

DTDD: Divergence-Triggered Dynamic Distillation for Reliable On-Policy Supervision

Qijia He^{1,2,*}, Zequn Li¹, Spencer Luo¹, Tianle Chen^{1,3}, Pradyumna Narayana¹, Zixian Ma², Ranjay Krishna², Lan Nie¹ and Sugato Basu¹

¹ Google, ² University of Washington, ³ Boston University

Project page: dtdd-proj.github.io

On-policy distillation (OPD) provides dense token-level supervision on student rollouts, but this supervision can become unreliable when the student and teacher diverge, either because of a large initial policy gap or compounding errors in long-horizon reasoning and tool use. We introduce Divergence-Triggered Dynamic Distillation (DTDD), an adaptive OPD framework that uses segment-level student-teacher divergence to trigger teacher takeover during training. When divergence triggers an intervention during reasoning, the student prefix continues to receive standard OPD supervision. The teacher then generates a recovery suffix, which the student learns through behavior cloning, using cross-entropy on the teacher-generated tokens. We evaluate DTDD in three regimes: long-horizon reasoning, distillation from a base initialization without offline supervised fine-tuning (SFT), and multi-turn agents. We also compare alternative trigger designs and teacher-intervention rates. DTDD reduces the post-warmup peak gradient norm by $8.6\times$ and, from a base initialization, reaches 21.3% held-out math accuracy in $2.6\times$ fewer optimizer updates than OPD; both methods finish near 22%. It improves best-observed performance by 4.8 and 4.1 percentage points on ALFWorld with LoRA and agentic code, respectively. Ablations show that segment-averaged triggers outperform token-spike triggers. Behavior cloning is also more effective than the tested OPD surrogate for training teacher-generated recoveries. Across 23 experimental arms, takeover tends to help when the teacher substantially outperforms the OPD-trained student, provided that the teacher can recover from student-visited states, the student can learn those recoveries, and teacher intervention is neither excessive nor insufficient.

Keywords: on-policy distillation, knowledge distillation, large language models, reasoning, multi-turn agents

1. Introduction

Knowledge distillation transfers capabilities from a strong teacher to a smaller student (Hinton et al., 2015). Training on fixed teacher-generated sequences creates a distribution mismatch: the student learns from teacher-generated prefixes during training but must condition on its own prefixes at inference (Agarwal et al., 2024; Kim and Rush, 2016). One motivation for on-policy distillation (OPD) is to reduce this state-distribution mismatch by sampling student trajectories and querying the teacher at student-visited states (Agarwal et al., 2024). OPD therefore combines student-state coverage with dense teacher supervision. The quality of this signal depends on whether the teacher provides learnable guidance along the states the student visits.

Reliable on-policy supervision can fail for two broad reasons: the teacher signal may be poorly matched to the student at initialization, or it may deteriorate as errors accumulate along a rollout. In reasoning, incompatible student-teacher thinking patterns and limited teacher information gain can make early OPD ineffective, motivating supervised fine-tuning (SFT) on teacher rollouts and teacher-aligned prompts (Li et al., 2026b). Even when initial learning is viable, longer reasoning rollouts can amplify deviations and destabilize training (Luo et al., 2026). In multi-turn tool use, an

*This work was done while Qijia was an intern at Google Ads AI.
Corresponding author(s): heqj3@uw.edu, sugato@google.com

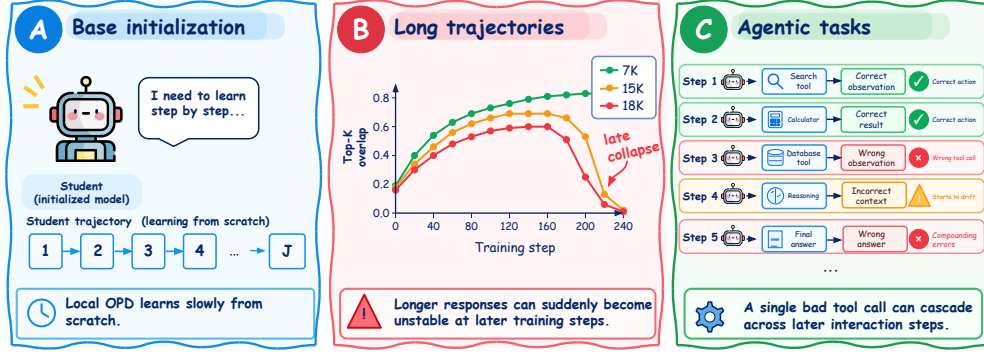


Figure 1 | Representative failures of local OPD supervision: (A) slow learning from a base initialization, (B) late instability under long generation budgets, and (C) compounding errors after an incorrect action in multi-turn interaction. Appendix A reports measured examples of (A) and (B).

incorrect tool call additionally changes subsequent observations and contexts, causing errors and student–teacher divergence to compound across interactions (Wang et al., 2026; Zhong et al., 2026). Figure 1 illustrates these cases. Despite their different origins, each can move the student into prefixes where local reverse-KL feedback is no longer informative (Fu et al., 2026; Liu et al., 2026). OPD ensures coverage of student-visited states, but effective learning still requires a reliable and learnable signal at those states.

Prior work addresses unreliable OPD supervision through two broad strategies. The first modifies the token-level learning signal while retaining the student trajectory: StableOPD constrains policy drift and mixes in fixed demonstrations, TGPO introduces directional teacher guidance, and SOD attenuates distillation in high-divergence regions (Liu et al., 2026; Luo et al., 2026; Zhong et al., 2026). These methods protect or redirect optimization within the student trajectory. The second strategy introduces coherent teacher-generated corrections, including curriculum-controlled trajectory segments, expert suffixes, rewritten trajectories, per-step teacher targets, and temporary teacher relays (Jiang et al., 2026; Lauffer et al., 2025; Li et al., 2026a; Peng et al., 2025; Sun et al., 2026; Wang et al., 2026; Xu et al., 2026). Together, these approaches motivate studying where to intervene and how to train on the resulting trajectory. We investigate segment-averaged divergence as a boundary for retaining OPD on the student prefix and applying behavior cloning to a teacher-generated recovery suffix.

To this end, we propose **DTDD**, a **Divergence-Triggered Dynamic Distillation** framework built around a simple teacher-takeover rule during training. DTDD partitions each student rollout into short token segments and measures student–teacher divergence within each segment. When a segment exhibits high divergence, DTDD rewinds to its beginning and lets the teacher complete the trajectory from the retained student prefix. The prefix continues to receive standard OPD supervision, while the teacher-generated suffix is trained with behavior cloning, using cross-entropy on teacher-generated tokens. Without a trigger, DTDD reduces to standard OPD. The same mechanism can intervene sparsely to stabilize an already capable student or more frequently to provide coherent supervision from a base initialization. In multi-turn settings, takeover instead replaces the current reasoning-and-tool-call action before interaction continues.

We evaluate DTDD on long-horizon math, base-model distillation without offline SFT, and multi-turn agents. With 15K- and 18K-token response budgets, it preserves held-out accuracy while reducing post-warmup peak gradient norms by 8.6× and 7.0×. From Qwen3-1.7B-Base, high-dose DTDD reaches 21.3% held-out accuracy after roughly 70 updates and stays near that level; OPD reaches

it around update 180, and both finish near 22%. At comparable accuracy, DTDD requires 22.0 serial hours and 43M teacher-written tokens; offline SFT followed by OPD requires 42.8 hours and 95M. Action-level takeover improves best-observed performance by 4.8 and 4.1 percentage points on ALFWorld with LoRA and agentic code, respectively. Ablations isolate two design choices: the takeover trigger and the teacher-suffix loss. Segment-averaged divergence outperforms individual token spikes, and behavior cloning outperforms the tested OPD surrogate on teacher suffixes. Across 23 arms in 13 settings, 19 follow a common pattern. DTDD tends to help when the teacher is stronger than the OPD student, can recover from student prefixes, and provides continuations the student can learn; the intervention rate must provide enough guidance without replacing on-policy training.

2. Related Work

On-policy distillation. Knowledge distillation transfers teacher knowledge through output-level supervision (Hinton et al., 2015). Conventional off-policy approaches either match teacher token distributions under fixed contexts or train on complete teacher-generated sequences (Kim and Rush, 2016). On-policy distillation instead computes teacher supervision along student-generated trajectories (Agarwal et al., 2024; Gu et al., 2024). Recent analyses identify thinking-pattern incompatibility, limited teacher information gain, rollout-length effects, and unreliable sampled-token feedback as important failure sources (Fu et al., 2026; Li et al., 2026b). To reduce initial pattern mismatch, Li et al. (2026b) first fine-tune the student on offline teacher rollouts before applying OPD. StableOPD (Luo et al., 2026) combines a reference-based divergence constraint with auxiliary SFT on fixed, high-quality “golden” demonstrations to counter repetition-driven length inflation. Other methods reshape supervision within the student rollout: SOD (Zhong et al., 2026) reweights interaction steps using step-level divergence, while Prune-OPD (Yang et al., 2026) downweights and truncates a rollout after detecting prefix drift. These approaches improve initialization, augment the training distribution, or modify the original learning signal. DTDD complements them by conditionally regenerating an unreliable region of the current student trajectory.

Teacher-generated trajectory intervention. A complementary line of work changes the training trajectory itself by introducing teacher-generated corrections. These methods differ primarily in how the intervention boundary is chosen. TRD (Jiang et al., 2026) rewrites complete student rollouts, whereas SOPD (Sun et al., 2026) generates a teacher target step from each student-visited step prefix. Other methods allocate trajectory control independently of the current student–teacher discrepancy. OEC (Lauffer et al., 2025) uses a random one-way expert handoff; TCOD (Wang et al., 2026) schedules the horizon controlled by the student, with its backward-to-forward curriculum replaying prefixes from pre-collected successful teacher trajectories. More closely related to our setting, Relay-OPD (Xu et al., 2026) and DASH-OPD (Xia et al., 2026) use state-dependent signals to insert temporary teacher segments. DASH-OPD also uses different distillation objectives for student- and teacher-generated turns. AdaSwitch (Peng et al., 2025) uses a context-relative next-token divergence spike to trigger a one-way teacher takeover and optimizes the resulting hybrid sequence with a unified KD objective. Compared with AdaSwitch, DTDD makes two design changes. First, it averages divergence over short token segments and triggers at the first segment whose average exceeds the threshold, rather than reacting to an individual token spike. Second, in reasoning tasks, DTDD rewinds to that segment, retains the preceding student prefix, and asks the teacher to regenerate the remaining suffix. The student prefix is trained with OPD, while the teacher-generated suffix is trained with behavior cloning. In multi-turn tasks, the same trigger replaces the current action or transfers the remaining episode to the teacher, depending on the interactive realization. At inference time, ARCHER (He et al., 2026) chooses among reflection, replanning, and escalation to a stronger model after execution failures.

DTDD uses teacher recovery as supervision during student training.

3. Divergence-Triggered Dynamic Distillation

We first formulate the OPD objective, then present DTDD and formalize its segment-level trigger, teacher takeover, and training objective.

3.1. On-Policy Distillation Preliminaries

Let $x \sim \mathcal{D}_x$ be a prompt, π_θ the student, and π_T a frozen teacher. The student samples an autoregressive rollout $y^S = (y_1^S, \dots, y_L^S) \sim \pi_\theta(\cdot | x)$ under a maximum generation budget R , where $L \leq R$. At position t , let $h_t = (x, y_{<t}^S)$ denote the student-visited history, with $p_t(v) = \pi_\theta(v | h_t)$ and $q_t(v) = \pi_T(v | h_t)$ denoting the student and teacher next-token distributions. On-policy distillation evaluates the teacher along trajectories sampled from the current student (Agarwal et al., 2024). We use a reverse-KL objective (Gu et al., 2024), writing $\ell_t^{\text{RKL}}(\theta) = D_{\text{KL}}(p_t \| q_t)$ for its per-position loss:

$$\mathcal{L}_{\text{OPD}}(\theta) = \mathbb{E}_{\substack{x \sim \mathcal{D}_x, \\ y^S \sim \pi_\theta(\cdot | x)}} \left[\frac{1}{L} \sum_{t=1}^L \ell_t^{\text{RKL}}(\theta) \right]. \quad (1)$$

To obtain multi-token supervision without evaluating the full vocabulary, we retain the student’s K highest-probability next-token candidates. Let $\mathcal{V}_t = \text{TopK}(p_t, K)$ and $\bar{p}_t(v) = p_t(v) / \sum_{u \in \mathcal{V}_t} p_t(u)$ for $v \in \mathcal{V}_t$. For each candidate, we construct the token-level signal

$$A_{t,v}^{\text{OPD}} = \bar{p}_t(v) (\log q_t(v) - \log p_t(v)). \quad (2)$$

A positive signal increases the probability of a candidate favored more strongly by the teacher, whereas a negative signal suppresses a candidate over-preferred by the student. We treat these signals as detached token-level advantages during optimization. The corresponding unclipped surrogate is

$$\widehat{\ell}_t^{\text{OPD}}(\theta) = - \sum_{v \in \mathcal{V}_t} \text{sg}[A_{t,v}^{\text{OPD}}] \log \pi_\theta(v | h_t), \quad (3)$$

where sg denotes stop-gradient. This is the policy-gradient surrogate used to optimize the top- K approximation of reverse KL. We use it for reasoning OPD, while Section 3.3 defines the separate nonnegative discrepancy used for triggering.

3.2. DTDD Overview

DTDD routes supervision in three stages (Figure 2). First, the student generates an on-policy rollout partitioned into fixed-length segments. A prespecified controller, configured for the desired intervention regime, selects the first segment whose aggregated student–teacher divergence crosses its threshold. Second, DTDD rewinds to the preceding boundary and asks the frozen teacher to regenerate the remaining suffix from the retained student prefix. Third, the student prefix receives OPD supervision, while the teacher suffix receives behavior cloning. If no crossing occurs, the entire rollout receives standard OPD.

