix A.2).

B.2 IMPLEMENTATION CONVENTIONS

Candidate supports $\mathcal{K}_t$ and detached coefficients $A_t(a)$ remain fixed during each student update. Batch-scale normalization uses valid response prefixes across the global rollout batch B. The low-norm fallback retains the endpoint signal when a segment has too little weighted variation to define a reliable direction, using $\varepsilon = 10^{-8}$. The normalization floor ε_{num} is the smallest positive normal float32 value ($\approx 1.18 \times 10^{-38}$), distinct from the low-norm threshold. Training and evaluation configurations appear in Appendix C.

C FULL EXPERIMENTAL PROTOCOL

We use `verl` (Sheng et al., 2025) with FSDP (Zhao et al., 2023) for training and vLLM (Kwon et al., 2023) for rollouts. Training and inference are performed on a single node with 8 NVIDIA H20 GPUs. Tables 4, 5, and 6 summarize the teacher and transfer training hyperparameters, model configurations, and evaluation benchmarks.

Teacher training. For math RL training, we select approximately 57K examples with difficulty levels ≥ 6 from DeepMath (DeepMath-103K) (He et al., 2026). For code RL training, we use Eurus-RL-Code (Cui et al., 2026), which contains approximately 25K training examples. We train both Qwen3-4B teachers with group relative policy optimization (GRPO) (Shao et al., 2024) for 1,200 optimization steps using the same hyperparameters in Table 4; only the training datasets differ.

Table 4: GRPO training configuration shared by the math and code teachers.

Setting	Value
Train batch size	128
Micro batch size	128
Rollout n	8
Maximum prompt length	2,048
Maximum response length	16,384
Temperature	1.0
Top-p	1.0
Learning rate	10^{-6}
Optimization steps	1,200
KL coefficient	0.0

Evaluation settings. Math and code evaluation use the same rollout settings: temperature 0.7, top-p 0.95, and a maximum generation length of 16,384 tokens. We report `mean@32` for math and `mean@8` for code, averaging correctness over 32 and eight responses per problem, respectively, and then over the problems in each benchmark. MBPP (Austin et al., 2021) and MBPP+ from EvalPlus (Liu et al., 2023) use the same problems with the original tests and the original plus additional tests, respectively. We also evaluate on LiveCodeBench v4, LiveCodeBench v5, and LiveCodeBench v6

Table 5: Transfer configuration shared by math and code, except for the transfer datasets.

Setting	Value
Math training dataset	DAPO-Math-17K
Code training dataset	Codeforces-derived
Train batch size	256
Mini batch size	64
Rollout n	1
Maximum prompt length	2,048
Maximum response length	16,384
Temperature	1.0
Learning rate	10^{-6}
Optimization steps	100
KL coefficient β	2.5
Student support k	16
Low-norm threshold ε	10^{-8}
Target gate mean τ	0.8
Scale clip	$[0.75, 1.5]$

(Jain et al., 2025). For both domains, we select each trained student's checkpoint by its validation accuracy on AIME 2024 and report all benchmark results from that checkpoint.

Table 6: Model and evaluation configuration.

Item	Value
Initial teacher	Qwen3-4B
Waypoint teacher	RL step 600
Final teacher	RL step 1200
Students	Qwen3-8B, Qwen3-14B, Qwen3-30B-A3B
Math evaluation	AIME 2024, AIME 2025, AIME 2026, HMMT February 2025, HMMT November 2025, HMMT February 2026, AMC 2023
Code evaluation	MBPP, MBPP+, LiveCodeBench v4, LiveCodeBench v5, LiveCodeBench v6

D Full Code Results

Table 7 presents the complete code evaluation results corresponding to the averages summarized in Table 2 in the main text. It includes the three teacher checkpoints and all evaluated student methods.

E Component Ablation Details

We detail the three controls evaluated on Qwen3-8B mathematical reasoning in Table 3. All controls retain q_t-centering and the student KL penalty and use the training and evaluation settings in Appendix C. Each control collects rollouts from its current student policy.

