DENSE IS NOT ENOUGH: HIERARCHICAL SUPERVISION ALLOCATION FOR LONG-HORIZON ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) transfers the capabilities of a large language model to a smaller student by providing teacher supervision on the student's own rollouts. In long-horizon agentic tasks, however, uniform token-level matching can allocate supervision poorly: a large local discrepancy need not improve future behavior, while consequential guidance may be beyond the current student's reach or fail to persist without privileged input. We formulate long-horizon OPD as *hierarchical supervision allocation* and argue that productive guidance lies at the intersection of *future utility* and *current learnability*. Crucially, this intersection evolves as the student learns. Based on this principle, we propose **LENS-OPD**, a coarse-to-fine framework that organizes supervision through LOCATE, VALIDATE, and REFINE. LOCATE adapts trajectory exposure to the student's evolving competence and proposes a candidate decision for intervention. VALIDATE tests whether teacher guidance at that decision improves the same student's subsequent behavior. REFINE internalizes the beneficial guided behavior into the deployable policy and concentrates token-level supervision on decisive teacher–student conflicts within the validated turn. These stages are nested: each finer allocation is conditioned on the coarser decision, rather than being optimized as an independent importance score. Experiments across multiple long-horizon agent benchmarks and student–teacher configurations show that LENS-OPD consistently improves task performance over vanilla OPD and strong curriculum- and selection-based baselines. Our results suggest that effective long-horizon distillation requires teaching at the right depth, the right decision, and the right token.

1 INTRODUCTION

On-policy distillation (OPD) transfers the capabilities of a large language model to a smaller student (Gu et al., 2023; Agarwal et al., 2024). Unlike offline distillation on fixed teacher demonstrations, OPD collects student rollouts and queries a frozen teacher on the contexts the student actually visits. It therefore combines on-policy data with dense distributional supervision and reduces exposure bias (Ross et al., 2011; Bengio et al., 2015). This recipe is well suited to single-turn generation, but is incomplete for long-horizon agentic tasks, where each action can change later observations and decisions (Shridhar et al., 2021; Yao et al., 2022; Wang et al., 2022). Effective supervision must therefore decide how far the student should explore, which decision to correct, and what part of the correction it can absorb.

Recent methods approach this problem from two directions. *Trajectory-control methods* regulate rollout depth or alternate teacher and student control to reduce distribution drift and compounding errors (Wang et al., 2026a; Li et al., 2026a; Zhou et al., 2026b). *Selective-supervision methods* allocate teacher guidance using disagreement, confidence, environment feedback, or future continuations (Zhou et al., 2026a; Chen et al., 2026). Both improve upon uniform OPD, but neither jointly determines whether a teacher signal has downstream utility and is learnable by the current student. Local disagreement cannot distinguish these cases, and a global curriculum cannot determine which intervention is productive at a particular state.

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

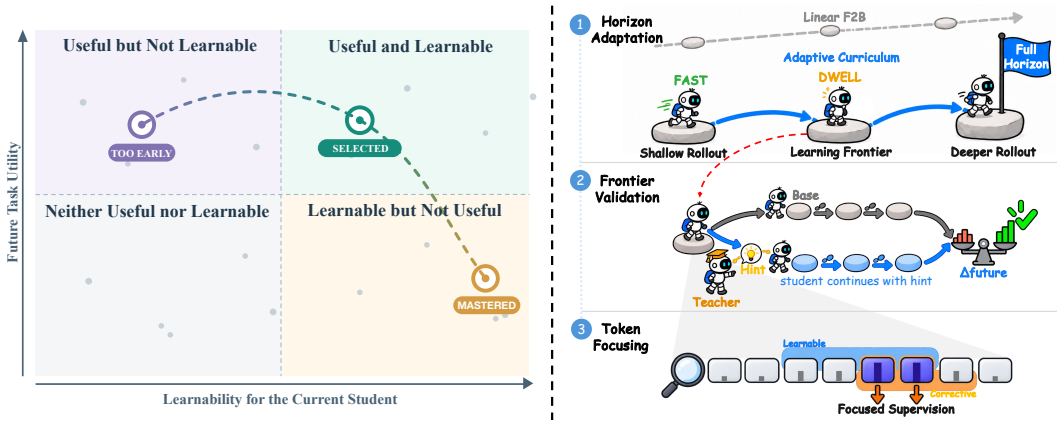


Figure 1: Overview of LENS-OPD. Productive guidance combines future utility and current learnability, while Locate–Validate–Refine allocates supervision from trajectories to turns and tokens.

We therefore view long-horizon OPD as *hierarchical supervision allocation*. Our key insight is that productive guidance lies at the intersection of *future utility* and *current learnability*, as illustrated in Figure 1. Future utility asks whether an intervention improves the student's subsequent behavior; current learnability asks whether the student can realize that improvement under guidance and internalize it into its policy. Neither condition can be decided by token-level discrepancy alone. Moreover, the allocation decisions are nested: trajectory exposure determines which states can be visited, turn-level validation determines which intervention is worth learning from, and token-level refinement determines where the accepted signal should exert its learning pressure. This productive region also evolves throughout training. Guidance that is initially beyond the student's reach may later become learnable, while previously useful supervision may become redundant once absorbed. Effective OPD must therefore continually adapt both what the student explores and where it concentrates its learning capacity.

Based on this view, we propose **LENS-OPD**, a coarse-to-fine framework that allocates supervision through LOCATE, VALIDATE, and REFINE. LOCATE first adapts the rollout horizon to the student's evolving competence, then uses local discrepancy only to propose a candidate decision rather than treating it as evidence of usefulness. VALIDATE restores the same state and compares paired continuations generated by the same student with and without a short teacher hint. Because the student retains control in both branches, this comparison tests whether the guidance can improve its own future behavior instead of whether the teacher can execute a better action. For guidance that passes this test, REFINE transfers the induced behavior into the hint-free policy and reallocates token-level supervision toward decisive teacher–student conflicts. Each finer allocation is thus conditioned on the preceding coarser decision, progressively narrowing supervision from the right trajectory depth to the right decision and finally the right tokens.

In summary, this work makes three contributions. First, we formulate long-horizon OPD as hierarchical supervision allocation and identify future utility and current learnability as the two conditions for productive guidance. Second, we introduce LENS-OPD, which coordinates trajectory, turn, and token supervision through Locate–Validate–Refine. Third, experiments across long-horizon agent benchmarks, student scales, and model families show consistent gains, supported by analyses of adaptive supervision dynamics and component ablations.

2 Related Work

2.1 Knowledge Distillation and On-Policy Distillation

Knowledge distillation (Hinton et al., 2015) transfers teacher behavior through soft output distributions. Sequence-level distillation (Kim & Rush, 2016) trains students on teacher-generated sequences. For pretrained Transformers, distillation also uses hidden-state alignment (Sanh et al.,

2019; Sun et al., 2019; Jiao et al., 2020; Sun et al., 2020) and attention-relation matching (Wang et al., 2020; 2021). On-policy distillation (Gu et al., 2023; Agarwal et al., 2024) supervises students on their own generated prefixes, reducing the mismatch between teacher-generated training contexts and student-generated inference contexts. On-policy self-distillation (Zhao et al., 2026; Shenfeld et al., 2026) further derives teacher supervision from the same model conditioned on privileged information, transferring this guidance to a student that operates without such information.