The controller setting determines the intervention dose. Frequent takeover supplies coherent trajectory-level guidance when a base student has not acquired a teacher-compatible reasoning pattern, whereas sparse takeover repairs localized failures in otherwise effective OPD. In multi-turn tasks, the same routing logic operates at action boundaries, where the teacher generates from the triggering state before interaction continues.

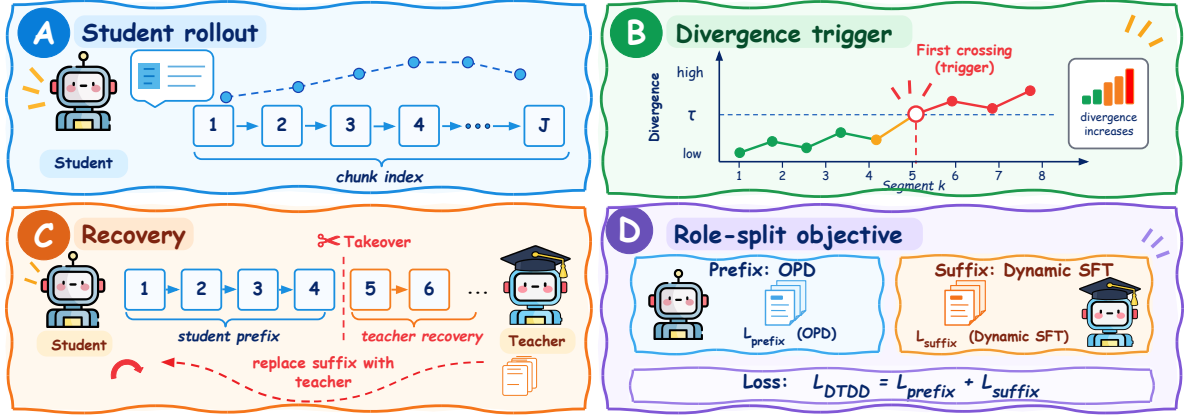


Figure 2 | Reasoning realization of DTDD. (A) The student generates a segmented rollout. (B) The first threshold crossing selects the rewind boundary. (C) The teacher regenerates the suffix from the retained student prefix. (D) The prefix is trained with OPD and the suffix with behavior cloning; without a crossing, DTDD reduces to OPD.

3.3. Segment-Level Trigger and Recovery

We now define the segment discrepancy, intervention boundary, and recovery trajectory. Algorithm 1 summarizes the reasoning realization.

We construct the trigger from a nonnegative token-level discrepancy that captures the magnitude of local disagreement rather than its direction. In our reasoning experiments, it is computed from the same student top- K comparisons as the sparse OPD estimator:

$$\delta_t = \sum_{v \in \mathcal{V}_t} \bar{p}_t(v) |\log p_t(v) - \log q_t(v)|. \quad (4)$$

Let $(W_1, \dots, W_J)$ be an ordered partition of the model-generated positions into *segments*. We define the discrepancy of segment j as

$$d_j = \frac{1}{|W_j|} \sum_{t \in W_j} \delta_t. \quad (5)$$

For a reasoning rollout, we use fixed-length segments of C tokens: $J = \lceil L/C \rceil$ and $W_j = \{(j-1)C + 1, \dots, \min(jC, L)\}$. For an agentic trajectory, each W_j instead contains the tokens of the j -th model-generated action; environment observations provide context but are not included in the discrepancy.

Let $j_{\min}$ be the first eligible segment and τ_s the threshold at optimization step s . DTDD selects the first eligible threshold crossing, $j^* = \min \{j \geq j_{\min} : d_j > \tau_s\}$. If no eligible segment crosses the threshold, DTDD leaves the trajectory unchanged and applies standard OPD. Averaging divergence within a segment prevents isolated token-level spikes from causing intervention.

We instantiate τ_s with either a fixed or rate-controlled threshold. Both variants use divergence to select which rollouts trigger and the first crossing to determine the intervention boundary. Appendix B.1.1 gives the controller and recovery details.

In reasoning tasks, when a trigger occurs, we define the rewind position $c = (j^* - 1)C$ as the boundary immediately preceding the triggering segment. DTDD then replaces the student suffix $y_{c+1:L}^S$ with a continuation $y_{c+1:\tilde{L}}^T$ generated by the frozen teacher conditioned on $(x, y_{1:c}^S)$ within the remaining generation budget. The resulting hybrid trajectory is $\tilde{y} = [y_{1:c}^S; y_{c+1:\tilde{L}}^T]$. Thus the teacher replaces the divergent segment itself, rather than inheriting it and correcting only later tokens. Once

Algorithm 1 DTDD Training for Reasoning Rollouts

Require: Student π_θ ; frozen teacher π_T ; threshold controller H ; rollout parameters $R, C, j_{\min}$

- 1: **for** $s = 1, \dots, S$ **do**
- 2: Roll out $\mathcal{B} = \{(x_i, y_i^S)\}$ with $y_i^S \sim \pi_\theta(\cdot | x_i)$
- 3: Compute $d_{i,1:j_i}$ by Equations 4–5; $\tau_s \leftarrow H_s(\{d_{i,j}\}_{i,j})$
- 4: **for each** $(x_i, y_i^S) \in \mathcal{B}$ **do**
- 5: $j_i^* \leftarrow$ first $j \geq j_{\min}$ with $d_{i,j} > \tau_s$, if one exists
- 6: **if** no such j_i^* exists **then**
- 7: $(c_i, \tilde{y}_i) \leftarrow (L_i, y_i^S)$ {standard OPD}
- 8: **else**
- 9: $c_i \leftarrow (j_i^* - 1)C$ {rewind}
- 10: Sample $y_i^T \sim \pi_T(\cdot | x_i, y_{i,1:c_i}^S)$; $\tilde{y}_i \leftarrow [y_{i,1:c_i}^S; y_i^T]$ {teacher recovery}
- 11: **end if**
- 12: **end for**
- 13: Update θ using $\mathcal{L}_{\text{DTDD}}$: OPD on student prefixes and behavior cloning on teacher suffixes
- 14: **end for**

control is transferred, the teacher completes the suffix without returning generation to the student. If no trigger occurs, we set $c = L$ and $\tilde{y} = y^S$, so DTDD reduces exactly to OPD.

The recovery horizon follows how errors propagate. In agentic code, later tool calls depend on earlier outputs, so errors can compound. When a student action triggers intervention, that action is discarded and the teacher takes over from that point, generating the replacement action and the rest of the episode. In ALFWorld and WebShop, the teacher replaces only the triggering action, and the student resumes at the next turn from the resulting environment state. We use this shorter recovery horizon because each executed action yields a new observation that provides a natural point for returning control to the student. Teacher-written tokens receive behavior cloning in agentic code, as in reasoning. ALFWorld and WebShop instead reuse their sampled-token OPD objective for both student- and teacher-written actions, weighting each token by its teacher–student log-probability gap. Appendix B.2 gives the corresponding objectives.

3.4. Role-Split Training Objective

For a recovered rollout, DTDD applies OPD to the retained student prefix and behavior cloning to the teacher-generated suffix. Behavior cloning is standard token-level cross-entropy: each teacher-written token is the target, conditioned on the preceding hybrid trajectory. Let $\hat{\ell}_t^{\text{OPD}}(\theta)$ be the sparse OPD surrogate from Equation 3. The suffix loss is $\ell_t^{\text{BC}}(\theta) = -\log \pi_\theta(\tilde{y}_t | x, \tilde{y}_{<t})$, and the combined objective is

$$\mathcal{L}_{\text{DTDD}}(\theta) = \mathbb{E} \left[\frac{1}{\tilde{L}} \left(\sum_{t=1}^c \hat{\ell}_t^{\text{OPD}}(\theta) + \lambda_{\text{BC}} \sum_{t=c+1}^{\tilde{L}} \ell_t^{\text{BC}}(\theta) \right) \right], \quad (6)$$

where λ_{BC} weights suffix supervision. If no trigger occurs, $c = L$ and $\tilde{y} = y^S$, so the objective reduces to OPD. We use $\lambda_{\text{BC}} = 1$ in the main experiments. Section 4.1 (Table 2) varies this weight from 0.1 to 3 and also evaluates a soft cross-entropy target.

4. Experiments

We evaluate DTDD across long-form reasoning and multi-turn agents, asking whether takeover improves long-horizon stability (Q1), base-model learning efficiency (Q2), transfer to interactive

tasks (Q3), and which student–teacher regimes support intervention (Q4).

4.1. Experimental Setup

Evaluation settings. We evaluate DTDD in two complementary regimes. The reasoning suite tests long-horizon stability by distilling DeepSeek-R1-Distill-Qwen-1.5B from JustRL-DeepSeek-1.5B on DAPO-Math-17K (DeepSeek-AI, 2025; Yu et al., 2025a), and base-initialization efficiency by pairing Qwen3-1.7B-Base with Qwen3-4B-Base-GRPO on DAPO-Math-17K and with a Qwen3-4B non-thinking teacher on OpenThoughts3 (Guha et al., 2025; Yang et al., 2025). Both math settings follow the experimental configurations in Li et al. (2026b). The interactive suite tests action-level takeover on ALFWorld (Shridhar et al., 2020) with a Qwen2.5-3B-Instruct student (Yang et al., 2024) and GiGPO-7B teacher (Feng et al., 2025), and on the code subset of Open-AgentRL-30K with a Qwen3-1.7B-SFT student and DemyAgent-GRPO-4B teacher.

Comparison and metrics. Vanilla OPD is the direct baseline. We also evaluate AdaSwitch (Peng et al., 2025) in the base-student setting (Table 1; Appendix D.2). Within each setting, paired OPD and DTDD runs share the student initialization, teacher, prompts, batch order, generation budget, and optimizer, providing a controlled comparison of the intervention controller. For training-dynamics and ablation claims, we compare fixed shared updates or the full learning curves. Table 3 reports the best observed checkpoint for each method, selected by Average within its block. Appendix Table 7 reports shared-update comparisons; Appendices C.1 and D.1 summarize the corresponding reasoning curves. We report average@16 accuracy on AIME 2024, AIME 2025, and AMC 2023 for math; success on the seen and unseen splits for ALFWorld; and average@8 execution accuracy on LiveCodeBench (Jain et al., 2025) and held-out in-distribution problems for agentic code. Appendix B specifies the full training, sampling, checkpoint-selection, and reporting protocols.

Design controls. Table 1 ablates two choices in the base-student setting: the triggering mechanism and the teacher-suffix loss. We cross segment-averaged divergence and token-spike triggers with BC (cross-entropy on teacher tokens) and KD (the top- K OPD surrogate on both prefix and suffix), yielding four configurations. DTDD pairs segment-averaged triggering with BC; the other three change the trigger, suffix loss, or both. The token-spike controls use a history-normalized z-score trigger. We additionally evaluate AdaSwitch (Peng et al., 2025) as a separate baseline, pairing unified KD with its published relative-threshold rule: takeover occurs when token divergence exceeds three times the mean over the preceding ten tokens. At step 120, DTDD reaches 20.83%, compared with 10.23–16.79% for the other three and 19.95% for AdaSwitch. The suffix-loss variants retain the early accuracy gain across weights and target types (Table 2). Offline SFT followed by OPD provides a complementary warm-up baseline. Appendix D.2 reports the individual seeds, trigger details, and controller costs. Among the online variants in Table 1, DTDD achieves the highest accuracy at all four checkpoints. At update 20, AdaSwitch leads (16.35% versus 12.65%); the two-seed DTDD mean is higher from update 40 onward (Appendix D.2).

4.2. Q1: Does DTDD Stabilize Long-Horizon OPD?

Long generation budgets expose a late OPD instability that DTDD suppresses. We hold the student, teacher, data, sampling, and optimizer fixed while varying the maximum rollout length over 7,168, 15,360, and 18,432 tokens. We quantify late-training instability by the maximum gradient norm from update 51 onward, after the short-lived optimization transient at the start of training.

Table 1 | Trigger and suffix-loss controls ablations on DAPO-Math-17K.

Method	Suffix loss	Optimizer step				Tok. (M)
		60	80	100	120	
OPD	–	12.14	14.00	13.15	14.54	0
DTDD	BC	20.32	21.47	21.08	20.83	0.52
Token spike	KD	12.47	12.14	12.81	13.51	0.40
Token spike	BC	13.19	13.07	13.26	16.79	0.43
Segment	KD	5.78	7.80	8.04	10.23	0.67
AdaSwitch	KD	18.16	19.57	18.68	19.95	0.86
SFT + OPD	–	20.63	22.81	22.45	22.03	0 [‡]

Accuracy is average@16 (%). OPD and DTDD report two-seed means; other methods use one run. Triggers: segment-averaged (DTDD/Segment), z-score (Token spike), or AdaSwitch’s relative-threshold rule. BC: behavior cloning on teacher tokens. KD: the top- K OPD surrogate on both prefix and suffix. Tok.: teacher-written tokens per update, in millions. [‡]SFT uses an additional 95M teacher-written tokens offline.

The 7K control remains stable under both methods, with peak gradient norms of 0.54 for OPD and 0.96 for DTDD. At 15K, OPD enters a sharp late event: its gradient norm reaches 12.02 and its truncated-rollout ratio reaches 52%. DTDD limits the same quantities to 1.40 and 32%, reducing the gradient peak by 8.6 $\times$. The 18K run reproduces the pattern, with gradient peaks of 12.81 for OPD and 1.84 for DTDD (7.0 $\times$). Held-out accuracy remains within one point across all three budgets. Thus, DTDD removes the long-budget optimization failure while preserving the performance level reached by OPD. Appendix C provides the complete entropy, truncation, overlap, reward, and accuracy trajectories.

Clipping OPD’s token-level advantages to $[-1, 1]$ also suppresses the late spikes at both long budgets, with held-out accuracy within 1.1 points of the unclipped reproduction (Appendix C.2).