No gate. This control sets $g_t = 1$ at every valid response prefix while retaining the centered endpoint signal and the gate-mean normalization rule in Eq. 11. With the default settings, the rule reduces to the constant multiplier

$$\kappa_B = \mathrm{clip}(0.8, 0.75, 1.5) = 0.8.$$

This control removes consistency-dependent weighting while retaining centering and the normalization rule.

Shuffled gate. We compute gates using the standard rule, including the low-norm fallback, and randomly permute them among valid response prefixes in the global rollout batch B. For a permutation

Table 7: **Full code reasoning results across three student sizes.** Scores are `mean@8` percentages. Avg. is the unweighted mean across the five evaluations. Bold marks the best score within each student block. Teacher checkpoints are reference rows.

Method	MBPP	MBPP+	LCB v4	LCB v5	LCB v6	Avg.
Teacher: Qwen3-4B						
Initial	62.53	54.53	25.37	20.36	25.43	37.64
Waypoint	83.33	70.37	36.01	37.20	34.21	52.22
Final	88.00	72.65	38.61	40.57	37.00	55.37
Student: Qwen3-8B						
Base	67.16	58.63	30.82	23.35	25.21	41.03
OPD	69.58	57.67	35.64	26.95	29.71	43.91
OPD^2	79.10	68.68	36.54	37.72	34.00	51.21
Direct-OPD	78.14	65.81	36.63	37.95	34.43	50.59
Waypoint-OPD	**82.97**	**69.38**	**37.87**	**40.19**	**35.86**	**53.25**
Student: Qwen3-14B						
Base	83.86	71.69	30.94	26.95	25.14	47.72
OPD	84.49	72.62	35.64	33.53	32.00	51.66
OPD^2	85.98	72.92	38.61	**43.56**	36.93	55.60
Direct-OPD	85.32	73.58	43.44	42.96	36.21	56.30
Waypoint-OPD	**87.57**	**74.60**	**45.54**	43.11	**38.29**	**57.82**
Student: Qwen3-30B-A3B						
Base	86.34	73.18	32.55	28.67	30.93	50.33
OPD	86.61	72.25	35.15	32.49	33.07	51.91
OPD^2	**88.23**	**74.44**	40.10	41.62	35.43	55.96
Direct-OPD	87.24	73.77	42.33	41.54	36.86	56.35
Waypoint-OPD	88.00	74.21	**43.94**	**42.89**	**38.64**	**57.54**

σ, the advantage is

$$A_t^{\text{shuffle}}(a) = \text{sg}[\kappa_B q_t(a) g_{\sigma(t)} d_t(a)].$$

Candidate weights q_t and endpoint signals d_t remain at their original prefixes. For a fixed rollout batch, the permutation preserves the gate distribution and its mean, leaving κ_B unchanged. It breaks the association between each prefix and its own temporal-agreement weight. Advantage and gradient norms depend additionally on this gate–signal pairing.

No normalization. This control fixes $\kappa_B = 1$ while retaining centering, the positive-cosine gate, and the low-norm fallback:

$$A_t^{\text{unnorm}}(a) = \text{sg}[q_t(a) g_t d_t(a)].$$

For a fixed batch, removing the common multiplier preserves relative gate weights across prefixes while removing the batch-dependent rescaling of the distillation signal relative to the retained KL penalty.

F Sensitivity to Waypoint Checkpoint Selection

Experimental setup. We examine whether transfer performance depends on the precise position of the intermediate teacher checkpoint. On Qwen3-8B mathematical reasoning, we replace the default Waypoint at teacher RL step 600 with checkpoints at steps 500 and 700 from the same Qwen3-4B RL run. The Initial teacher before RL and the Final teacher at step 1200 remain fixed. The other training and evaluation settings follow the default Qwen3-8B math setup in Appendix C. Table 8 reports all seven benchmark scores and their unweighted average.