2.2 Adaptive Supervision in On-Policy Distillation

Adaptive supervision in single-response generation adjusts the selection, strength, and form of token-level feedback. Teacher verification and entropy-based sampling (Huang et al., 2026; Ko et al., 2026) determine where supervision is applied, while uncertainty–disagreement analysis and teachability-aware selection (Xu et al., 2026; Wang et al., 2026b) distinguish informative corrections from locally incompatible teacher signals. Decision–evidence relations and reasoning-progress estimates (Xiao et al., 2026; Yang et al., 2026) also guide supervision allocation. Weighting strategies (Li et al., 2026b; Xie et al., 2026) account for response reliability and accumulated prefix mismatch. Position-dependent objectives (Jin et al., 2026; Jia et al., 2026) adapt supervision to teacher uncertainty or the sign of token-level advantage. Multi-turn supervision additionally controls state visitation and allocates feedback across decisions. Trajectory curricula and teacher–student control strategies (Wang et al., 2026a; Li et al., 2026a; Xia et al., 2026) regulate student exposure by progressively extending rollout horizons, annealing teacher intervention, or switching executors in response to accumulated discrepancy and recovery signals. Turn-aware budgeting and supervision schedules (Zhou et al., 2026b; Tan et al., 2026) further adapt rollout depth, redistribute learning effort across turns, and balance distillation with reward-driven optimization. Selective intervention and confidence-based weighting (Zhou et al., 2026a; Tan et al., 2026) use environment feedback, teacher judgment, or model uncertainty and disagreement to determine where guidance is needed and how strongly it should contribute. Future trajectory validation (Chen et al., 2026) tests whether teacher guidance yields more favorable downstream distillation signals in subsequent student continuations. Building on these directions, LENS-OPD conditions both guidance internalization and within-turn token refinement on a paired test of subsequent student behavior, within a trajectory curriculum that adapts to the student's evolving competence.

3 Effective Supervision as a Dynamic Frontier

3.1 Agentic On-Policy Distillation

We consider a student policy π_θ interacting with an environment for at most T turns. At turn t, it observes o_t, conditions on $h_t = (x, o_1, y_1, \ldots, o_t)$, samples the response $y_t \sim \pi_\theta(\cdot \mid h_t)$, and executes its action to obtain o_{t+1}. A frozen rollout snapshot $\bar{\pi}_\theta$ induces the training trajectory $\tau = (h_1, y_1, \ldots, h_T, y_T)$, and a frozen teacher π_{T} is evaluated on the same student-visited token contexts. Let $c_{t,i} = (h_t, y_{t,<i})$ denote the context for predicting token $y_{t,i}$, and let $m_{t,i} \in \{0, 1\}$ indicate whether that token is included in the training loss. Standard OPD minimizes

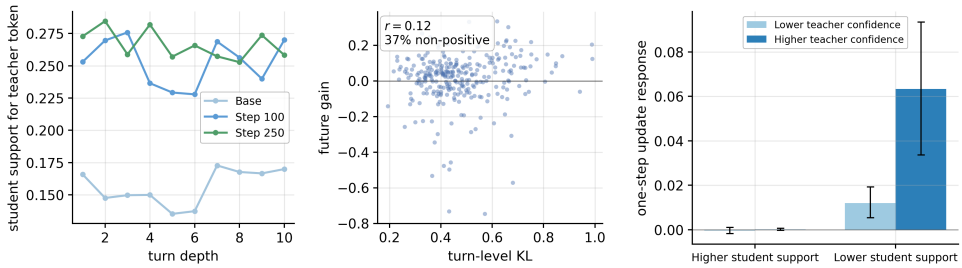
$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{\tau \sim \bar{\pi}_\theta}\left[\frac{\sum_{t,i} m_{t,i} D_{\mathrm{KL}}(\pi_\theta(\cdot \mid c_{t,i}) \,\|\, \pi_{\mathrm{T}}(\cdot \mid c_{t,i}))}{\sum_{t,i} m_{t,i}}\right]. \tag{1}$$

Although Equation 1 is on-policy with respect to the visited contexts, it weights all masked tokens uniformly, even though supervision differs in both future utility and current learnability. We call the evolving set of signals satisfying both conditions the *effective supervision frontier*.

3.2 Empirical Diagnostics

We conduct three controlled diagnostics on frozen student checkpoints, summarized in Figure 2. These probes examine how student support changes during training, whether local discrepancy predicts turn-level utility, and where update signal concentrates within future-beneficial turns.

Student support evolves with training. We first examine positions where the student and teacher prefer different tokens. At each such position, student support is the clipped probability ratio be-



(a) Support evolves with training (b) KL versus future gain (c) Token update response

Figure 2: **Diagnostics of hierarchical supervision allocation.** (a) Student support for teacher-preferred tokens at disagreement positions changes markedly with training. (b) Turn-level KL is a weak proxy for the future benefit of guidance, with 37% of valid forks producing non-positive gain. (c) Within future-beneficial turns, one-step corrective response is largest when the student assigns lower support and the teacher expresses higher confidence. Error bars in (c) are 95% bootstrap confidence intervals clustered by fork.

tween the teacher-preferred token and the student's own top token. Figure 2(a) compares this quantity across turn depths for the initial student and two later checkpoints. Support rises substantially after training and continues to shift between checkpoints, while its profile remains broadly consistent across depths. Thus, whether a teacher correction lies within the student's current support cannot be fixed in advance; supervision exposure must track the evolving student.

Local discrepancy does not determine future utility. We next sample candidate turns across the local-KL range and, from each restored state, compare three-turn continuations produced by the same frozen student with and without a one-sentence teacher cue. We define future gain $U_t \equiv \Delta_t^{\text{future}}$ as the difference in teacher log-likelihood between the two continuations. Across 286 valid forks, local KL is only weakly associated with U_t, and 37% of interventions yield non-positive gain (Figure 2(b)). Moreover, the maximum-KL turn is also the highest-gain turn in only one third of the trajectories. Local discrepancy can therefore propose a candidate decision, but cannot determine whether its guidance will improve the future.

Future utility does not imply uniform token actionability. Finally, among forks with positive future gain, we partition disagreement tokens by student support and teacher confidence. Starting from the same checkpoint, we apply one fixed OPD update and measure the increase in the student's log-probability of the teacher-preferred token. As shown in Figure 2(c), the response concentrates where student support is lower and teacher confidence is higher, reaching roughly five times the largest response in the other groups. Thus, future validation determines whether a turn provides productive guidance, but does not make all of its tokens equally actionable.

Together, these diagnostics motivate hierarchical rather than global supervision selection: trajectory exposure should adapt online, local KL should only nominate a candidate turn, and paired future validation should determine whether its guidance is productive. Only validated guidance is then refined at the behavior and token levels.

4 LENS-OPD

4.1 Framework Overview

Let π_θ be the trainable student, $\bar{\pi}_\theta$ the frozen snapshot used for rollout and target construction, and π_{T} a frozen teacher. LENS-OPD allocates supervision at three nested resolutions. At the trajectory level, a gap-adaptive curriculum controls how deeply $\bar{\pi}_\theta$ interacts with the environment. Within the resulting rollout, a cheap turn-level discrepancy proposes one decision for closer inspection. A paired future test then determines whether guidance at the proposed turn both improves the future and can be realized by the current student. Only an accepted turn activates two complementary mechanisms for absorption: guidance internalization transfers the hint-induced behavior into a hint-

free student, while confident-disagreement token focusing reallocates ordinary OPD supervision toward decisive corrective conflicts in the original on-policy response.

4.2 Gap-Adaptive Horizon Curriculum

LENS-OPD retains a stable Forward-to-Backward curriculum clock (Wang et al., 2026a), but adapts its pace to competence gained at the current depth. The controller maintains the rollout horizon K, a continuous clock e, a nominal residence budget η, and a maximum horizon $K_{\max}$.