Table 2 | Teacher-suffix target and weight ablations.

Suffix loss	λ_{BC}	Optimizer step			
		60	80	100	120
BC	1	20.16	21.34	21.25	20.76
<i>Loss-weight ablation</i>					
BC	3	21.15	21.48	19.91	19.10
BC	0.1	20.94	20.31	22.40	22.29
<i>Target ablation</i>					
Soft CE	1	21.27	22.01	19.95	20.59

Accuracy is average@16 (%), one run per variant. BC: behavior cloning with one-hot cross-entropy on teacher tokens. Soft CE: cross-entropy with soft teacher targets. λ_{BC} weights the suffix loss. The highlighted default uses DTDD seed 1; Table 1 reports the two-seed mean.

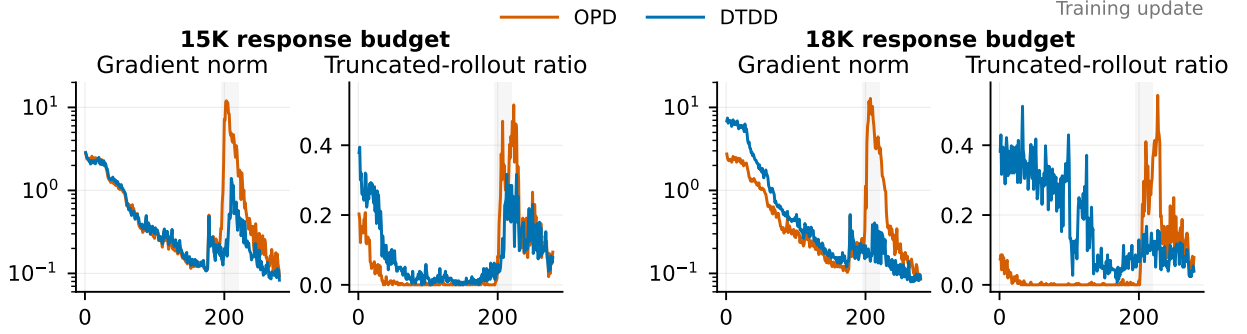


Figure 3 | DTDD suppresses late OPD spikes on DAPO-Math-17K at 15K/18K response budgets.

4.3. Q2: Can Takeover Warm Up a Base Student Online?

DTDD can warm up a base student online, without first preparing an SFT dataset and warm-starting the student on it. Figure 4 compares a Qwen3-1.7B-Base student with a Qwen3-4B-Base-GRPO teacher on DAPO-Math-17K. High-dose DTDD reaches the 21.3% held-out reference level near update 70, whereas OPD reaches it near update 180. The trajectories later converge near 22% average accuracy. The time to match OPD’s update-240 accuracy varies across window sizes (Appendix Table 12). Appendix D reports the complete dose, window-size, and cost analyses.

We measure intervention frequency by the *takeover rate*: the fraction of sampled rollouts in which

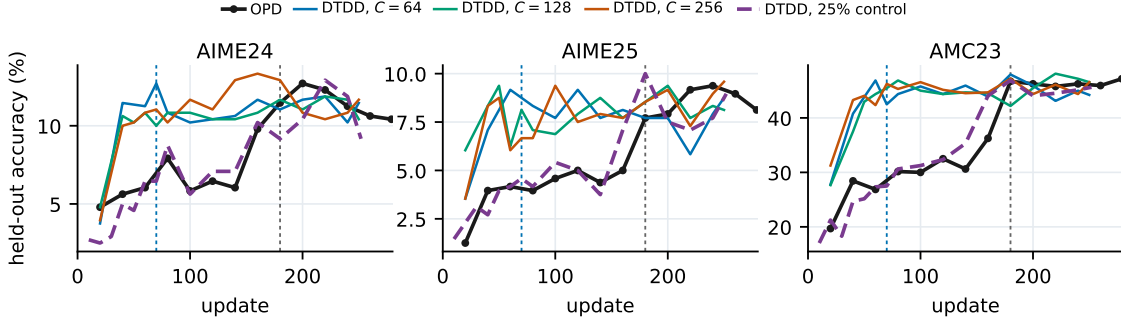


Figure 4 | Held-out accuracy for Qwen3-1.7B-Base ← Qwen3-4B-Base-GRPO on DAPO-Math-17K. The length-gated control starts near 25% takeover and adapts thereafter.

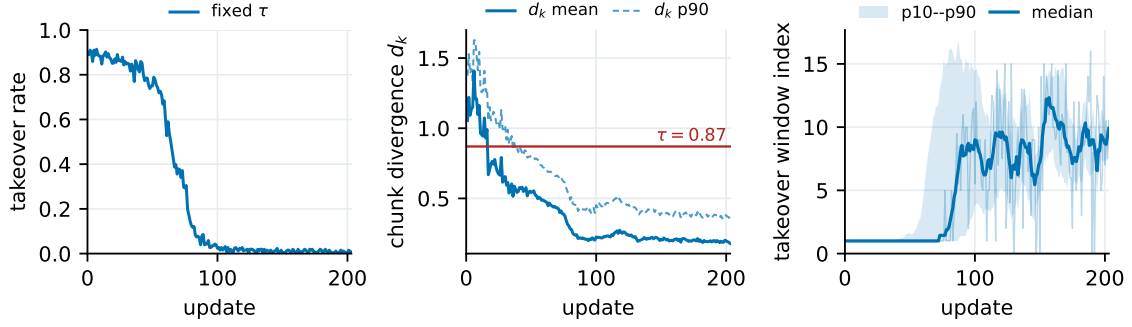


Figure 5 | Fixed-threshold dynamics ($\tau = 0.87$) for Qwen3-1.7B-Base ← Qwen3-4B on OpenThoughts3.

divergence triggers a teacher takeover. *Rewind* replaces the divergent suffix; *restart* replaces the entire rollout. Fixed thresholds allow takeover rates to vary with divergence, whereas target-rate controllers adjust τ to maintain a prescribed rate.

Figure 4 compares OPD with four rewind configurations. Three form a window-size ablation: $C \in \{64, 128, 256\}$, with fixed thresholds $\tau \in \{0.475, 0.362, 0.296\}$ calibrated to match an initial takeover rate of approximately 90%. The fourth fixes $\tau = 0.2$ at $C = 256$ and makes the third segment ($j_{\min} = 3$) the first eligible for takeover. This control starts near 25% takeover, providing limited teacher supervision early in training. As model rollouts become longer, more of them satisfy the trigger; the resulting increase in takeover rate supplies additional teacher supervision.

At update 120, the three high-initial-takeover arms reach 21.32% ($C = 64$), 20.90% ($C = 128$), and 21.23% ($C = 256$), whereas the fixed- $\tau = 0.2$ control remains close to OPD (14.81% versus 14.65%). A separate rate-controlled restart setting targeting 45% takeover reaches 20.49%. By update 240, all four rewind configurations reach 21.04–22.40%, comparable to OPD at 22.29%. Thus, for a base student without prior SFT warm-up, sufficient coherent teacher continuations at a high initial dose accelerate online warm-up; segment-level divergence routes them online and lets fixed-threshold intervention recede with alignment.

The three high-coverage DAPO arms—excluding the low-initial-takeover $\tau = 0.2$ control—show decreasing segment divergence and teacher-written token share, together with later takeover positions (Appendix Figure 11). Figure 5 illustrates these dynamics on OpenThoughts3.

With $\tau = 0.87$, the OpenThoughts3 takeover rate falls from 88.3% to below 5% by update 87 and to zero by update 150, while the median trigger moves from window 1 to 10.5. Among rollouts

that still trigger, the later position retains a longer student prefix before teacher continuation. At matched update 200, the three OpenThoughts3 controllers reach 14.36–15.75% accuracy, compared with 12.85% for OPD (Appendix Table 7).

On DAPO-Math-17K, DTDD reaches 21.34% held-out accuracy in 22.0 hours; offline SFT followed by OPD reaches 21.55% in approximately 42.8 serial hours, including teacher-data generation and SFT. DTDD uses 43M versus 95M teacher-written tokens, allocated online within one training stage. Appendix D.3 reports the checkpoint-level timing and hardware details.

4.4. Q3: Does Divergence-Guided Routing Transfer to Agentic Tasks?

In multi-turn tasks, DTDD triggers teacher recovery at action boundaries, where each action contains a reasoning span and tool call. Takeover is used only during training; evaluation uses the student alone.

On ALFWorld, we pair a LoRA-adapted Qwen2.5-3B-Instruct student with the ALFWorld-specific GiGPO-7B teacher and set $\tau = 0.7$. As in the reasoning experiments, training begins with high teacher coverage (90.2% takeover). As the student improves over training, takeover falls to near zero, automatically reducing teacher supervision. Near update 180, a transient rise in discrepancy reactivates takeover despite its previously near-zero rate (Appendix Figure 12). DTDD therefore also acts as an online monitor and recovery mechanism: it remains mostly inactive after alignment but resumes corrective teacher supervision when instability emerges. With checkpoints evaluated every 20 updates, DTDD leads OPD at all ten matched checkpoints after update 200 in this run. Best-observed average success increases from 76.49% to 81.32%, with the largest gain on the unseen split (Table 3). Appendix E.1 reports the full trajectory and adaptation controls, including the matched update-400 comparison.

The agentic code experiment examines intervention dose for an SFT-initialized policy, pairing Qwen3-1.7B-SFT with DemyAgent-GRPO-4B on the code-only subset of Open-AgentRL-30K (Yu et al., 2025b). Our main setting uses $\tau = 0.50$, yielding a sparse 2.8% teacher-token dose. At their best-observed checkpoints, sparse takeover improves all four evaluation splits and raises aggregate accuracy from 36.49% to 40.59% (Table 3). This result complements the base-student study: high coverage supplies a missing reasoning pattern, whereas sparse correction preserves the on-policy interaction of an already viable agent. Appendix E.2 reports matched endpoints and the dose ablation.

4.5. Q4: What Determines Whether Teacher Takeover Helps?

Headroom is an empirical diagnostic of the observed regimes. We define it as $h = 1 - \text{OPD}_{\text{end}}/\text{teacher}$, with both scores measured on the same evaluation harness. Across 23 arms from 13 settings, 19 follow a common pattern: gains concentrate in regimes where OPD retains more headroom below the teacher (Figure 6).

Three controls refine this diagnostic. In the rebuilt WebShop environment, the task-specific teacher falls below OPD, so takeover routes training toward an invalid supervision source. Conversely, the 0.5B ALFWorld student retains substantial teacher headroom but does not reliably learn the recovery, exposing a student-learnability constraint. Finally, the agentic-code dose ablation shows that excessive intervention can replace too much on-policy interaction. Appendix F reports the complete settings, per-arm results, and TCOD comparison.

Table 3 | Best-observed results (%; selection details in Appendix B.3). Subscripts show percentage-point changes from OPD; bold marks the best trained method.

Math reasoning: student Qwen3-1.7B-Base; teacher Qwen3-4B (non-thinking); data OpenThoughts3					
Method	AIME24	AIME25	AMC23	Average ↑	
Teacher	21.04	20.62	67.66	36.44	
OPD	5.62	5.42	27.50	12.85	
DTDD fixed $\tau = 0.87$	6.88 ^{+1.26}	7.08 ^{+1.66}	31.41 ^{+3.91}	15.12 ^{+2.27}	
DTDD rewind, target 90%→5%	7.29 ^{+1.67}	4.79 ^{-0.63}	32.19 ^{+4.69}	14.76 ^{+1.91}	
DTDD restart, target 90%→5%	8.75 ^{+3.13}	5.21 ^{-0.21}	33.28 ^{+5.78}	15.75 ^{+2.90}	
Agentic interaction: student Qwen2.5-3B-Instruct (LoRA); teacher GiGPO-7B; data ALFWorld					
Method	Seen	Unseen	Average ↑		
Teacher	85.71	88.06	86.89		
OPD	83.57	69.40	76.49		
DTDD	84.29 ^{+0.72}	78.36 ^{+8.96}	81.32 ^{+4.83}		
Agentic code: student Qwen3-1.7B-SFT; teacher DemyAgent-GRPO-4B; data Open-AgentRL-30K (code only)					
Method	HO-200 full	HO-200 sub	LCB v1	LCB v6	Average ↑
Teacher	62.12	64.19	68.78	47.43	60.63
OPD	31.00	30.56	50.91	33.48	36.49
DTDD	35.81 ^{+4.81}	37.62 ^{+7.06}	53.65 ^{+2.74}	35.27 ^{+1.79}	40.59 ^{+4.10}

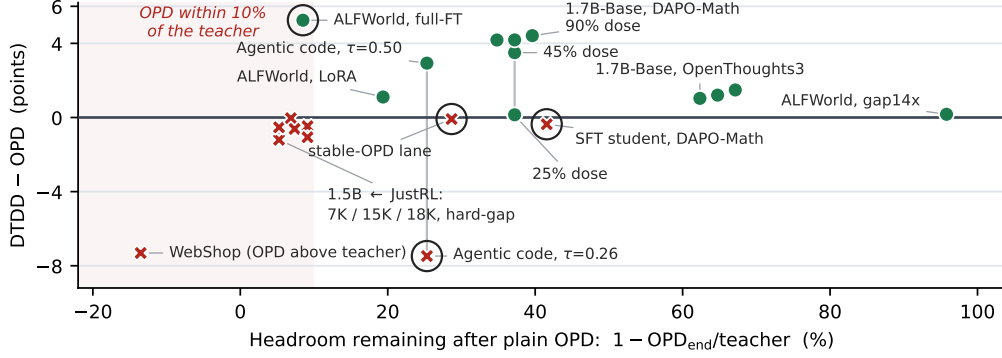


Figure 6 | DTDD minus OPD, averaged over all matched checkpoints, against the headroom plain OPD leaves below its teacher. Points left of zero are settings where OPD ends above the teacher’s own score. Ringed markers identify arms analyzed by the validity, learnability, and dose controls in Appendix F.