Small changes in aggregate performance. The average scores are 56.18, 56.14, and 56.13 for steps 500, 600, and 700, respectively, giving a total range of only 0.05 percentage points. Relative

Table 8: **Sensitivity to the Waypoint checkpoint on Qwen3-8B mathematical reasoning.** Only the intermediate teacher checkpoint is varied, while the Initial and Final teachers are fixed. Step numbers refer to the teacher's RL training. Bold marks the best score in each column.

Waypoint	AIME24	AIME25	AIME26	HMMT25 Feb	HMMT25 Nov	HMMT26 Feb	AMC23	Avg.
Step 500	69.06	**58.13**	59.79	33.96	42.60	**36.74**	**92.97**	**56.18**
Step 600	68.85	57.92	**60.21**	34.38	42.60	36.36	92.66	56.14
Step 700	**69.27**	57.60	59.79	**34.58**	**42.81**	36.36	92.50	56.13

to the default step 600, moving the Waypoint earlier changes Avg. by $+0.04$ points, and moving it later changes Avg. by -0.01 points. All three choices exceed the Direct-OPD average of 53.96 by 2.17–2.22 points and outperform it on every individual benchmark (Table 1). Thus, the observed transfer advantage is preserved across these neighboring Waypoints.

Stability across Waypoint choices. Across the three tested Waypoints, the score range on each benchmark is at most 0.62 percentage points, with the largest range on HMMT25 February (33.96–34.58). Together with the 0.05-point range in Avg. and the consistent gains over Direct-OPD, these results show stable transfer performance across the tested neighboring Waypoints on Qwen3-8B math.

G Gate Diagnostics

Figure 4 examines agreement gating and batch-scale normalization over 100 training steps of Waypoint-OPD on Qwen3-8B mathematical reasoning, using the configuration in Appendix C.

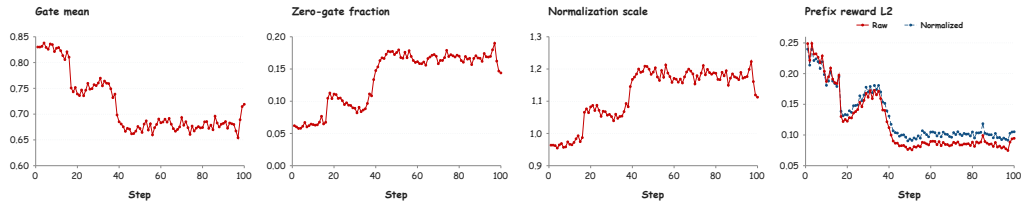


Figure 4: **Gate diagnostics on Qwen3-8B mathematical reasoning.** Statistics for Waypoint-OPD over 100 training steps. From left to right: mean gate g_t, zero-gate fraction, batch-scale normalization multiplier κ_B, and mean per-prefix reward-vector L2 norm. Raw uses the gated, student-weighted endpoint signal $q_t(a)g_t d_t(a)$; Normalized uses the final candidate coefficients $A_t(a)$ in Eq. 12 from the same run. All statistics aggregate valid response prefixes.

Temporal weighting over training. The mean gate g_t decreases from 0.810 in the first 20 steps to 0.683 in the last 20, while the zero-gate fraction increases from 7.26% to 16.59%. These summaries are unweighted averages of step-level statistics. The gate therefore applies stronger average attenuation before batch-scale normalization in later batches and removes the centered endpoint signal at a larger fraction of valid response prefixes. With the Initial, Waypoint, and Final teachers frozen, these statistics track temporal agreement on the evolving student-generated prefixes and candidate distributions.