Frontier competence. Consider an episode that reaches the current cap K. At content-token position i of its frontier turn, let v_i^{T} and v_i^{S} denote the teacher and student top-1 tokens under the same student-generated context. We define the soft support of the teacher token as

$$a_i = \min\left(1, \frac{\bar{\pi}_\theta(v_i^{\mathrm{T}} \mid c_{K,i})}{\bar{\pi}_\theta(v_i^{\mathrm{S}} \mid c_{K,i})}\right), \tag{2}$$

and average it over valid content positions to obtain episode competence $c_{b,K}$. Across episodes that reach K, the batch-level frontier competence is $C_{n,K} = \mathrm{Median}_{b \in \mathcal{B}_n^{\mathrm{valid}}} c_{b,K}$.

Gap-adaptive pace. Upon entering depth K, the first valid batch records the entry gap $G_K^{\mathrm{entry}} = 1 - C_{\mathrm{entry},K}$. For a later batch, let $G_{n,K} = 1 - C_{n,K}$. LENS advances the curriculum clock at rate

$$v_{n,K} = \mathrm{clip}\left(\frac{G_K^{\mathrm{entry}} + \epsilon}{G_{n,K} + \epsilon}, 0.5, 1.5\right), \qquad e \leftarrow e + v_{n,K}. \tag{3}$$

When $e \geq \eta$, the controller sets $K \leftarrow \min(K+1, K_{\max})$ and carries the fractional remainder $e \leftarrow e - \eta$ into the next depth. The clock therefore accelerates as the student closes the gap at the current frontier and slows when the gap expands.

4.3 Turn Proposal and Future Validation

The horizon controller determines which states are exposed, but it does not designate an intervention turn. For each valid turn $t \leq K$, we compute the mean content-token divergence

$$D_t = \frac{1}{|\mathcal{C}_t|} \sum_{i \in \mathcal{C}_t} D_{\mathrm{KL}}(\bar{\pi}_\theta(\cdot \mid c_{t,i}) \,\|\, \pi_{\mathrm{T}}(\cdot \mid c_{t,i})), \qquad t^\star = \arg\max_{t \leq K} D_t. \tag{4}$$

We use D_t only as a low-cost, high-recall proposal signal. As shown in Figure 2(b), local divergence is weakly associated with downstream benefit; thus, $t^\star$ identifies where to inspect, not where to supervise. We need test whether intervening at $t^\star$ improves the same student's subsequent behavior.

Paired future validation. Rather than replacing the student's proposed action or handing execution to the teacher as in teacher-execution schemes (Li et al., 2026a; Xia et al., 2026), the teacher supplies only a short hint $z_{t^\star}$. Starting from the same restored state, the same frozen student generates two counterfactual branches, without and with the hint:

$$x^{\mathrm{B}}, \xi^{\mathrm{B}} \sim \bar{\pi}_\theta(\cdot \mid h_{t^\star}), \tag{5}$$

$$x^{\mathrm{H}}, \xi^{\mathrm{H}} \sim \bar{\pi}_\theta(\cdot \mid h_{t^\star}, z_{t^\star}). \tag{6}$$

Because every environment action in both branches is selected by the student, the comparison remains close to the student's rollout distribution and tests whether the current student can realize the guidance, rather than whether the teacher can execute a better action. With a continuation evaluator V, we define

$$\Delta_{t^\star}^{\mathrm{future}} = V(\xi^{\mathrm{H}}) - V(\xi^{\mathrm{B}}), \qquad g_{t^\star} = \mathbf{1}[\Delta_{t^\star}^{\mathrm{future}} > 0]. \tag{7}$$

Our current instantiation uses the teacher's length-normalized likelihood of the realized continuation,

$$V(\xi) = \frac{1}{|\mathcal{A}(\xi)|} \sum_{j \in \mathcal{A}(\xi)} \log \pi_{\mathrm{T}}(x_j \mid c_j), \tag{8}$$

where $\mathcal{A}(\xi)$ contains the continuation's action/content tokens.

4.4 HIERARCHICAL REFINEMENT

Guidance internalization. Validation answers whether the hint elicits a better future, but the hint is unavailable at deployment. We therefore transfer the induced behavior across contexts: generate with privileged guidance, but learn to reproduce it without that guidance. For the hinted response $x^{\mathrm{H}} = (x_1^{\mathrm{H}}, \ldots, x_M^{\mathrm{H}})$, define the frozen privileged target and hint-free learner as

$$\begin{aligned} q_i^{\mathrm{H}}(\cdot) &= \mathrm{sg}\big[\bar{\pi}_\theta(\cdot \mid h_{t^\star}, z_{t^\star}, x_{<i}^{\mathrm{H}})\big], \\ p_i^0(\cdot) &= \pi_\theta(\cdot \mid h_{t^\star}, x_{<i}^{\mathrm{H}}). \end{aligned} \tag{9}$$

The guidance-internalization objective is

$$\mathcal{L}_{\mathrm{GI}} = g_{t^\star} \frac{1}{M} \sum_{i=1}^{M} D_{\mathrm{KL}}\big(q_i^{\mathrm{H}} \,\|\, p_i^0\big). \tag{10}$$

Because the learner conditions only on the canonical hint-free context, the accepted behavior is transferred into the deployable policy rather than retained as a dependence on privileged guidance.

Confident-disagreement token focusing. Within a future-validated turn, we further identify token positions that provide clear corrective supervision. We prioritize positions where the student strongly resists the teacher candidate while the teacher expresses a decisive preference.

For content token i of the original on-policy response, let v_i^{S} and v_i^{T} be the student and teacher top-1 candidates under the same context $c_{t^\star,i}$. We measure the student's counter-preference and the teacher's decisiveness as

$$\begin{aligned} g_i^{\mathrm{S}} &= \Big[\log \bar{\pi}_\theta(v_i^{\mathrm{S}} \mid c_{t^\star,i}) - \log \bar{\pi}_\theta(v_i^{\mathrm{T}} \mid c_{t^\star,i})\Big]_+, \\ \rho_i^{\mathrm{T}} &= \sigma\big(\log \pi_{\mathrm{T}}(v_i^{\mathrm{T}} \mid c_{t^\star,i}) - \log \pi_{\mathrm{T}}(v_i^{\mathrm{S}} \mid c_{t^\star,i})\big). \end{aligned} \tag{11}$$

Student disagreement alone can amplify ambiguous differences; teacher confidence alone can emphasize positions the student already matches. Their product isolates decisive corrective conflicts,

$$s_i = g_i^{\mathrm{S}} \rho_i^{\mathrm{T}}, \qquad \bar{r}_i = \min(c_{\mathrm{tok}}, 1 + \beta s_i), \qquad r_i = \frac{\bar{r}_i}{\frac{1}{|\mathcal{C}_{t^\star}|} \sum_{j \in \mathcal{C}_{t^\star}} \bar{r}_j + \epsilon}. \tag{12}$$

where c_{tok} caps the raw focusing weight. For every other trained position $i \notin \mathcal{C}_{t^\star}$, we set $r_i = 1$. We apply this redistribution conditionally as $\widetilde{r}_{t,i} = 1 + \mathbf{1}[t = t^\star] g_{t^\star}(r_i - 1)$, preserving the supervision budget while concentrating it on decisive corrective conflicts only within a validated turn.