5. Conclusion and Limitations

DTDD uses segment-level student–teacher divergence to route supervision between standard OPD and teacher-generated recovery. Across paired experiments, sparse takeover suppresses long-budget optimization extremes, high initial teacher coverage accelerates learning from a base initialization, and action-level takeover improves ALFWorld and agentic code. Controlled comparisons favor segment-level triggering with behavior cloning. Relative to offline SFT followed by OPD, this single-stage approach reaches comparable accuracy while cutting measured serial time by a factor of 1.9 and using 55% fewer teacher-written tokens. DTDD uses manually specified thresholds or target takeover rates. Its divergence trigger does not assess whether teacher recovery is useful for the student. Automatic dose calibration and recovery-quality assessment are directions for future work.

References

- R. Agarwal, N. Vieillard, Y. Zhou, P. Stanczyk, S. Ramos Garea, M. Geist, and O. Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.
- DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- L. Feng, Z. Xue, T. Liu, and B. An. Group-in-group policy optimization for LLM agent training. In *Advances in Neural Information Processing Systems*, 2025. URL <https://arxiv.org/abs/2505.10978>.
- Y. Fu, H. Huang, K. Jiang, J. Liu, Z. Jiang, Y. Zhu, and D. Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.
- Y. Gu, L. Dong, F. Wei, and M. Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.
- E. Guha et al. OpenThoughts: Data recipes for reasoning models. *arXiv preprint arXiv:2506.04178*, 2025.
- Q. He, J. Cheng, C. Le, R. Wang, X. Liu, Y. Chen, J. Mei, Z. Wang, X. Chen, Y. Chen, and T. Wang. ARCHER: Adaptive recovery routing with conformal-risk calibration for coding agents. In *COLM 2026 Workshop on Efficient Reasoning*, 2026. URL <https://openreview.net/forum?id=y8w7HIaLtu>.
- G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- N. Jain, K. Han, A. Gu, W.-D. Li, F. Yan, T. Zhang, S. Wang, A. Solar-Lezama, K. Sen, and I. Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025.
- L. Jiang, H. Xu, Y. Ding, and A. Zhang. Trajectory-refined distillation. *arXiv preprint arXiv:2606.08432*, 2026.
- Y. Kim and A. M. Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, 2016.
- N. Lauffer, X. Deng, S. Kundurthy, B. Kenstler, and J. Da. Imitation learning for multi-turn LM agents via on-policy expert corrections. *arXiv preprint arXiv:2512.14895*, 2025.
- C. Le, J. Cheng, Q. He, R. Li, Y. Li, and X. Chen. What does multi-harness RL learn? credit assignment and portability in coding agents. *arXiv preprint arXiv:2609.04518*, 2026. URL <https://arxiv.org/abs/2609.04518>.
- G. Li, M. Zheng, M. Song, R. Liu, T. Yang, J. Sun, Q. Zhong, H. Guo, J. Fang, D. Zhang, and J. Wang. On-policy distillation with curriculum turn-level guidance for multi-turn agents. *arXiv preprint arXiv:2606.15912*, 2026a.
- Y. Li, Y. Zuo, B. He, J. Zhang, C. Xiao, C. Qian, T. Yu, H.-a. Gao, W. Yang, Z. Liu, and N. Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026b.

- X. Liu, K. Jiao, C. Xiao, R. Zhao, J. Ruan, B. Li, J. Liu, Q. Wang, X. Chen, J. Wang, C. Wang, T. Xiao, and J. Zhu. Teacher-guided policy optimization for on-policy reasoning distillation under large policy divergence. *arXiv preprint arXiv:2605.13230*, 2026.
- F. Luo, Y.-N. Chuang, G. Wang, Z. Xu, X. Han, T. Zhang, and V. Braverman. Demystifying OPD: Length inflation and stabilization strategies for large language models. *arXiv preprint arXiv:2604.08527*, 2026.
- J. Peng, M. Wang, H. Cai, Y. Li, K. Zhang, S. Wang, D. Yin, and X. Zhao. AdaSwitch: Balancing exploration and guidance in knowledge distillation via adaptive switching. *arXiv preprint arXiv:2510.07842*, 2025.
- M. Shridhar, X. Yuan, M.-A. Côté, Y. Bisk, A. Trischler, and M. Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*, 2020.
- C. Sun, L. Liu, H. Lei, T. Ling, J. Xie, Z. Zheng, Y. Wang, H. Liu, F. Xiao, L. Liu, Y. Du, Z. Cheng, Z. Jiang, and Q. Gu. Step-level on-policy distillation: Interpolating between on-policy distillation and supervised fine-tuning. *arXiv preprint arXiv:2608.16333*, 2026.
- J. Wang, W. Zhang, W. Shi, Y. Li, and J. Cheng. TCOd: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. *arXiv preprint arXiv:2604.24005*, 2026.
- Y. Xia, Q. Sun, C. Song, J. Wu, Y. Qi, and Y. Xu. DASH-OPD: Discrepancy-aware switching with hysteresis for on-policy distillation. *arXiv preprint arXiv:2607.29078*, 2026.
- H. Xu, X. Xu, H. Hong, Z. Ni, H. Li, Y. Qiu, W. Lu, and Y. Shen. Pass the baton: Trajectory-relayed on-policy distillation. *arXiv preprint arXiv:2607.26057*, 2026.
- A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- Z. Yang, Z. Guo, Y. Song, M. Xu, Y. Wang, Y. Wang, X. Liang, and J. Tang. Prune-OPD: Efficient and reliable on-policy distillation for long-horizon reasoning. *arXiv preprint arXiv:2605.07804*, 2026.
- Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, W. Dai, T. Fan, G. Liu, L. Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025a.
- Z. Yu, L. Yang, J. Zou, S. Yan, and M. Wang. Demystifying reinforcement learning in agentic reasoning. *arXiv preprint arXiv:2510.11701*, 2025b.
- Q. Zhong, M. Zheng, M. Song, X. Lin, J. Sun, Y. Zhao, X. Wang, H. Jiang, X. Wang, and J. Fang. SOD: Step-wise on-policy distillation for small language model agents. *arXiv preprint arXiv:2605.07725*, 2026.

Appendix

A	Measured Limitations of Vanilla OPD	15
B	Experimental Details and Reporting Protocols	15
B.1	Reasoning Training Protocol	16
B.2	Interactive Training Protocol	16
B.3	Evaluation and Checkpoint Selection	18
B.4	Matched-Checkpoint Comparisons	19
C	Additional Long-Horizon Stability Results	20
C.1	Training Dynamics across Response Budgets	20
C.2	Signal-Clipping Control	20
C.3	Intervention Dynamics and Exploration-Floor Ablation	22
D	Additional Base-Initialization Results	22
D.1	Learning Curves and Matched-Checkpoint Results	22
D.2	Controller Dynamics and Design Ablations	23
D.3	Offline SFT Baseline and Computational Cost	25
E	Additional Agentic Results	27
E.1	ALFWorld Action-Level Intervention	27
E.2	Agentic Code Dose	29
F	Headroom Analysis and Boundary Cases	29
F.1	Headroom Inventory and Reporting Details	29
F.2	Boundary Cases	30
F.3	Reanalysis of TCOB Results	31

A. Measured Limitations of Vanilla OPD

Figure 7 supplements the schematic failures in Figure 1(A–B) with measured OPD learning curves: a controlled comparison of base and SFT initializations, and the gradient trajectory under a long response budget.

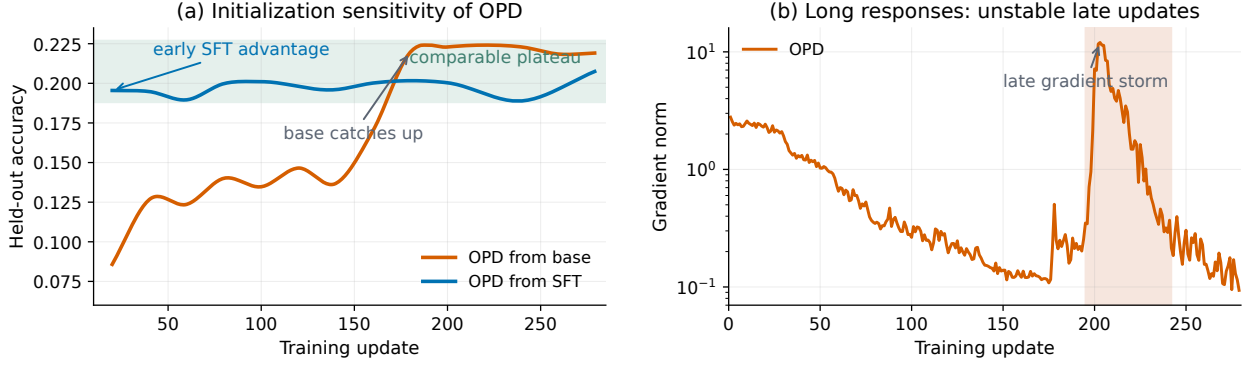


Figure 7 | Two representative limitations of vanilla OPD. *Left*: Qwen3-1.7B-Base and Qwen3-1.7B-SFT are distilled from the same Qwen3-4B-Base-GRPO teacher on DAPO-Math-17K with a 7,168-token generation budget. The teacher, data, optimizer, and generation budget are held fixed; the SFT initialization has a large early advantage, while the base initialization eventually reaches a comparable plateau. Held-out accuracy is the mean avg@16 score on AIME 2024, AIME 2025, and AMC 2023. *Right*: DeepSeek-R1-Distill-Qwen-1.5B is distilled from JustRL-DeepSeek-1.5B on DAPO-Math-17K with a 15,360-token generation budget. OPD develops a large late gradient spike despite otherwise decreasing gradient norms.

B. Experimental Details and Reporting Protocols

This appendix expands Section 4.1 with task-specific generation budgets (Table 4), optimization and controller settings, interactive training objectives, and evaluation protocols.

Table 4 | Experimental settings. The budget specifies the maximum generated tokens for reasoning tasks and the maximum number of environment or tool turns for interactive tasks.

Setting	Student	Teacher	Training data	Budget	Evaluation
Rollout length	R1-Distill-Qwen-1.5B	JustRL-DeepSeek-1.5B	DAPO-Math-17K	7,168 / 15,360 / 18,432 tokens	AIME24/25, AMC23
Base student	Qwen3-1.7B-Base	Qwen3-4B-Base-GRPO	DAPO-Math-17K	7,168 tokens	AIME24/25, AMC23
Base student	Qwen3-1.7B-Base	Qwen3-4B (non-thinking)	OpenThoughts3	7,168 tokens	AIME24/25, AMC23
ALFWorld	Qwen2.5-3B-Instruct	GiGPO-7B	ALFWorld train	30 turns; 512 tokens/action	Seen / unseen success
WebShop	Qwen2.5-3B-Instruct	GiGPO-7B (WebShop)	Rebuilt 120K catalog	15 turns; 512 tokens/action	Success
Agentic code	Qwen3-1.7B-SFT	DemyAgent-GRPO-4B	Open-AgentRL-30K (code only)	8 turns / 12,288 tokens	LCB v1/v6, HO-200 full/sub

B.1. Reasoning Training Protocol

The reasoning comparisons in Sections 4.2 and 4.3 use the shared training protocol below; Table 5 specifies the controller for each arm. Each update samples 64 prompts with four rollouts per prompt. Student and teacher-logit temperatures are both 1.0, and each rollout batch receives one policy update. Optimization uses AdamW with $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 0.01, and global gradient clipping at 1.0. The learning rate is held constant at 10^{-6} without warm-up. Distillation uses the student top- K set with $K = 16$, student-probability weighting, and token-mean aggregation. The teacher is frozen; no sequence-level outcome reward enters the objective, and teacher-written tokens use $\lambda_{\text{BC}} = 1$ unless varied in the suffix-loss ablation. The length sweep and length-gated $\tau = 0.2$ control set $j_{\min} = 3$; the other base-initialization configurations use $j_{\min} = 1$. The 256-token continuation-budget check does not filter any reported trigger.

Both terms in Equation 6 share the same optimization path. The prefix uses the clipped policy-gradient form of $\widehat{\ell}_t^{\text{OPD}}$; the suffix uses a one-hot target, which gives the standard behavior-cloning gradient when clipping is inactive.

Reasoning takeover is retrospective. The student first generates a full rollout, after which DTDD computes segment discrepancies and requests a teacher continuation for each triggered example. The update retains the student prefix and excludes the replaced student suffix. This pipeline changes the training sequence without reducing student rollout compute, so Section 4.3 measures optimization-step efficiency.

B.1.1. Controller and Recovery Details

For rollout-level rewind, we suppress triggers too close to the maximum generation budget to leave at least $B_{\min}$ tokens for teacher continuation. We instantiate τ_s with either a fixed or rate-controlled threshold. The fixed controller holds $\tau_s = \tau$, allowing takeover to recede as student-teacher discrepancy falls. The rate-controlled variant sets τ_s to the empirical $(1 - r_s)$ -quantile of per-rollout maximum eligible segment discrepancies across the batch, where the target takeover rate r_s may be fixed or scheduled. Both variants use divergence to select which rollouts trigger and the first crossing to determine the intervention boundary. In the length sweep, an exploration branch triggers a fraction $\epsilon = 0.05$ of rollouts at a uniformly sampled candidate boundary as an exploration floor. Once segment discrepancy contracts below τ , this floor rather than the divergence signal accounts for most interventions; Section C reports where each length-sweep arm sits relative to the floor. Unless otherwise specified, DTDD uses rewind; we consider full-rollout restart ($c = 0$) only as an ablation.