Batch-scale normalization. The multiplier κ_B ranges from 0.955 to 1.223, remaining within the configured bounds $[\kappa_{\min}, \kappa_{\max}] = [0.75, 1.5]$. It is below one for the first 16 steps and above one thereafter, maintaining the mean rescaled gate at the target $\tau = 0.8$ to numerical precision. As specified in Eq. 11, this shared positive multiplier preserves relative gate weights and zero gates. It also scales each prefix's reward-vector L2 norm by κ_B. Over the last 20 steps, the mean per-prefix norms average 0.0855 before scaling and 0.1002 after scaling. Both norms decrease overall during training, while the mean rescaled gate remains stable. These diagnostics show how batch-scale

normalization maintains the average gate scale as temporal weighting adapts to student-generated prefixes. The performance benefit is reflected in Table 3: mean normalization improves the Qwen3-8B math average from 55.22 to 56.14.

H Training Time and Response Length

Table 9 compares the total training time over 100 training steps of Qwen3-8B on math using eight H20 GPUs, excluding evaluation. Both methods follow the configuration in Appendix C. Waypoint-OPD completes training in 8 h 2 min 21 s, compared with 8 h 21 min 3 s for Direct-OPD, reducing total training time by 18 min 42 s (3.73%). The corresponding mean step times are 289.41 s and 300.63 s, a reduction of 11.22 s per step. Total GPU-hours decrease from 66.81 to 64.31.

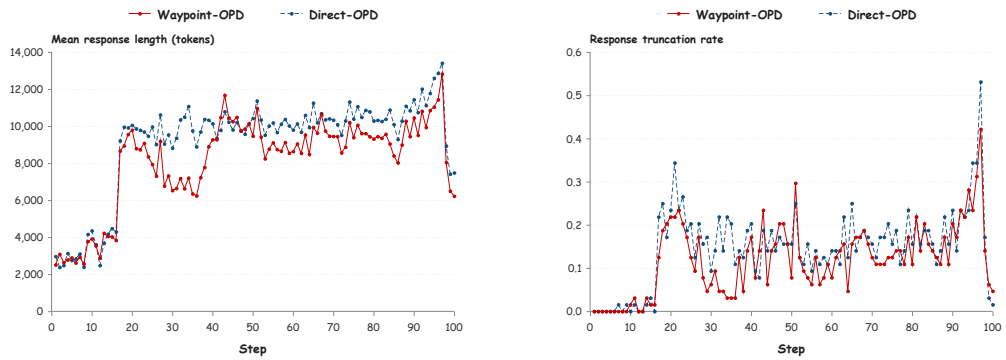
Table 9: **Training-time comparison on Qwen3-8B math.** Total training time over 100 training steps on eight H20 GPUs, excluding evaluation.

Method	Time (h:m:s)	Seconds/step	GPU-hours
Direct-OPD	8:21:03	300.63	66.81
Waypoint-OPD	8:02:21	289.41	64.31

GPU-hours are computed as eight times the reported training duration in hours and exclude evaluation.

Relative to Direct-OPD, Waypoint-OPD adds one scoring pass with the frozen Waypoint checkpoint while using the same number of student rollouts. With this additional source of temporal consistency information, Waypoint-OPD achieves a lower measured training time in this setting.

Figure 5 compares rollout generation over the same 100 training steps. Response length is the mean number of tokens per rollout response at each training step, and response truncation rate is the fraction of rollout responses reaching the maximum length of 16,384 tokens. Waypoint-OPD has a lower mean response length at 87 of the 100 steps and a lower response truncation rate at 62 steps, with equal rates at 18 steps. Averaged uniformly across the 100 steps, the response truncation rate decreases from 14.83% for Direct-OPD to 11.97% for Waypoint-OPD, a reduction of 2.86 percentage points. These results show that Waypoint-OPD produces shorter responses at most training steps and reaches the response-length limit less frequently on average.



(a) Mean response length

(b) Response truncation rate

Figure 5: **Response length and response truncation rate.** The two panels compare Qwen3-8B trained with Waypoint-OPD (solid red) and Direct-OPD (dashed blue) over 100 training steps. (a) Mean number of tokens per rollout response at each training step. (b) Fraction of rollout responses reaching the maximum length of 16,384 tokens; 0.2 corresponds to 20%.