4.5 UNIFIED OBJECTIVE

Let $m_{t,i}$ be the action/content-token mask and $\ell_{t,i}^{\mathrm{OPD}}$ the token-level estimator used by the underlying OPD algorithm. The focused on-policy loss is

$$\mathcal{L}_{\mathrm{focus}} = \frac{\sum_{t,i} m_{t,i} \widetilde{r}_{t,i} \ell_{t,i}^{\mathrm{OPD}}}{\sum_{t,i} m_{t,i} \widetilde{r}_{t,i} + \epsilon}. \tag{13}$$

LENS-OPD then optimizes $\mathcal{L}_{\mathrm{LENS}} = \mathcal{L}_{\mathrm{focus}} + \lambda_{\mathrm{GI}} \mathcal{L}_{\mathrm{GI}}$. The two terms share the future-validation gate but operate on different supervision sources: $\mathcal{L}_{\mathrm{focus}}$ reweights the original on-policy response, whereas $\mathcal{L}_{\mathrm{GI}}$ transfers the hint-induced response into a hint-free policy. The complete training procedure is provided in Algorithm 1 in the Appendix A.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

We evaluate LENS-OPD on ALFWorld (Shridhar et al., 2021), WebShop (Yao et al., 2022), and ScienceWorld (Wang et al., 2022). Our primary setting distills a frozen Qwen3-32B teacher into Qwen3-1.7B and Qwen3-4B students; an additional ALFWorld setting uses a Qwen2.5-7B-RL

Table 1: Main results on ALFWorld, WebShop, and ScienceWorld under two Qwen3 student scales. SR denotes success rate; higher is better.

Method	ALFWorld	WebShop		ScienceWorld	
	SR ↑	Score ↑	SR ↑	Score ↑	SR ↑
Qwen3-32B teacher → Qwen3-1.7B student					
Student (zero-shot)	5.8	32.7	5.0	3.7	0.4
Teacher (zero-shot)	49.3	51.8	19.0	44.8	28.7
OPD	36.4 ± 2.3	35.8 ± 13.9	9.6 ± 5.3	42.1 ± 0.3	14.3 ± 0.1
TCOD-F2B	42.6 ± 1.4↑6.2	30.4 ± 1.8↓5.4	7.4 ± 1.3↓2.2	29.4 ± 0.1↓12.7	15.4 ± 0.2↑1.1
TCOD-B2F	31.5 ± 0.6↓4.9	34.7 ± 7.0↓1.1	8.6 ± 4.8↓1.0	$\mathbf{42.6 \pm 0.5}$↑0.5	15.6 ± 1.1↑1.3
Guided-OPD	34.4 ± 2.3↓2.0	42.0 ± 4.8↑6.2	10.2 ± 1.2↑0.6	29.1 ± 0.1↓13.0	11.0 ± 0.1↓3.3
FTB	38.2 ± 4.0↑1.8	49.5 ± 1.9↑13.7	$\mathbf{18.2 \pm 3.7}$↑8.6	31.7 ± 0.1↓10.4	7.7 ± 0.2↓6.6
LENS-OPD (Ours)	$\mathbf{47.9 \pm 2.2}$↑11.5	$\mathbf{49.7 \pm 2.2}$↑13.9	14.4 ± 5.8↑4.8	35.8 ± 0.7↓6.3	$\mathbf{16.9 \pm 0.2}$↑2.6
Qwen3-32B teacher → Qwen3-4B student					
Student (zero-shot)	37.6	27.8	6.0	31.0	13.9
Teacher (zero-shot)	49.3	51.8	19.0	44.8	28.7
OPD	42.8 ± 1.3	53.3 ± 2.9	18.8 ± 0.8	40.3 ± 0.2	20.5 ± 0.2
TCOD-F2B	51.7 ± 2.5↑8.9	50.8 ± 0.6↓2.5	20.4 ± 0.8↑1.6	$\mathbf{43.3 \pm 0.3}$↑3.0	23.3 ± 0.2↑2.8
TCOD-B2F	43.5 ± 6.0↑0.7	49.1 ± 11.9↓4.2	20.2 ± 2.2↑1.4	39.8 ± 0.2↓0.5	21.9 ± 0.4↑1.4
Guided-OPD	46.1 ± 0.6↑3.3	54.3 ± 3.3↑1.0	20.2 ± 3.2↑1.4	38.8 ± 0.1↓1.5	14.7 ± 0.4↓5.8
FTB	45.2 ± 3.5↑2.4	52.0 ± 4.3↓1.3	18.8 ± 0.7	40.9 ± 0.3↑0.6	$\mathbf{26.8 \pm 0.3}$↑6.3
LENS-OPD (Ours)	$\mathbf{52.4 \pm 0.3}$↑9.6	$\mathbf{54.7 \pm 1.8}$↑1.4	$\mathbf{24.5 \pm 0.8}$↑5.7	41.2 ± 1.1↑0.9	23.4 ± 1.1↑2.9

teacher and a Qwen2.5-3B-Instruct student. We compare against vanilla OPD (Agarwal et al., 2024), TCOD (Wang et al., 2026a), Guided-OPD (Li et al., 2026a), and FutureBridge-OPD (FTB) (Chen et al., 2026) under matched training and evaluation budgets. We report mean ± standard deviation over five random seeds. Dataset splits, metrics, baselines, and complete training and evaluation configurations are provided in Appendix B, with prompt templates in Appendix C.

5.2 Main Results

Overall performance. Table 1 compares LENS-OPD with vanilla OPD and recent curriculum- and selection-based baselines across three benchmarks and two student scales. LENS-OPD delivers strong gains across these settings. On ALFWorld, it achieves 47.9 SR with the Qwen3-1.7B student, improving vanilla OPD by 11.5 points and the strongest baseline by 5.3 points. The gain remains substantial for the Qwen3-4B student, where LENS-OPD reaches 52.4 SR, a 9.6-point improvement over vanilla OPD. On WebShop, the 4B student obtains 24.5 SR, outperforming vanilla OPD by 5.7 points. On ScienceWorld, LENS-OPD improves SR from 14.3 to 16.9 for the 1.7B student and from 20.5 to 23.4 for the 4B student. These gains show that hierarchical supervision allocation provides consistently effective transfer across tasks and student scales.

Cross-family transfer. We further evaluate whether this consistency transfers to a second model family. Table 2 reports success rate and interaction efficiency on the seen, unseen, and hard ALFWorld splits for a Qwen2.5-7B-RL teacher paired with a Qwen2.5-3B student.

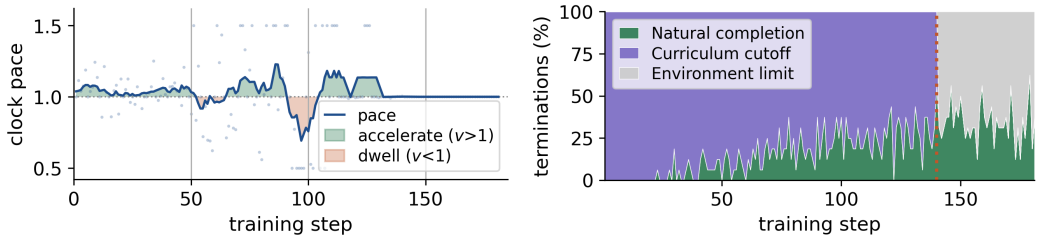
Under the Qwen2.5 teacher–student pair, LENS-OPD achieves 46.3, 41.9, and 9.8 SR on Valid Seen, Valid Unseen, and Hard, improving vanilla OPD by 5.9, 5.6, and 3.5 points, respectively. It simultaneously reduces the average number of interaction rounds by 1.5, 1.3, and 0.6 on the three splits, and consistently improves upon both fixed TCOD curricula. Together with the Qwen3 results, these gains demonstrate stable transfer across benchmarks, student scales, and model families rather than dependence on a particular backbone or predefined trajectory ordering.