B.2. Interactive Training Protocol

This section specifies the normalized action-token objectives and optimization settings for the interactive realization in Section 3.3. Let $z_j \in \{S, T\}$ denote whether the student or teacher writes action j , let W_j contain its generated-token positions, and let N_S and N_T count student- and teacher-written tokens. Let $a_{j,t}$ be token t of the action executed at turn j , let $h_{j,t}$ be the interaction history and action prefix before that token, and let π_{old} denote the student that collected the interaction. The sampled-token OPD signal is

$$A_{j,t}^{\text{act}} = \log \pi_T(a_{j,t} \mid h_{j,t}) - \log \pi_{\text{old}}(a_{j,t} \mid h_{j,t}). \quad (7)$$

For student-written actions, define the unclipped sampled-token OPD term $\widehat{\ell}_{j,t}^{\text{OPD}} = -\text{sg}[A_{j,t}^{\text{act}}] \log \pi_{\theta}(a_{j,t} \mid h_{j,t})$. The implementations differ in the recovery horizon and the loss applied to teacher-written actions, as detailed below.

Table 5 | Reasoning configurations. Target rate is a rollout-level control parameter; the realized rate and teacher-token dose are measured from the generated batches. A dash denotes vanilla OPD.

Study	Budget	Segment	Mode	Controller	Initialization / data
Length sweep	7,168 / 15,360 / 18,432	256 tokens	rewind	fixed $\tau = 0.7$, $\epsilon = 0.05$	DeepSeek-R1-Distill-Qwen-1.5B; DAPO-Math-17K
DAPO base, low initial coverage	7,168	256 tokens	rewind	fixed $\tau = 0.2$; initial takeover $\approx 25\%$	Qwen3-1.7B-Base; DAPO-Math-17K
DAPO base, medium coverage	7,168	256 tokens	restart	target rate 45%	Qwen3-1.7B-Base; DAPO-Math-17K
DAPO base, high initial coverage	7,168	64 / 128 / 256 tokens	rewind	fixed $\tau = .475/.362/.296$; initial takeover $\approx 90\%$	Qwen3-1.7B-Base; DAPO-Math-17K
OpenThoughts3 base	7,168	256 tokens	rewind restart	/ target rate 90% $\rightarrow$ 5%, or fixed $\tau = 0.87$	Qwen3-1.7B-Base; OpenThoughts3

All interactive runs use AdamW with $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 0.01, a constant learning-rate schedule without warm-up, and global gradient clipping at 1.0.

B.2.1. ALFWorld and WebShop

In ALFWorld and WebShop, the teacher replaces the triggering action, and the student resumes at the next turn. Both action types pass through the same sampled-token multi-turn OPD surrogate. With $N = N_S + N_T$, the objective is

$$\hat{\mathcal{L}}_{\text{env}}(\theta) = -\mathbb{E} \left[\frac{1}{N} \sum_j \sum_{t \in W_j} \text{sg} \left[A_{j,t}^{\text{act}} \right] \log \pi_{\theta}(a_{j,t} \mid h_{j,t}) \right], \quad (8)$$

implemented with the clipped surrogate. Student actions are sampled from the rollout policy; teacher actions are sampled from the teacher, making the latter term log-ratio-weighted imitation rather than unit-weight cross-entropy.

Teacher-action supervision. The ALFWorld and WebShop implementation reuses the framework’s existing sampled-token advantage in Equation 7. Its scoring interface returns log-probabilities for the generated tokens. For a teacher-written action, we record both the rollout student’s and the teacher’s log-probabilities on those same tokens. This shared representation allows student and teacher actions to pass through the same trainer. Agentic code uses a separate actor with an explicit behavior-cloning loss (Equation 9).

On a teacher-written token, a positive $A_{j,t}^{\text{act}}$ encourages the student to increase its probability; a negative value reverses this direction. The weight approaches zero as the student and teacher probabilities match. Thus, the environment objective uses a signed log-probability-gap weight, whereas behavior cloning assigns each teacher target a fixed positive weight. Because these tokens are sampled from the teacher, the weighted surrogate is not an unbiased estimator of either KL direction. We have not compared it against unit-weight behavior cloning within ALFWorld or WebShop.

ALFWorld pairs a Qwen2.5-3B-Instruct student with GiGPO-7B for episodes of at most 30 turns and 512 generated tokens per action. A replacement turn includes both reasoning and action, generated from the same interaction history as the discarded student turn. We test both LoRA and full-model

Table 6 | Held-out evaluation protocol.

Domain	Evaluation set	Sampling / metric
Math	AIME24, AIME25, AMC23	16 samples, $T = 0.7$, top- $p = 0.95$; average@16
ALFWorld	140 seen, 134 unseen tasks	Student-only environment success
Code	LCB v1/v6, HO-200 full/sub	8 samples, $T = 1.0$, top- $p = 0.6$; execution accuracy

optimization. The LoRA setting uses rank and scale 32 on all linear layers with learning rate 10^{-4} ; full fine-tuning uses learning rate 10^{-6} . Both use a training batch size of 32, training temperature 1.0, and evaluation temperature 0.4. WebShop uses the same student, LoRA configuration, and WebShop-specific GiGPO-7B teacher on the rebuilt 120K-product catalog, with at most 15 turns, 512 generated tokens per action, and $\tau = 0.25$.

B.2.2. Agentic Code

Our agentic-code implementation builds on the released SOD training stack (Zhong et al., 2026), using the code-only subset of Open-AgentRL-30K and the execution-based evaluation protocol in Appendix B.3. It applies OPD to student-written tokens and behavior cloning to teacher-written tokens, with role-specific normalization:

$$\mathcal{L}_{\text{code}}(\theta) = \mathbb{E} \left[\frac{1}{N_S} \sum_{j: z_j=S} \sum_{t \in W_j} \tilde{\ell}_{j,t}^{\text{OPD}}(\theta) - \frac{\lambda_{\text{BC}}}{N_T} \sum_{j: z_j=T} \sum_{t \in W_j} \log \pi_{\theta}(a_{j,t} | h_{j,t}) \right], \quad (9)$$

where an empty role-specific term is omitted and the OPD term uses its clipped implementation.

Agentic code pairs Qwen3-1.7B-SFT with DemyAgent-GRPO-4B on the code-only portion of Open-AgentRL-30K. Each episode allows eight model turns and 12,288 generated tokens. Takeover occurs online at an action boundary, so no future student tokens are generated after a trigger; the teacher generates all remaining turns. Fixed thresholds 0.50 and 0.26 produce the sparse- and high-dose conditions. Training uses learning rate 10^{-6} , a training batch of 32 prompts with eight rollouts each, a policy minibatch of 16, and $\lambda_{\text{BC}} = 1$. Training and evaluation use temperature 1.0; evaluation uses top- $p = 0.6$ and eight samples per problem.

B.3. Evaluation and Checkpoint Selection

The evaluation summary in Section 4.1 is expanded here with decoding parameters, split sizes (Table 6), and the checkpoints selected for Table 3. Math checkpoints use a 16,384-token generation budget and 16 samples per problem at temperature 0.7 and top- $p = 0.95$; accuracy averages over samples and problems. ALFWorld evaluates the student alone, with no teacher intervention. Execution-harness choices can substantially affect coding-agent performance (Le et al., 2026); we therefore use the same execution sandbox and decoding parameters for all code methods.

Table 3 reports the performance envelope obtained by selecting each arm’s checkpoint by the Average within its block, considering only checkpoints with results for every benchmark in that block. For OpenThoughts3, the selected updates are 200 for OPD, 180 for fixed-threshold DTDD, 180 for scheduled rewind, and 200 for scheduled restart. ALFWorld selects updates 280 and 140 for OPD and DTDD, respectively; agentic code selects updates 80 and 60.

Appendix Table 7 evaluates every trained method within a block at one shared update: 200 for

Table 7 | Matched-checkpoint results (%). All trained methods within each block are evaluated at the same update: 200 for math, 400 for ALFWorld, and 60* for agentic code. Subscripts show the absolute percentage-point change from OPD; bold marks the best trained method. For code, Held-out averages the two HO-200 splits and LCB averages LiveCodeBench v1 and v6.

Math reasoning: student Qwen3-1.7B-Base; teacher Qwen3-4B (non-thinking); data OpenThoughts3				
Method	AIME24	AIME25	AMC23	Average ↑
Teacher	21.04	20.62	67.66	36.44
OPD	5.62	5.42	27.50	12.85
DTDD fixed $\tau = 0.87$	6.88 _{+1.25}	5.42 _{+0.00}	31.25 _{+3.75}	14.51 _{+1.67}
DTDD rewind, target 90% → 5%	8.96 _{+3.33}	4.58 _{-0.83}	29.53 _{+2.03}	14.36 _{+1.51}
DTDD restart, target 90% → 5%	8.75 _{+3.13}	5.21 _{-0.21}	33.28 _{+5.78}	15.75 _{+2.90}
Agentic interaction: student Qwen2.5-3B-Instruct (LoRA); teacher GiGPO-7B; data ALFWorld				
Method	Seen	Unseen		Average ↑
Teacher	85.71	88.06		86.89
OPD	70.71	69.40		70.06
DTDD	84.29 _{+13.57}	70.90 _{+1.49}		77.59 _{+7.53}
Agentic code: student Qwen3-1.7B-SFT; teacher DemyAgent-GRPO-4B; data Open-AgentRL-30K (code only)				
Method	Held-out	LCB		Average ↑
Teacher	63.16	58.11		60.63
OPD	26.03	41.65		33.84
DTDD (2.8% dose)	36.72 _{+10.69}	44.46 _{+2.81}		40.59 _{+6.75}
DTDD (~ 50% dose)	27.31 _{+1.28}	37.43 _{-4.22}		32.37 _{-1.47}

*Agentic code is compared at update 60 rather than at its final update, 130. By update 50, OPD’s aggregate accuracy has stopped improving. It stays between 33.2% and 36.5% over updates 50–130 and is 33.84% at update 60 versus 33.93% at update 130, so later checkpoints add no OPD gain. Update 60 is also the best checkpoint for sparse DTDD. After it, sparse DTDD drifts downward, but it still leads OPD by 1.03 points at update 130 (Table 19).

OpenThoughts3, 400 for ALFWorld, and 60 for agentic code. Stability and boundary analyses use matched checkpoints, and efficiency analyses use the first evaluated checkpoint that reaches the specified performance level. Every curve represents one training run; average@ k refers to k evaluation generations. Checkpoints from the same run are correlated and are not treated as independent training replicates; average@ k measures evaluation-time sampling variation rather than training-seed uncertainty.

Stability statistics use the maximum gradient norm after update 50, applying one cutoff across generation budgets. Truncation ratio measures the fraction of rollouts that reach the maximum budget. Takeover rate measures how often an intervention occurs, while teacher-token dose measures how much of the training sequence the teacher writes. Takeover-position statistics condition on triggered examples; each figure specifies whether it reports a mean or median.

B.4. Matched-Checkpoint Comparisons

Table 7 complements the best-observed results in Table 3 by evaluating each block at a shared update. Learning-curve summaries accompany the long-horizon stability, DAPO base-initialization, and OpenThoughts3 results in Appendices C.1 and D.1.

The learning-curve summaries distinguish endpoint performance from gains earlier in training. Last Δ is the DTDD–OPD accuracy difference at the final update evaluated for both methods. Last-five Δ averages the paired differences over the final five shared checkpoints; all-shared Δ averages them over all N shared evaluation checkpoints. All differences are measured in percentage points of held-out average@16 accuracy. The OPD late range is its maximum minus minimum accuracy over the final five shared checkpoints. It describes variation within one training run, including any learning trend over that window.

C. Additional Long-Horizon Stability Results

C.1. Training Dynamics across Response Budgets

Figures 8–10 extend Figure 3 with a 7K stable control and additional entropy, teacher-overlap, train-batch reward, and takeover-rate curves. At 7K, OPD and DTDD follow almost the same entropy, gradient, truncation, teacher-overlap, and train-batch reward trajectories (Figure 8).

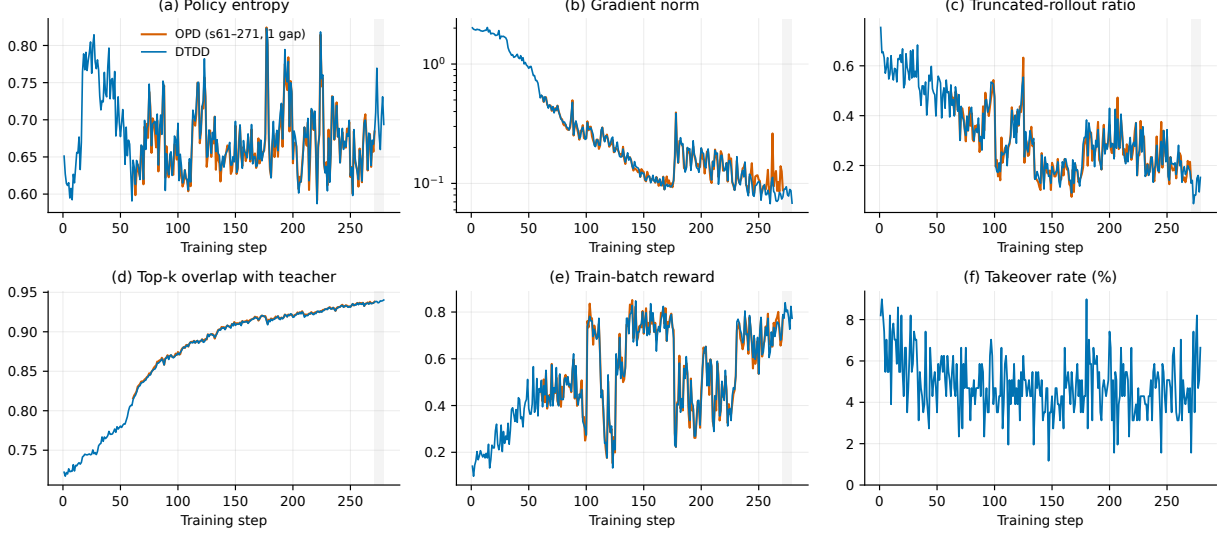


Figure 8 | Complete 7K training dynamics for DeepSeek-R1-Distill-Qwen-1.5B distilled from JustRL-DeepSeek-1.5B on DAPO-Math-17K. This setting serves as the stable control. The OPD telemetry begins at update 61; held-out comparisons use checkpoints available for both methods.

At 15K, OPD destabilizes near update 200: gradient norm and entropy rise, truncation increases, and teacher top- K overlap drops together (Figure 9). These additional metrics show that the gradient event in Figure 3 coincides with changes in rollout behavior, not just the update magnitude.

The 18K run reproduces the coupled event (Figure 10), with entropy and truncation rising as teacher overlap falls. These runs have limited accuracy headroom: the JustRL teacher scores 58.45% and OPD reaches 54.2–55.4%, leaving $h = 5.3$ –7.4%, the smallest positive headroom among the 23 arms in Section 4.5 (Table 20). In this regime, the benefit of DTDD is optimization stability rather than higher accuracy.

The matched-curve summaries give the same accuracy picture: the final-five DTDD–OPD differences are -0.73 and -0.20 points at 15K and 18K, respectively. Averaging over all 10 and 14 shared checkpoints through update 279 gives -0.62 and -1.20 points (definitions in Appendix B.4).

C.2. Signal-Clipping Control

The clipping result in Section 4.2 shows that limiting OPD advantage magnitudes can suppress late instability without teacher takeover. Here we specify the clamp and compare tight ($b = 1$) and loose ($b = 5$) bounds.

At each position t , the advantage $A_{t,v}^{\text{OPD}}$ in Equation 2 weights the teacher–student log-probability gap by the student’s renormalized top- K probability of candidate $v \in \mathcal{V}_t$. We clip each of these candidate-level advantages to $[-b, b]$:

$$A_{t,v}^{(b)} = \min\{b, \max\{-b, A_{t,v}^{\text{OPD}}\}\}, \quad b \in \{1, 5\}. \quad (10)$$

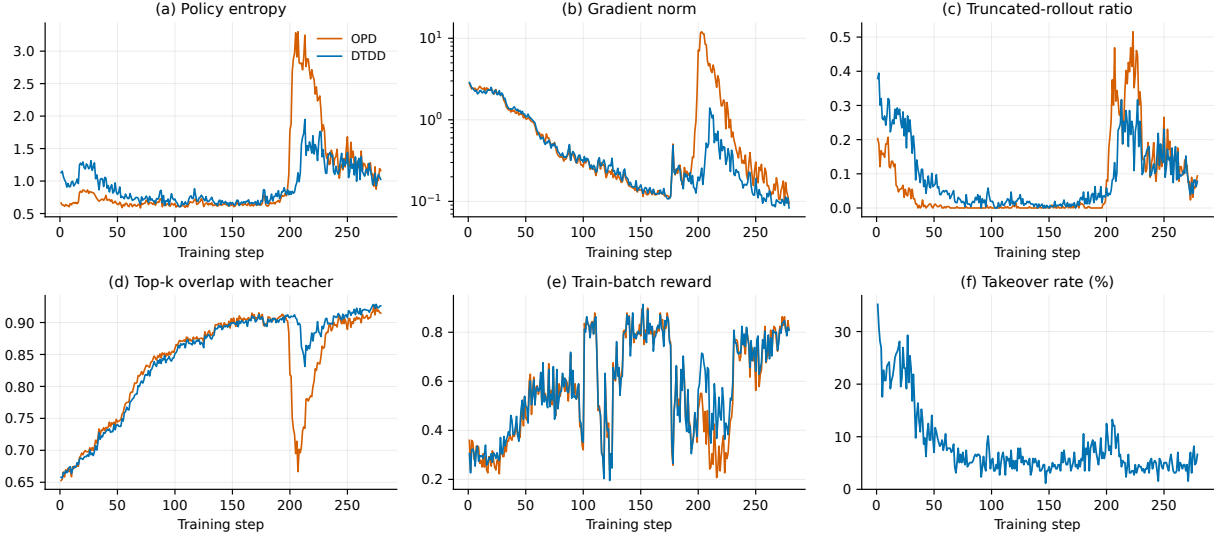


Figure 9 | Complete 15K training dynamics. The OPD gradient and entropy spikes coincide with increasing truncation and decreasing teacher overlap. DTDD limits each component of this event while preserving held-out accuracy.

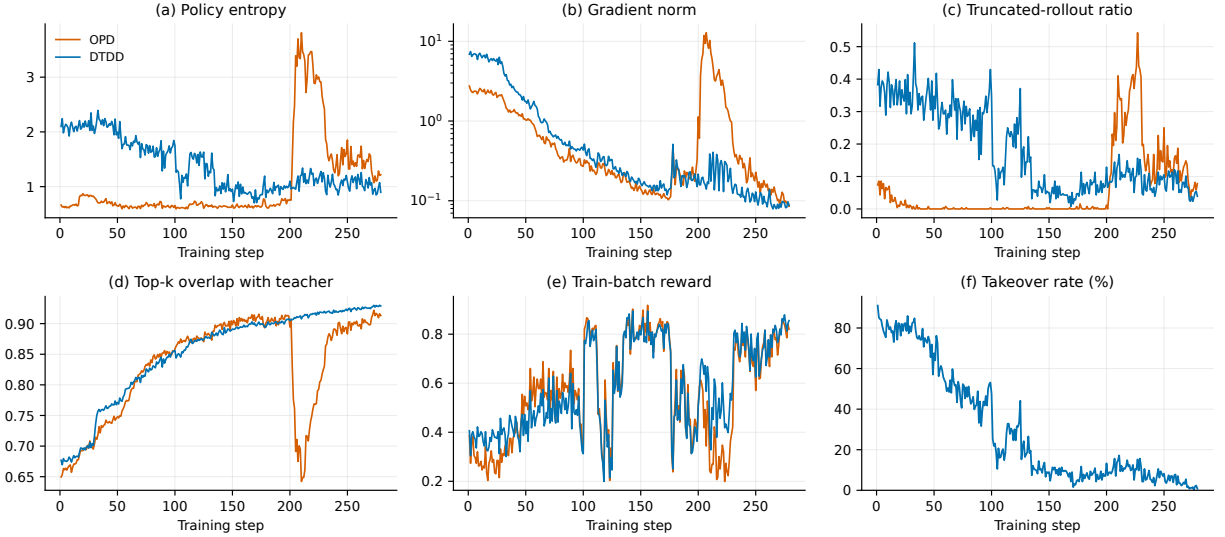


Figure 10 | Complete 18K training dynamics. DTDD suppresses the late OPD instability in gradient norm, entropy, truncation, and teacher overlap. Its intervention rate decreases over training and remains responsive to the late discrepancy increase.

Values inside the interval remain unchanged; larger magnitudes are capped at b while retaining their sign. We substitute $A_{t,v}^{(b)}$ into the OPD surrogate in Equation 3, keeping rollout generation, policy-ratio clipping, and global gradient clipping unchanged. The clamp therefore acts on the probability-weighted advantages before loss aggregation.

Table 8 shows that the tight $b = 1$ clamp suppresses the late gradient spikes at both budgets, whereas $b = 5$ leaves the peak and the number of high-gradient updates nearly unchanged. At update 279, the tight clamp changes held-out accuracy by +0.55 and -1.08 points relative to the unclipped OPD reproduction at 15K and 18K, respectively.

Table 8 | Clipped-OPD control at update 279. Accuracy is held-out average@16 (%); gradient statistics exclude the first 50 updates and are measured before global gradient clipping.

Budget	OPD signal	Peak gradient	Updates > 2	Accuracy
15K	Unclipped reproduction	14.33	22	54.50
15K	Clamp $[-5, 5]$	12.48	21	55.26
15K	Clamp $[-1, 1]$	0.95	0	55.05
18K	Unclipped reproduction	12.42	21	54.62
18K	Clamp $[-5, 5]$	11.28	21	53.72
18K	Clamp $[-1, 1]$	0.96	0	53.54

C.3. Intervention Dynamics and Exploration-Floor Ablation

For the length sweep in Section 4.2, we additionally examine intervention frequency, trigger timing in OPD shadow runs, and the effect of removing the random exploration floor.

Intervention dynamics. The three length-sweep arms share $\tau = 0.7$ and a 5% random exploration floor, but the divergence branch activates differently across budgets. At 7K, the realized takeover rate remains between 2% and 8%, making this run a no-harm control for sparse intervention. At 15K, takeover starts at 35%, recedes to the floor by update 100, and reactivates around the late instability. At 18K, takeover starts at 91% and remains divergence-driven for most of training before responding again to the late discrepancy increase. The 15K and 18K runs therefore test adaptive triggering, while 7K establishes the stable control.

Trigger timing. To examine when the trigger responds, OPD shadow runs evaluate the DTDD trigger on student rollouts without applying takeover or changing the training objective. At 18K, the would-trigger rate stays at 8–10% through update 198, rises to 25% at update 201, and the gradient spike begins at update 204. Policy entropy leaves its preceding range at update 192. In this run, divergence reacts near the optimization event rather than providing an earlier warning than entropy.

Removing the exploration floor. The length controller includes a 5% random-intervention floor. We fork the 18K run at update 170 and remove this branch. Across six evaluated checkpoints, the matched held-out accuracy changes by only -0.05 point. The random floor therefore has negligible effect on student-only accuracy in this regime.

D. Additional Base-Initialization Results

D.1. Learning Curves and Matched-Checkpoint Results

This section extends Figure 4 and Table 3 with per-controller and per-benchmark results at shared checkpoints, learning-curve summaries, and updates needed to reach additional accuracy targets.

DAPO-Math matched checkpoints. Table 9 reports per-controller results at updates 120 and 240, including the 45% restart arm not plotted in Figure 4. The individual endpoints extend the accuracy range summarized in Section 4.3.

Table 9 | Base-initialization dose comparison on DAPO-Math-17K (average@16, %). Updates 120 and 240 provide common early and late checkpoints for all reported arms.

Method	Window C	Controller	Update 120	Update 240
OPD	–	–	14.65	22.29
DTDD rewind	256	fixed $\tau = 0.2$, length-gated (initial takeover $\approx 25\%$)	14.81	21.58
DTDD restart	–	target rate 45%	20.49	21.67
DTDD rewind	64	fixed $\tau = 0.475$ (initial takeover $\approx 90\%$)	21.32	21.04
DTDD rewind	128	fixed $\tau = 0.362$ (initial takeover $\approx 90\%$)	20.90	22.40
DTDD rewind	256	fixed $\tau = 0.296$ (initial takeover $\approx 90\%$)	21.23	21.39

For the high-initial-coverage rewind run with $C = 128$, the mean held-out accuracy gain over OPD is 4.18 percentage points across 12 shared checkpoints through update 240, compared with 0.56 over the final five and 0.10 at the last checkpoint. OPD’s maximum-minus-minimum accuracy over the final five checkpoints is 5.40 points (reporting definitions in Appendix B.4). The larger all-shared gain and smaller late-window gain reflect faster early learning rather than a higher final plateau.

Per-benchmark results and OpenThoughts3 controls. Table 11 complements the best-observed OpenThoughts3 results in Table 3 with per-benchmark scores at the shared update 200. It also adds the per-benchmark DAPO-Math-17K results at update 240 for selected rewind and restart controls from Section 4.3.

Table 10 extends the checkpoint comparison in Table 3 to the final five and all shared OpenThoughts3 checkpoints. All three controller gains remain positive under these summaries as well as at the last shared checkpoint, update 200. The final-five gains range from 1.47 to 1.68 points, supporting a sustained late-training improvement beyond the selected best checkpoints.

Table 10 | OpenThoughts3 controller gains over OPD through update 200. Accuracy differences and OPD ranges are in percentage points; summaries are defined in Appendix B.4.

DTDD controller	N	Last Δ	Last-five Δ	All-shared Δ	OPD late range
Fixed $\tau = 0.87$	10	+1.67	+1.68	+1.49	1.46
Rewind, target 90% $\rightarrow$ 5%	10	+1.51	+1.47	+1.03	1.46
Restart, target 90% $\rightarrow$ 5%	10	+2.90	+1.53	+1.21	1.46

Accuracy targets and learning speed. Table 12 extends the 21.3% reference-level comparison in Figure 4 to two higher accuracy targets: OPD’s update-240 scores of 22.29% for seed 0 and 22.75% for the two-seed mean. It reports the first evaluated update reaching each target. Among the rewind runs, $C = 64$ and $C = 256$ reach 22.29% at updates 60 and 100, respectively, compared with update 200 for OPD seed 0. The $C = 128$ run reaches this level at update 220. At the 22.75% target, $C = 256$ and the 45% restart arm cross at updates 180 and 220, while $C = 64$ and $C = 128$ remain below the target through update 240. Thus, the window-size sweep shows consistent early accuracy gains, while the time to match OPD’s endpoint depends on the controller and target accuracy.

D.2. Controller Dynamics and Design Ablations

This section supplements Figure 4 and Table 1 with realized intervention rates, controller dynamics, trigger and loss details, and individual-seed results.

Table 11 | Per-benchmark base-initialization results (average@16, %) at shared update 240 for DAPO-Math-17K and 200 for OpenThoughts3.