5.3 Adaptive Supervision Dynamics

The trajectory frontier adapts to the student. Using the Qwen3-32B teacher and Qwen3-1.7B student on ALFWorld throughout this analysis, Figure 3 shows both the controller's instantaneous

Table 2: Results on ALFWorld with a Qwen2.5-7B-RL teacher and a Qwen2.5-3B student.

Method	Valid Seen		Valid Unseen		Hard	
	SR ↑	Rounds ↓	SR ↑	Rounds ↓	SR ↑	Rounds ↓
Qwen2.5-7B-RL teacher → Qwen2.5-3B student						
Student (zero-shot)	12.9	28.1	8.2	29.1	1.7	29.7
Teacher (zero-shot)	71.4	16.6	72.4	17.0	16.5	28.4
OPD	40.4 ± 3.6	21.7	36.3 ± 3.4	22.9	6.3 ± 0.9	29.1
Guided-OPD	22.3 ± 2.8↓18.1	26.0↑4.3	18.2 ± 2.1↓18.1	26.9↑4.0	1.0 ± 0.7↓5.3	29.9↑0.8
FTB	25.6 ± 1.9↓14.8	25.1↑3.4	22.4 ± 2.2↓13.9	26.0↑3.1	1.0 ± 1.4↓5.3	29.9↑0.8
TCOD-F2B	39.5 ± 3.5↓0.9	21.4↓0.3	32.5 ± 2.7↓3.8	23.2↑0.3	6.6 ± 1.6↑0.3	29.1
TCOD-B2F	40.7 ± 2.6↑0.3	21.0↓0.7	35.1 ± 1.9↓1.2	22.3↓0.6	7.4 ± 1.1↑1.1	28.8↓0.3
LENS-OPD (Ours)	$\mathbf{46.3 \pm 1.8}$↑5.9	**20.2**↓1.5	$\mathbf{41.9 \pm 2.3}$↑5.6	**21.6**↓1.3	$\mathbf{9.8 \pm 1.4}$↑3.5	**28.5**↓0.6



(a) Gap-adaptive clock pace

(b) Trajectory termination modes

Figure 3: **Gap-adaptive trajectory curriculum.** (a) Clock pace relative to fixed F2B ($v = 1$). (b) Trajectory termination modes as the accessible horizon expands.

pace and the resulting trajectory termination modes. The pace repeatedly rises above and falls below the fixed-F2B reference $v = 1$, including a pronounced slowdown near step 100, rather than following a uniform schedule. As the accessible horizon expands, curriculum cutoffs gradually give way to natural task completion; after the full horizon is reached, remaining failures are governed by the environment limit rather than curriculum truncation.

Future validation is beneficial on average but selective across turns. In Figure 4(a), guidance maintains a positive average future-score advantage throughout training. This average does not imply that every candidate is useful: the turn-level gains in Figure 4(b) span both sides of zero. The validation gate therefore performs a substantive selection, retaining positive-gain interventions and returning non-positive candidates to baseline OPD.

Validated guidance is internalized and focused. Although guided experience occupies a small and gradually decreasing fraction of the training tokens in Figure 4(c), the internalization gap contracts sharply, indicating that behaviors first elicited with guidance become increasingly reproducible by the deployable student policy. Within validated turns, Figure 4(d) shows a second level of allocation: target disagreement tokens account for only roughly one quarter of the tokens but consistently receive a much larger share of the normalized supervision mass. Thus, LENS-OPD adapts not only how far the student explores and which turns are reinforced, but also where learning capacity is concentrated within those turns.

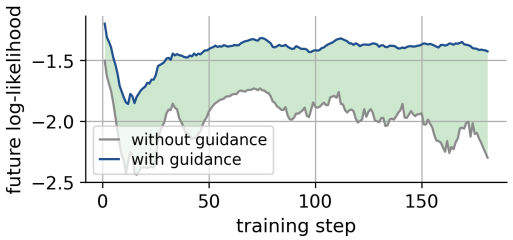
5.4 Ablation Study

Table 3 ablates the three levels of hierarchical allocation under the same ALFWorld configuration.

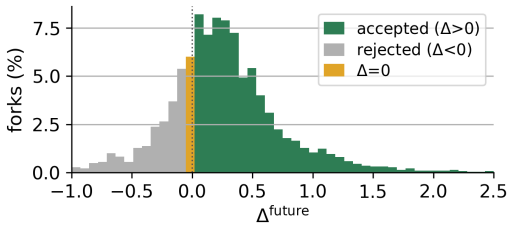
Trajectory-level curriculum. We first replace the adaptive curriculum with a predetermined forward-to-backward schedule while leaving turn validation and token refinement unchanged. This reduces SR by 6.9, 10.4, and 3.1 points on Valid Seen, Valid Unseen, and Hard, respectively, and requires 0.9 and 1.3 more rounds on the two validation splits. These results show that a fixed training schedule cannot reliably track the student's evolving supervision frontier.

Table 3: Ablation study on ALFWorld with a Qwen3-32B teacher → Qwen3-1.7B student.

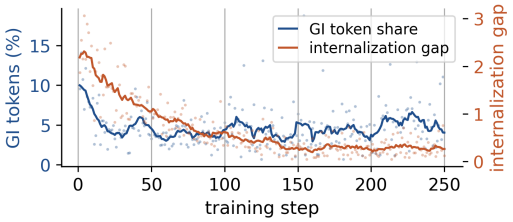
Method	Valid Seen		Valid Unseen		Hard	
	SR ↑	Rounds ↓	SR ↑	Rounds ↓	SR ↑	Rounds ↓
Fixed F2B Curriculum	37.7 ± 1.9	23.4	40.9 ± 1.7	23.4	16.4 ± 2.9	**28.0**
Last-Turn Proposal	37.4 ± 1.9	24.0	45.1 ± 3.0	23.6	13.7 ± 1.3	28.5
w/o Guidance Internalization	36.6 ± 2.0	23.0	44.8 ± 2.8	**21.9**	13.4 ± 1.9	28.1
w/o Token Focusing	39.4 ± 3.7	23.7	43.1 ± 3.4	23.3	$\mathbf{19.7 \pm 1.8}$	28.4
Reconsideration Prompt	39.9 ± 2.5	23.5	44.9 ± 3.6	22.9	19.5 ± 1.6	28.1
LENS-OPD (Full)	$\mathbf{44.6 \pm 2.9}$	**22.5**	$\mathbf{51.3 \pm 3.0}$	22.1	19.5 ± 2.2	28.1



(a) Batch-level future behavior



(b) Turn-level validation outcomes



(c) Guided-experience internalization



(d) Token-level supervision concentration

Figure 4: **Adaptive supervision dynamics.** Future validation (a,b), guided-experience internalization (c), and token-level supervision allocation (d).

Turn-level proposal. We replace discrepancy-based localization with the last valid turn while retaining the same paired validation and refinement mechanisms. This reduces SR by 7.2, 6.2, and 5.8 points on Valid Seen, Valid Unseen, and Hard, respectively, and requires 1.5 additional rounds on both validation splits. Thus, paired validation cannot fully compensate for a systematically poor proposal; it must begin from a plausible decision point.