Method	AIME24	AIME25	AMC23	Average
<i>Qwen3-1.7B-Base $\leftarrow$ Qwen3-4B-Base-GRPO, DAPO-Math-17K, 7,168 tokens</i>				
Teacher: Qwen3-4B-Base-GRPO	24.17	18.12	64.22	35.50
OPD	11.25	9.38	46.25	22.29
DTDD restart, target rate 45%	11.25	7.50	46.25	21.67
DTDD rewind, initial takeover $\approx$ 90%, $C = 128$	11.67	8.33	47.19	22.40
DTDD fixed $\tau=0.2$, initial takeover $\approx$ 25%	11.88	7.71	45.16	21.58
<i>Qwen3-1.7B-Base $\leftarrow$ Qwen3-4B-Base (non-thinking), OpenThoughts3, 7,168 tokens</i>				
Teacher: Qwen3-4B-Base	21.04	20.62	67.66	36.44
OPD	5.62	5.42	27.50	12.85
DTDD rewind, target rate 90% $\rightarrow$ 5%	8.96	4.58	29.53	14.36
DTDD restart, target rate 90% $\rightarrow$ 5%	8.75	5.21	33.28	15.75
DTDD fixed $\tau=0.87$	6.88	5.42	31.25	14.51

Table 12 | First evaluated update reaching each accuracy level on DAPO-Math-17K (average@16, %). The two OPD-derived targets use update 240; “–” denotes a target not reached by that update. *After update 120, OPD seed 1 is evaluated every 40 updates. Crossing times reflect the available evaluation checkpoints.

Arm	$\geq 21.3\%$ (reference)	$\geq 22.29\%$ (OPD seed 0)	$\geq 22.75\%$ (OPD mean, $n = 2$)	Best (step)
OPD, seed 0	180	200	–	22.41 (220)
OPD, seed 1*	240	240	240	23.21 (240)
DTDD rewind, $C = 64$	60	60	–	22.43 (60)
DTDD rewind, $C = 128$	80	220	–	22.57 (220)
DTDD rewind, $C = 256$	70	100	180	22.88 (180)
DTDD restart, 45%	100	100	220	22.97 (220)
SFT + OPD	20	80	80	22.81 (80)

Realized intervention rates. Realized coverage differs from the initial or target rate used to label each arm. The length-gated control rises from approximately 25% to 94% rollout-level takeover as rollouts lengthen, whereas the 45% rate-controlled restart arm realizes approximately 37% after eligibility filtering.

Window-size dynamics. Figure 11 complements the accuracy curves in Figure 4 with controller measurements for $C \in \{64, 128, 256\}$: takeover rate, normalized segment divergence, teacher-written token fraction, and takeover position. These measurements separate how often takeover occurs from how much of each triggered rollout the teacher writes. The normalized divergence curves also show how student–teacher discrepancy evolves relative to each controller’s fixed threshold.

Trigger and suffix-loss controls. Table 13 expands Table 1 with evaluations at updates 20 and 40 and the individual OPD and DTDD seeds underlying its two-seed means. The following analyses add implementation details, intervention statistics, runtime, and seed-to-seed variation.

The suffix-loss controls in Table 1 also differ in their supervision support. In the segment-plus-unified-KD row, the reverse-KL surrogate is evaluated on the student’s top- K support, which can exclude the token written by the teacher. This arm therefore measures support mismatch as well as the

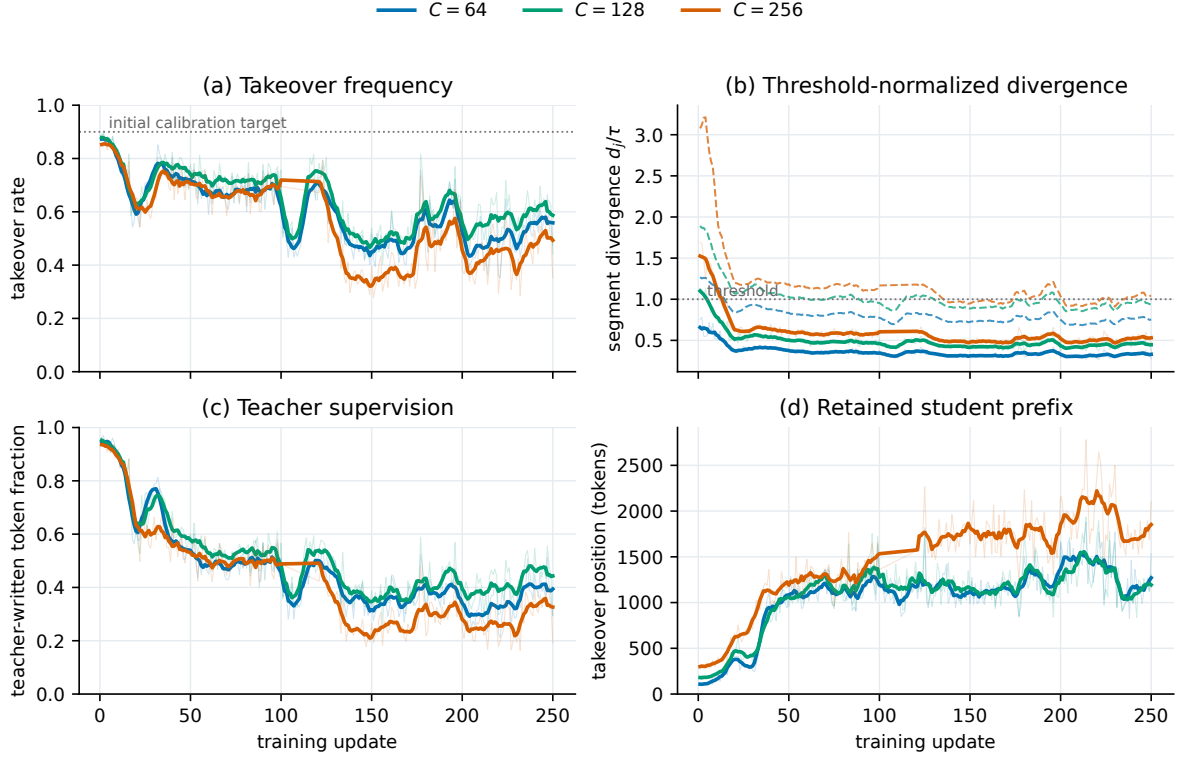


Figure 11 | Controller dynamics for the three high-coverage DAPO-Math rewind arms. Faint curves show per-update values and solid curves show seven-update moving averages. **(a)** Rollout-level takeover rate; the dotted line is the initial calibration target. **(b)** Segment divergence normalized by each arm’s fixed threshold (solid: mean; dashed: 90th percentile). **(c)** Fraction of generated training tokens written by the teacher. **(d)** Mean takeover position, conditional on triggering and converted from window index to token position.

choice between a unified and role-split objective. The controllers also differ in realized intervention position and teacher-token dose (Table 14). We therefore interpret Table 13 as a comparison of complete controller designs, not an isolated causal estimate of trigger granularity or suffix objective.

Table 14 extends the teacher-token column of Table 1 with trigger frequency, median trigger position, and runtime. In these later-training measurements, AdaSwitch (Peng et al., 2025) triggers on every rollout, with a median trigger at token 14; DTDD triggers on 71% of rollouts, with a median trigger at segment 5.

Training-seed replication. Table 15 adds sample standard deviations ($n - 1$ in the denominator) to the two-seed means in Table 1, and includes the earlier evaluations at updates 20 and 40. From update 60 onward the seed-to-seed standard deviation is at most 0.45 points for both methods, against a gap of 6.3–8.2 points between them.

D.3. Offline SFT Baseline and Computational Cost

This section extends the SFT comparison in Table 1 and Section 4.3 with earlier and later checkpoints, stage-level timing, and hardware records.

Table 13 | Base-student controller and objective controls on DAPO-Math-17K (held-out average@16, %). A dash denotes an unscheduled evaluation.

Trigger / method	Teacher-suffix objective	Optimizer step					
		20	40	60	80	100	120
OPD (seed 0)	–	8.58	12.67	12.36	14.01	13.47	14.65
Segment, DTDD (seed 0)	BC	12.83	18.82	20.49	21.60	20.90	20.90
OPD (seed 1)	–	9.69	10.36	11.91	13.99	12.83	14.43
Segment, DTDD (seed 1)	BC	12.47	19.65	20.16	21.34	21.25	20.76
Token spike	Unified KD	8.84	9.53	12.47	12.14	12.81	13.51
Token spike	BC	8.61	10.31	13.19	13.07	13.26	16.79
Segment	Unified KD	6.89	4.77	5.78	7.80	8.04	10.23
AdaSwitch ($K = 3$)	Unified KD	16.35	18.94	18.16	19.57	18.68	19.95

Table 14 | Later-training controller and runtime measurements for the base-student controls. Trigger position is reported in each controller’s native granularity. Runtime follows each completed job’s allocation; GPU models are shown where recorded, and times are not hardware-normalized.

Controller / objective	Median trigger	Trigger rate	Teacher-written tokens/update	to- Time/update
DTDD (segment + BC)	segment 5	71%	0.52M	15.1 min (4×A100); 19.5 min (4×L40S)
Token spike + unified KD	token 91	79%	0.40M	10.2 min
Token spike + BC	token 88	82%	0.43M	13.0 min
Segment + unified KD	segment 1	80%	0.67M	14.2 min
AdaSwitch ($K = 3$)	token 14	100%	0.86M	31.3 min
OPD	–	–	0	5.4 min (2×A100)

Table 15 | Two-seed mean $\pm$ SD on DAPO-Math-17K (average@16, %).

Method	Optimizer step					
	20	40	60	80	100	120
OPD	9.13 $\pm$ 0.79	11.52 $\pm$ 1.63	12.14 $\pm$ 0.32	14.00 $\pm$ 0.01	13.15 $\pm$ 0.45	14.54 $\pm$ 0.16
DTDD	12.65 $\pm$ 0.25	19.23 $\pm$ 0.59	20.32 $\pm$ 0.23	21.47 $\pm$ 0.18	21.08 $\pm$ 0.25	20.83 $\pm$ 0.10

Learning curves. Table 16 extends the comparison at updates 60–120 in Table 1 to updates 20–240. The direct OPD and DTDD rows report seed 0, while the offline SFT baseline is the same run as in the main table. SFT followed by OPD reaches 21.55% at its first evaluation, update 20, and ranges from 20.6% to 22.8% through update 240. Most of its observed gain is present by the first evaluated OPD checkpoint.

Comparison with direct OPD. The per-update timings in Table 14 complement the update-based comparison in Figure 4. Later-training updates take 5.4 minutes for OPD on two A100 GPUs, compared with 15.1 minutes for DTDD on four A100 GPUs and 19.5 minutes on four L40S GPUs. These measurements use different resource allocations and are reported as observed rather than hardware-normalized costs.

End-to-end pipeline cost. Table 17 decomposes the serial-time comparison in Section 4.3 into data generation, SFT, checkpoint merging, and online training, and adds accelerator use. The offline route

Table 16 | Fixed-update comparison of direct OPD, online takeover, and offline SFT warm start (held-out average@16, %). A dash denotes an unscheduled checkpoint.

Training route	Optimizer step								
	20	40	60	80	100	120	160	200	240
OPD from base	8.58	12.67	12.36	14.01	13.47	14.65	17.01	22.29	22.29
DTDD from base	12.83	18.82	20.49	21.60	20.90	20.90	21.02	21.96	22.40
SFT → OPD	21.55	21.27	20.63	22.81	22.45	22.03	22.31	21.67	21.77

Table 17 | Serial cost to comparable observed checkpoints. Training time sums logged step times. Stage times are rounded; totals use unrounded measurements. Accelerator types differ across routes.

Route	Measured stages	Checkpoint (ac- curacy)	Serial time	GPU-h	Teacher-written to- kens
Online (seed 1)	DTDD online training (22.0 h)	Step 80 (21.34)	22.0 h	88	43M
SFT → OPD	teacher data (26.9 h) + SFT (11.7 h) + merge (0.1 h) + OPD (4.0 h)	Step 20 (21.55)	≈42.8 h	≈66	95M

spends 26.9 hours generating teacher data and 11.7 hours on SFT before OPD begins. Measured accelerator use is approximately 66 GPU-hours for the offline route and 88 GPU-hours for online DTDD. Generation, SFT, and checkpoint merging use completed-job elapsed times; OPD and DTDD use cumulative logged step times through updates 20 and 80, respectively. These costs exclude queue delays and failed launches. Totals use unrounded measurements. These wall-clock values use the hardware assigned to each completed job—A40 and A100 GPUs for the SFT route, and four L40S GPUs for DTDD—and are not normalized across device types.

E. Additional Agentic Results

E.1. ALFWorld Action-Level Intervention

Table 18 extends the ALFWorld comparison in Table 3 to the final shared update, 400, and adds a full-model training control alongside LoRA. At update 400, LoRA improves average success from 70.06% to 77.59%, driven primarily by the seen split. Full-model training raises OPD to 79.50% and narrows the DTDD advantage to about 0.4 point (Table 18). Parameter-efficient adaptation therefore leaves a larger recoverable gap, while full-model OPD closes most of that gap on its own.

Table 18 | ALFWorld success rate (%) at update 400, the last common checkpoint. Full-model training provides a capacity control for LoRA.

Method	Seen	Unseen	Average
<i>Qwen2.5-3B-Instruct</i> ← <i>GiGPO-7B</i> , LoRA			
OPD	70.71	69.40	70.06
DTDD	84.29	70.90	77.59
<i>Qwen2.5-3B-Instruct</i> ← <i>GiGPO-7B</i> , full fine-tuning			
OPD	82.14	76.87	79.50
DTDD	82.14	77.61	79.88

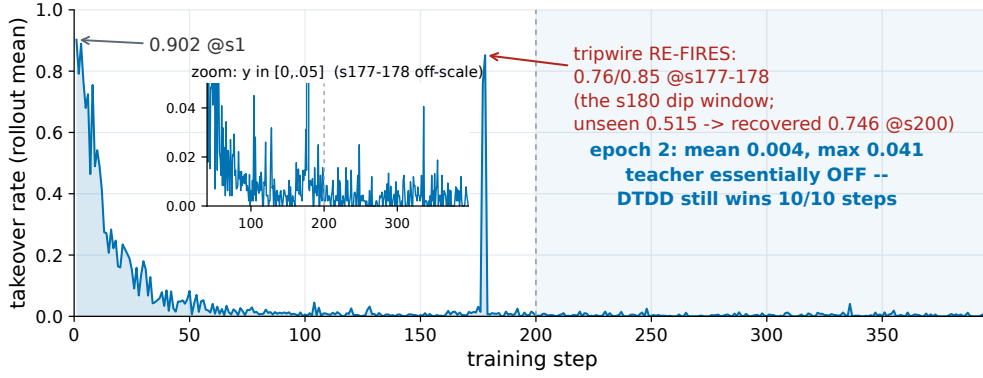


Figure 12 | ALFWorld action-level takeover (Qwen2.5-3B-Instruct $\leftarrow$ GiGPO-7B). The controller withdraws as discrepancy contracts and reactivates around the update-180 disruption.

SB.1 ALFWorld LoRA v2 DTDD (turn-level, $\tau=0.7$) — 397 logged steps, run ended at s397

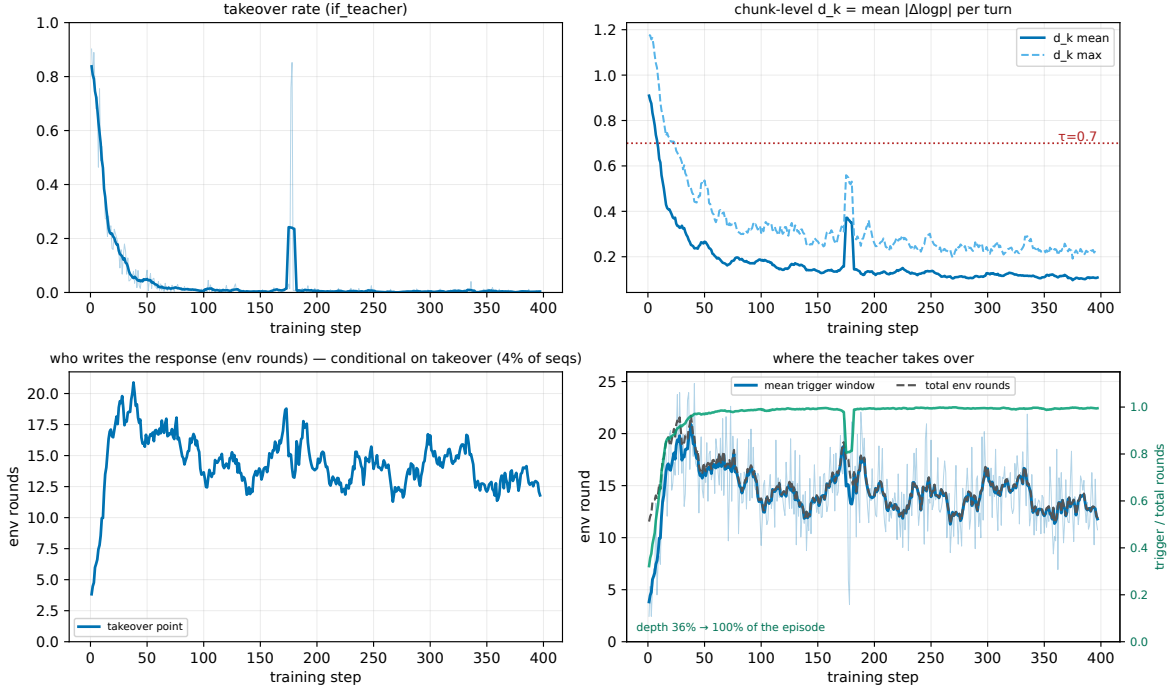


Figure 13 | ALFWorld action-level controller dynamics. Top: takeover rate and action discrepancy. Bottom: intervention turn and its depth within the episode, conditional on takeover. The student is Qwen2.5-3B-Instruct, the teacher is GiGPO-7B, and the maximum episode length is 30 turns.

Across the last five matched checkpoints, DTDD leads by 7.22 points on average. The OPD arm varies by 10.2 points over the same window, so repeated runs are needed to separate the intervention effect from late-training variation.

Figures 12 and 13 provide the controller measurements behind the ALFWorld analysis in Section 4.4. The first resolves the low takeover rates and brief reactivation; the second adds action discrepancy, intervention turn, and relative episode depth, showing where the remaining interventions occur.

Table 19 | Agentic code execution accuracy (average@8, %) at update 130. Held-out averages the two in-distribution splits; LCB averages LiveCodeBench v1 and v6.

Method	Dose	Held-out	LCB	Avg.
<i>Qwen3-1.7B-SFT $\leftarrow$ DemyAgent-GRPO-4B, code interaction</i>				
Teacher	–	63.16	58.11	60.63
OPD	0%	25.60	42.26	33.93
DTDD ($\tau = 0.50$)	2.8%	31.50	38.43	34.96
DTDD ($\tau = 0.26$)	~50%	19.78	36.48	28.13

E.2. Agentic Code Dose

Table 19 complements the best-observed code results in Table 3 with the final shared update, 130, and a higher-dose control ($\tau = 0.26$) for the same student–teacher pair. At the shared update-130 endpoint, takeover with $\tau = 0.50$ remains 1.03 points above OPD in aggregate because of gains on the two in-distribution splits, although both LiveCodeBench scores trail OPD (Table 19).

The two thresholds deliver 2.8% and approximately 50% teacher-token dose. Increasing the dose to approximately 50% reverses the gain over OPD, reducing aggregate accuracy from 33.93% to 28.13% at update 130. The teacher itself reaches 60.63% on the same aggregate evaluation, so teacher quality leaves ample headroom. The contrasting outcome points to excessive replacement of a warm-start student’s on-policy interaction.

F. Headroom Analysis and Boundary Cases

This appendix supplements Figure 6 and Section 4.5 with the per-arm measurements underlying the headroom plot, detailed boundary experiments, and a reanalysis of published TCOD results.

F.1. Headroom Inventory and Reporting Details

Table 20 provides the teacher scores, OPD endpoints, paired-checkpoint counts, and mean gains for all 23 arms from the 13 settings in Figure 6. Headroom follows the definition in Section 4.5; gains average the DTDD–OPD difference over the matched checkpoints available for each arm. This differs from the best-observed protocol of Table 3: the ALFWorld LoRA arm, for example, has 19.4% headroom and a +1.11-point matched-checkpoint mean here, against a +4.83-point best-observed gap in Table 3. Within each baseline group, the OPD endpoint is the last checkpoint shared by OPD and all full-length DTDD arms. The shorter 25%-target control and the 18K exploration-floor ablation use the same group headroom but contribute only their available matched checkpoints to the mean gain. Each included arm has at least three paired evaluations.

The analysis is descriptive: arms that share a setting are not independent, and the 10% boundary summarizes the observed split rather than defining a fitted decision threshold. The split summarizes the observed sign rather than the magnitude. Six DAPO-Math arms share one headroom value but range from +0.11 to +4.42 points as dose and intervention form change.

Four arms fall outside the common split. ALFWorld full fine-tuning averages a 5.25-point gain across 20 matched checkpoints but finishes only about 0.4 point above OPD, indicating transient acceleration. Both axes for the code points use LiveCodeBench v1. The $\tau = 0.26$ arm loses 7.48 points when its LiveCodeBench-v1 gap is averaged over 12 matched checkpoints. The –5.80-point result in Table 19 instead compares OPD and DTDD at their shared update-130 endpoint after averaging HO-200 full, HO-200 subset, LiveCodeBench v1, and LiveCodeBench v6. The same arm has an

Table 20 | Complete inventory for Figure 6. Scores and headroom h are percentages; mean Δ is the DTDD–OPD difference in percentage points across N matched checkpoints. Step identifies the OPD endpoint used to compute headroom. Math uses average@16, ALFWorld and WebShop use success rate, and code uses LiveCodeBench v1 average@8.

Arm	Teacher	OPD endpoint	Step	h (%)	N	Mean Δ
<i>DAPO-Math: R1-Distill-1.5B student / JustRL-1.5B teacher</i>						
7K rewind	58.45	54.44	260	6.86	10	−0.04
15K rewind	58.45	54.15	279	7.37	10	−0.62
18K rewind	58.45	55.38	279	5.26	14	−1.20
18K, no exploration floor	58.45	55.38	279	5.26	6	−0.53
Hard-gap, fixed threshold	58.45	53.12	150	9.12	7	−0.47
Hard-gap, scheduled restart	58.45	53.12	150	9.12	8	−1.07
<i>DAPO-Math: Qwen3-1.7B-Base student / Qwen3-4B-Base-GRPO teacher</i>						
25%-target control	35.50	22.29	240	37.21	6	+0.11
Fixed $\tau = 0.2$, length-gated	35.50	22.29	240	37.21	12	+0.15
45%-target restart	35.50	22.29	240	37.21	12	+3.50
90% initial, $C = 64$	35.50	22.29	240	37.21	12	+4.19
90% initial, $C = 128$	35.50	22.29	240	37.21	12	+4.18
90% initial, $C = 256$	35.50	22.29	240	37.21	12	+4.42
<i>DAPO-Math: Qwen3-1.7B-SFT student / Qwen3-4B-Base-GRPO teacher</i>						
SFT-initialized student	35.50	20.75	279	41.56	4	−0.36
<i>OpenThoughts3: Qwen3-1.7B-Base student / Qwen3-4B teacher</i>						
Scheduled rewind	36.44	12.85	200	64.75	10	+1.03
Scheduled restart	36.44	12.85	200	64.75	10	+1.21
Fixed $\tau = 0.87$	36.44	12.85	200	64.75	10	+1.49
<i>DAPO-Math: R1-Distill-1.5B student / R1-Distill-7B teacher</i>						
Stable-OPD setting	59.31	42.31	279	28.66	14	−0.08
<i>ALFWorld: Qwen2.5-3B-Instruct student / GiGPO-7B teacher</i>						
LoRA	86.89	70.06	400	19.37	20	+1.11
Full fine-tuning	86.89	79.50	400	8.50	20	+5.25
<i>ALFWorld: Qwen2.5-0.5B-Instruct student / GiGPO-7B teacher</i>						
Wide teacher–student gap	86.89	3.65	120	95.80	6	+0.17
<i>WebShop: Qwen2.5-3B-Instruct student / GiGPO-7B teacher</i>						
Rebuilt 120K catalog	18.69	21.21	180	−13.51	9	−7.31
<i>Agentic code: Qwen3-1.7B-SFT student / DemyAgent-GRPO-4B teacher</i>						
Sparse, $\tau = 0.50$	68.78	51.37	130	25.31	11	+2.93
High-dose, $\tau = 0.26$	68.78	51.37	130	25.31	12	−7.48

85.4% rollout-level takeover rate (approximately 50% teacher-token dose), showing that excessive replacement can erase otherwise useful headroom. The “SFT student” point is Qwen3-1.7B-SFT $\leftarrow$ Qwen3-4B-Base-GRPO on DAPO-Math-17K; the “stable-OPD lane” is DeepSeek-R1-Distill-Qwen-1.5B $\leftarrow$ DeepSeek-R1-Distill-Qwen-7B on DAPO-Math-17K. Their matched-checkpoint means differ from OPD by −0.36 and −0.08 point, respectively.

F.2. Boundary Cases

The teacher-validity and student-learnability controls summarized in Section 4.5 are detailed below, including the environment changes and matched evaluations. Table 21 collects their settings and quantitative outcomes alongside the dose controls.

Table 21 | Boundary cases and their empirical interpretation. Scores use the native evaluation metric of each environment.

Condition	Setting	Observation	Result	Interpretation
Little remaining gap	7K/15K/18K distilled math student	OPD finishes within 5–9% of teacher	Stability gain; accuracy tie	Stability without higher accuracy
Valid, learnable gap	Base math student	OPD learns slowly; teacher continuation is learnable	High dose reaches the reference score earlier	Faster early learning
Warm, healthy student	Agentic code, $\rho_T = 2.8\%$	Small in-distribution gain	+1.03 points at update 130	Sparse intervention can help
Excessive dose	Agentic code, $\rho_T \approx 50\%$	Teacher behavior replaces on-policy interaction	−5.80 aggregate points at update 130	Excessive replacement
Invalid teacher	Rebuilt WebShop	Teacher 18.69% < OPD 21.21%	DTDD 11.62% at update 180	Teacher mismatch
Limited learnability	ALFWorld, 0.5B $\leftarrow$ 7B	Matched mean: OPD 0.79%, DTDD 0.96%	+0.17 points; both remain near zero	Limited student transfer

Teacher validity. The rebuilt WebShop environment uses a 120K-product catalog that retains all goal-bearing products and adds random distractors. It therefore differs from the catalog and observation format on which the WebShop-specific GiGPO-7B teacher was trained. On this environment, that teacher reaches 18.69% success. At the last shared checkpoint, update 180, OPD reaches 21.21% and DTDD reaches 11.62%: takeover routes training toward a teacher that is weaker on the current environment.

Student learnability. The small-student ALFWorld experiment pairs Qwen2.5-0.5B-Instruct with the ALFWorld-specific GiGPO-7B teacher, creating a roughly 14 $\times$ parameter gap. Across the six matched evaluations from updates 20 through 120, OPD and DTDD average 0.79% and 0.96% success, respectively. At update 120, the scores are 3.65% and 1.80%; the additional DTDD evaluation at update 140 reaches 4.71%. The teacher supplies valid behavior, but takeover does not produce reliable student-only success under this training configuration.

Intervention dose. The dose control holds the code student, teacher, and task fixed while varying the takeover threshold. Appendix E.2 reports its realized teacher-token doses and per-split endpoint results.

F.3. Reanalysis of TCO Results

To examine the pattern in Figure 6 beyond our own runs, we apply the same headroom calculation to the published main TCO results (Wang et al., 2026). This separates its two student scales. The 1.7B OPD baselines retain over 99% headroom, and all 18 takeover cells improve by 9.2–24.7 points. The 4B baselines finish within 6.9–13.4% of their teachers; gains averaged across the six curriculum configurations within each benchmark fall to 0.3–1.5 points, with four negative cells. These published results show a similar qualitative pattern to Figure 6. This comparison uses reported scores rather than new TCO runs.