Post-validation refinement. Finally, we examine how accepted guidance is absorbed by the student. Removing guidance internalization reduces SR by 8.0, 6.5, and 6.1 points across the three splits. Restoring uniform token weights lowers SR by 5.2 points on Valid Seen and 8.2 points on Valid Unseen. Replacing the default feedback-incorporation instruction with the *Reconsideration Prompt* in Appendix C.3 reduces SR by 4.7 and 6.4 points on the two validation splits, while Hard performance remains comparable. These results show that useful guidance must be internalized into the hint-free policy and focused on decisive token-level conflicts.

6 Conclusion

We formulate long-horizon OPD as hierarchical supervision allocation: productive guidance must improve future behavior while remaining learnable by the current student. LENS-OPD implements this principle through Locate, Validate, and Refine, which adapt trajectory exposure, verify guidance with paired student-controlled continuations, and focus learning on validated behaviors and decisive token-level conflicts. Across agent benchmarks, student scales, and model families, LENS-OPD consistently improves over vanilla OPD and strong adaptive baselines, showing the importance of jointly coordinating exploration, intervention, and internalization.

AI USE STATEMENT

LLMs were used to aid and polish writing and to support the retrieval and discovery of related work. The authors independently reviewed and verified all suggested text, references, technical claims, and final presentation, and take full responsibility for the manuscript.

ETHICS STATEMENT

All benchmarks and model checkpoints used in this work are publicly available. Our experiments are conducted in simulated, text-based environments and involve neither human participants nor private data, and no agent is deployed in a real-world system. Although improving long-horizon agents may support useful applications, more capable agents may also inherit or amplify undesirable behaviors and biases from their underlying models. We therefore restrict our evaluation to controlled research benchmarks and make no claim that the resulting agents are safe for deployment. Future applications should include appropriate robustness evaluation, human oversight, and safeguards against misuse.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce our study in the appendices. Appendix A presents the complete training algorithm and additional method details; Appendix B reports training configurations, method-specific hyperparameters, and evaluation protocols; and Appendix C gives the environment, teacher-cue, and student-feedback prompt templates. Our experiments use the publicly available ALFWorld, WebShop, and ScienceWorld benchmarks, with model pairings, metrics, and repeated-run aggregation specified in the experimental setup. Together, these details define the full training and evaluation pipeline required to reproduce the reported results.

REFERENCES

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

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems 28*, pp. 1171–1179, 2015.

Chishui Chen, Yaoyou Fan, Te Sun, Yi Yang, Chenghao Sun, Delin Mao, Hongbo Qiao, Zuowei Zhang, Junxi Wang, Chenxing Sun, et al. Look ahead before you distill: Future trajectory validation of teacher guidance for agentic on-policy distillation. *arXiv preprint arXiv:2608.01953*, 2026.

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

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

Haiduo Huang, Jiangcheng Song, Yadong Zhang, and Pengju Ren. Selectkd: Selective token-weighted knowledge distillation for llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19287–19297, 2026.

Nan Jia, Haojin Yang, Xing Ma, Jiesong Lian, Shuailiang Zhang, Weipeng Zhang, Ke Zeng, Xunliang Cai, and Zequn Sun. Asymmetric on-policy distillation: Bridging exploitation and imitation at the token level. *arXiv preprint arXiv:2605.06387*, 2026.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. In *Findings of the association for computational linguistics: EMNLP 2020*, pp. 4163–4174, 2020.

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 *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

Jongwoo Ko, Sara Abdali, Young Jin Kim, Tianyi Chen, and Pashmina Cameron. Scaling reasoning efficiently via relaxed on-policy distillation. *arXiv preprint arXiv:2603.11137*, 2026.

Gengsheng Li, Mao Zheng, Mingyang Song, Ruiqi Liu, Tianyu Yang, Jie Sun, Qiyong Zhong, Haiyun Guo, Junfeng Fang, Dan Zhang, et al. On-policy distillation with curriculum turn-level guidance for multi-turn agents. *arXiv preprint arXiv:2606.15912*, 2026a.

Yuying Li, Leqi Zheng, Yongzi Yu, Wenrui Zhou, Xuchang Zhong, Xing Hu, Jing Jin, Hangjie Yuan, and Tao Feng. Filter, then reweight: Rethinking optimization granularity in on-policy distillation. *arXiv preprint arXiv:2606.02684*, 2026b.

Stéphane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *JMLR Proceedings*, pp. 627–635, 2011. URL http://proceedings.mlr.press/v15/ross11a/ross11a.pdf.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew J. Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=0IOX0YcCdTn.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 4323–4332, 2019.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. Mobilebert: a compact task-agnostic bert for resource-limited devices. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 2158–2170, 2020.

Qitai Tan, Zefang Zong, Yang Li, and Peng Chen. Atod: Annealed turn-aware on-policy distillation for multi-turn autonomous agents. *arXiv preprint arXiv:2606.27814*, 2026.

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. Tcod: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. *arXiv preprint arXiv:2604.24005*, 2026a.

Ruoyao Wang, Peter A. Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. ScienceWorld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788, 2020.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 2140–2151, 2021.

Yuanyi Wang, Su Lu, Yanggan Gu, Pengkai Wang, Yifan Yang, Zhaoyi Yan, Congkai Xie, Jianmin Wu, and Hongxia Yang. Not all disagreement is learnable: Token teachability in on-policy distillation. *arXiv preprint arXiv:2605.26844*, 2026b.

Yuchen Xia, Qianguo Sun, Chao Song, Junlong Wu, Yiyan Qi, and Yunjian Xu. Dash-opd: Discrepancy-aware switching with hysteresis for on-policy distillation. *arXiv preprint arXiv:2607.29078*, 2026.

Jinwei Xiao, Zhuowen Han, Yueqing Sun, Zhengxi Lu, Yuxin Liu, Zhiyuan Yao, Wentao Chen, Qi Gu, and Xunliang Cai. Finding the evidence: Discovering decision-supporting tokens for on-policy reasoning distillation. *arXiv preprint arXiv:2606.22830*, 2026.

Yan Xie, Sijie Zhu, Tiansheng Wen, Bo Chen, and Yifei Wang. On the position bias of on-policy distillation. *arXiv preprint arXiv:2606.22600*, 2026.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. Tip: Token importance in on-policy distillation. *arXiv preprint arXiv:2604.14084*, 2026.

Chen Yang, Haiyuan Wan, Rengrong Xiong, Yize Chen, and Danny HK Tsang. Beyond imitation: Filtering on-policy distillation by reasoning progress. *arXiv preprint arXiv:2608.19408*, 2026.

Shunyu Yao, Howard Chen, John Yang, and Karthik R. Narasimhan. WebShop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems 35*, 2022.

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

Yuhang Zhou, Lizhu Zhang, Yifan Wu, Mingyi Wang, Bo Peng, Jiayi Liu, Xiangjun Fan, and Zhuokai Zhao. Sage-opd: Selective agent-guided intervention for multi-turn on-policy distillation. *arXiv preprint arXiv:2606.19659*, 2026a.

Yuhang Zhou, Kai Zheng, Haoling Li, Dengyun Peng, Can Xu, and Jingjing Chen. Turnopd: Making on-policy distillation turn-aware for efficient long-horizon agent training. *arXiv preprint arXiv:2607.05804*, 2026b.

A Additional Method Details

A.1 Training Algorithm Overview

Algorithm 1 summarizes one LENS-OPD training iteration. The three stages operate at progressively finer resolutions: the curriculum controls trajectory exposure, future validation evaluates one proposed intervention, and refinement converts an accepted intervention into deployable behavioral and token-level supervision. The curriculum statistic is measured at the current rollout frontier, whereas the proposed intervention may occur at any valid turn within the exposed trajectory.

Algorithm 1 One training iteration of LENS-OPD

Require: Teacher π_{T}; trainable student π_θ; frozen rollout student $\bar{\pi}_\theta$; controller state $(K, e, G_K^{\mathrm{entry}})$; residence budget η
Ensure: Updated student π_θ and curriculum state

Locate: adaptive exposure and turn proposal
Roll out $\bar{\pi}_\theta$ for at most K turns to obtain a batch $\mathcal{B}$
if at least one valid episode reaches depth K **then**
  Compute frontier competence $C_{n,K}$ over valid episodes
  Initialize G_K^{entry} if the controller has just entered depth K
  $v \leftarrow \mathrm{clip}\big((G_K^{\mathrm{entry}} + \epsilon)/(1 - C_{n,K} + \epsilon), 0.5, 1.5\big)$
else
  $v \leftarrow 1$ ▷ fall back to the reference F2B pace
end if
$e \leftarrow e + v$; if $e \geq \eta$, set $K \leftarrow \min(K + 1, K_{\max})$ and $e \leftarrow e - \eta$
for all student trajectories $\tau_b \in \mathcal{B}$ **do**
  Compute mean teacher–student divergence $D_{b,t}$ for every valid turn t
  $t_b^\star \leftarrow \arg\max_t D_{b,t}$
Validate: paired future continuations
  Restore the canonical history and environment state at $t_b^\star$
  Generate base response and continuation $(x_b^{\mathrm{B}}, \xi_b^{\mathrm{B}})$
  Query one teacher hint z_b and generate $(x_b^{\mathrm{H}}, \xi_b^{\mathrm{H}})$
  $g_b \leftarrow \mathbb{I}\big[V(\xi_b^{\mathrm{H}}) - V(\xi_b^{\mathrm{B}}) > 0\big]$
Refine: guidance internalization and token focusing
  Initialize $\mathcal{L}_{\mathrm{GI}}^{(b)} \leftarrow 0$ and $\widetilde{r}_{b,t,i} \leftarrow 1$
  if $g_b = 1$ **then**
    Construct the hinted frozen-student target and hint-free learner
    Compute cross-context guidance-internalization loss $\mathcal{L}_{\mathrm{GI}}^{(b)}$
    Compute normalized confident-disagreement weights $r_{b,i}$ on original turn $t_b^\star$
    Set $\widetilde{r}_{b,t_b^\star,i} \leftarrow r_{b,i}$
  end if
  Compute $\mathcal{L}_{\mathrm{focus}}^{(b)}$ on the original on-policy trajectory
end for
$\mathcal{L} \leftarrow \mathrm{Aggregate}_b\Big[\mathcal{L}_{\mathrm{focus}}^{(b)} + \lambda_{\mathrm{GI}}\mathcal{L}_{\mathrm{GI}}^{(b)}\Big]$
Update θ using $\nabla_\theta \mathcal{L}$ and synchronize $\bar{\pi}_\theta$ according to the rollout schedule

B Experimental Details

B.1 Training Configuration

We adopt the shared training configuration in TCOD (Wang et al., 2026a), Appendix D.3, with five random seeds (42, 43, 44, 45, and 46). Table 4 records the optimization, rollout, and distributed execution settings. Method-specific objectives and guidance mechanisms are described in the main text. Each run uses 250 training steps, with evaluation every five steps and checkpoint saving at step 250. For the ALFWorld experiments with the Qwen2.5-7B-RL teacher, we use `siaosiao/grpo-qwen2.5-7b-alfworld-global-step-150`, a Qwen2.5-7B RL checkpoint obtained after 150 steps of GRPO training on ALFWorld. This checkpoint is kept frozen throughout student distillation. For LENS-OPD, the curriculum residence budget is set to $\eta = 5$, matching the fixed F2B clock when $v = 1$. We validate one Top-KL turn per trajectory using three-turn paired continuations.

Within each comparison, methods share the teacher checkpoint, student initialization, prompts, environment configuration, action parser, and rollout, optimization, and evaluation budgets in a common codebase. The teacher remains frozen, and student trajectories are collected on-policy at temperature 1.0 without beam search or candidate reranking. We compare with vanilla OPD (Agarwal et al., 2024), the forward-to-backward and backward-to-forward variants of TCOD (Wang et al., 2026a), Guided-OPD (Li et al., 2026a), and FutureBridge-OPD (Chen et al., 2026). The zero-shot student and teacher are reported as references and excluded from method rankings.

Table 4: Training configuration across the three benchmarks. Batch-size field names follow TCOD.

Parameter	Value
KL coefficient	1.0
Learning rate	1×10^{-6}
Gradient clipping	1.0
Repeat times	1
Maximum sample staleness	2
Training steps	250
Batch size / train batch size	16 / 64
Save / evaluation interval	250 / 5 steps
Maximum prompt / response tokens	10,240 / 512
Rollout temperature	1.0
Log-probability collection	All tokens
Random seeds	42, 43, 44, 45, 46
Curriculum residence budget η	5
Top-KL turn proposals per trajectory	1
Future-validation horizon H	3 turns
Guidance-score temperature τ	1.0
Token-focus scale β	1.0
Raw token-weight cap c_{tok}	5.0
Nodes / GPUs per node	1 / 8
GPU model	NVIDIA H100
Tensor / sequence parallelism	2 / 2 (Ulysses)
Maximum tokens per GPU	16,384
GPU memory utilization	0.7
Precision	BFloat16

B.2 Evaluation Configuration

Following TCOD and FutureBridge-OPD (Wang et al., 2026a; Chen et al., 2026), we use the standard ALFWorld training and unseen-evaluation splits, and the TCOD-aligned WebShop training subset with 100 held-out sessions. Evaluation uses the decoding and execution settings in TCOD Appendix D.4, summarized in Table 5. Thinking mode is disabled. The evaluation generation limit is 4,096 tokens, whereas training rollouts use a maximum response length of 512 tokens. The environment interaction limits apply during both training and evaluation. We report the mean and standard deviation across the five seeds above.

Table 5: Evaluation decoding and environment execution settings.

Parameter	Value
Maximum generation tokens	4,096
Temperature	0.4
Top-p / top-k / min-p	1.0 / -1 / 0.0
History length	2 steps
Maximum interactions: ALFWorld	30
Maximum interactions: ScienceWorld	30
Maximum interactions: WebShop	15
Evaluation workers	8
Process timeout	3,600 seconds
Synchronization	Dynamic by explorer

Evaluation metrics. For each benchmark, let N denote the number of evaluation episodes for a given seed. ALFWorld uses the environment's task-completion criterion: with $c_i \in \{0, 1\}$ indicating whether episode i successfully completes the task, we report

$$\mathrm{SR}_{\mathrm{ALFWorld}} = \frac{100}{N} \sum_{i=1}^{N} c_i. \tag{14}$$

For WebShop, let $r_i \in [0, 1]$ be the final environment reward. The score follows the reward average used by FutureBridge-OPD (Chen et al., 2026), while success requires $r_i = 1$:

$$\mathrm{Score}_{\mathrm{WebShop}} = \frac{100}{N} \sum_{i=1}^{N} r_i, \qquad \mathrm{SR}_{\mathrm{WebShop}} = \frac{100}{N} \sum_{i=1}^{N} \mathbf{1}[r_i = 1]. \tag{15}$$

For ScienceWorld, let s_i denote the final environment score on its native 0–100 scale, with any negative terminal score replaced by zero before averaging. Success follows the environment's standard task-completion criterion, denoted by $c_i \in \{0, 1\}$:

$$\mathrm{Score}_{\mathrm{ScienceWorld}} = \frac{1}{N} \sum_{i=1}^{N} \max(s_i, 0), \qquad \mathrm{SR}_{\mathrm{ScienceWorld}} = \frac{100}{N} \sum_{i=1}^{N} c_i. \tag{16}$$

Scores use the final outcome of each episode, including episodes that reach the interaction limit. We compute each metric separately for each seed and then report its mean and standard deviation across the five seeds.

C Prompt Templates

We use two distinct families of prompts. The environment interaction prompts define the ordinary student and teacher policies used by all OPD baselines. They follow the benchmark-specific templates and output conventions of TCOD Appendix E (Wang et al., 2026a). LENS-OPD additionally uses a short diagnostic cue and a student-side feedback injection only inside the paired future-validation branch. The cue is privileged audit information: the teacher never executes an environment action, and neither audit continuation is inserted into the ordinary OPD buffer.

C.1 Environment Interaction Prompts

The three environments share the same slot structure—task, recent interaction history, current observation, and currently available actions—while retaining their native action vocabularies. The templates below are functionally identical to those used by TCOD; braces denote runtime fields.

ALFWorld

```
You are an expert agent operating in the ALFRED Embodied Environment.
Your task is to: {task_description}
Prior to this step, you have already taken {step_count} step(s).
Below are the most recent {history_length} observations and the
corresponding actions you took: {action_history}
You are now at step {current_step} and your current observation is:
{current_observation}
Your admissible actions of the current situation are:
[{admissible_actions}].

Now it's your turn to take an action.
You should first reason step-by-step about the current situation.
Once you've finished your reasoning, you should choose an admissible
action for current step and present it within <action> </action> tags.
```

ScienceWorld

```
Your ScienceWorld task is: {task_description}
Prior to this step, you have already taken {step_count} step(s).
Below are the most recent {history_length} observations and the
corresponding actions you took: {action_history}
You are now at step {current_step} and your current observation is:
{current_observation}
Available action commands: [{action_templates}]
Available objects you can interact with: [{objects}]

Now it's your turn to take an action. Combine an action command with
appropriate object(s) to form a valid action.
You should first reason step-by-step about the current situation.
Once you've finished your reasoning, you should choose a valid action
for the current step and present it within <action> </action> tags.
```

WebShop

```
You are an expert autonomous agent operating in the WebShop
e-commerce environment.
Your task is to: {task_description}.
Prior to this step, you have already taken {step_count} step(s).
Below are the most recent {history_length} observations and the
corresponding actions you took: {action_history}
You are now at step {current_step} and your current observation is:
{current_observation}.
Your admissible actions of the current situation are:
[
{available_actions}
].

Now it's your turn to take one action for the current step.
You should first reason step-by-step about the current situation,
then think carefully which admissible action best advances the
shopping goal.
Once you've finished your reasoning, you should choose an admissible
action for current step and present it within <action> </action> tags.
```

WebShop actions use the environment-native `search[query]` and `click[element]` formats whenever those controls are available.

C.2 Teacher Diagnostic Cue

At the proposed fork turn, the teacher observes the same replay-restored environment state that the student is about to face. We provide the full admissible-action set, the student's proposed action, and the two most recent interaction turns. The teacher generates exactly one diagnostic sentence.

```
You are diagnosing a student agent's proposed action. You see exactly
what the student sees.

Task:
{task}

Current observation:
{observation}

Available actions:
{admissible_actions}
```

```
Interaction history:
{history}

The student is about to take this action:
{student_action}

Judge this action against the current state and give exactly one short
hint.

Rules:
- If the action is wrong, or has an important oversight/constraint the
  student should reconsider, POINT IT OUT DIRECTLY -- say what the
  issue is and what the student should do instead (you may name the
  correct action).
- If the action is right, ENCOURAGE / confirm the student: say the
  action is on the right track and (if useful) suggest the next step.
- Do not write a tool call.
- One sentence only.
```

We decode the cue greedily (temperature 0) with at most 80 generated tokens and retain the two most recent interaction turns as history. If cue generation fails, the trajectory receives no forked audit and is not retried. This asymmetric instruction deliberately supports both correction and confirmation: a wrong proposal receives a direct diagnosis, whereas a correct proposal receives confirmation and, when useful, a next-step suggestion.

C.3 Student Feedback Injection and Prompt Ablation

For the hinted branch, our default *Feedback-Incorporation Prompt* is inserted at the fork turn's student-message position. The prefix preceding the insertion is otherwise identical to the base branch. The cue therefore changes only the information available to the student; the teacher still does not act.

```
You proposed the following action earlier:

{student_action}

Teacher feedback:

{teacher_hint}

Take the teacher's feedback into account and produce the next action
yourself.
```

In the *Reconsideration Prompt* ablation, we keep all fields and formatting fixed but replace the final instruction with the following more forceful alternative:

```
Reconsider your decision and produce the next action yourself.
```

Both variants require the student to produce the next action itself. The ablation therefore tests the effect of feedback framing rather than teacher execution, additional state information, or a different action interface.

D Additional Results

D.1 WebShop Training Dynamics

We extend the trajectory and future-validation diagnostics in Figures 3 and 4(a,b) to WebShop, using a Qwen3-32B teacher with Qwen3-1.7B and Qwen3-4B students. Each row shows one training run.

Training-step coordinates identify the student version that generated each rollout. Clock-pace curves use centered seven-batch moving averages; future-score curves pool completed paired validations over the same window size. Termination proportions and gain histograms are unsmoothed.

Figure 5 shows that both curricula adjust their pace before reaching the full 15-turn horizon, first used at student versions 87 and 80 for the 1.7B and 4B students, respectively. Subsequent terminations arise from natural environment completion or the interaction limit. Natural completion does not imply task success. The shaded region in the pace panels marks the completed curriculum; no clock speed is imputed there.

Figure 6 compares length-normalized teacher log-likelihoods of guided and unguided continuations from the same fork. There are 760 complete pairs among 3,165 recorded episodes for the 1.7B student and 778 among 3,245 for the 4B student. Incomplete pairs, predominantly due to timeouts, are excluded rather than assigned zero gain; these comparisons therefore describe completed validations. One unfinished trailing batch per run, containing eight episodes without a logged student-version mapping, is excluded from both figures. Across complete pairs, mean future gains are 0.066 and 0.069, respectively, while both distributions retain a negative-gain region. Guidance is beneficial on average within this observed subset, but individual interventions still require validation.

D.2 ScienceWorld Training Dynamics

We repeat the same trajectory and future-validation diagnostics on ScienceWorld with a Qwen3-32B teacher and Qwen3-1.7B or Qwen3-4B students. As shown in Figure 7, the clock repeatedly accelerates and dwells relative to fixed F2B before both runs reach the full $K = 30$ horizon near step 200. Over the same period, curriculum cutoffs recede as natural completion becomes more frequent; after the full horizon is reached, trajectories terminate through natural completion or the environment limit. The distinct pace profiles at the two student scales further show that trajectory exposure follows the observed student–teacher gap rather than a shared preset schedule.

Figure 8 shows that guided continuations achieve higher teacher-evaluated future scores over most of training for both student scales. Nevertheless, the 2,839 and 3,040 valid paired forks for the 1.7B and 4B students, respectively, contain gains on both sides of zero. Thus, the average benefit of guidance transfers to ScienceWorld, while the turn-specific variation continues to justify paired validation rather than accepting every proposed intervention.

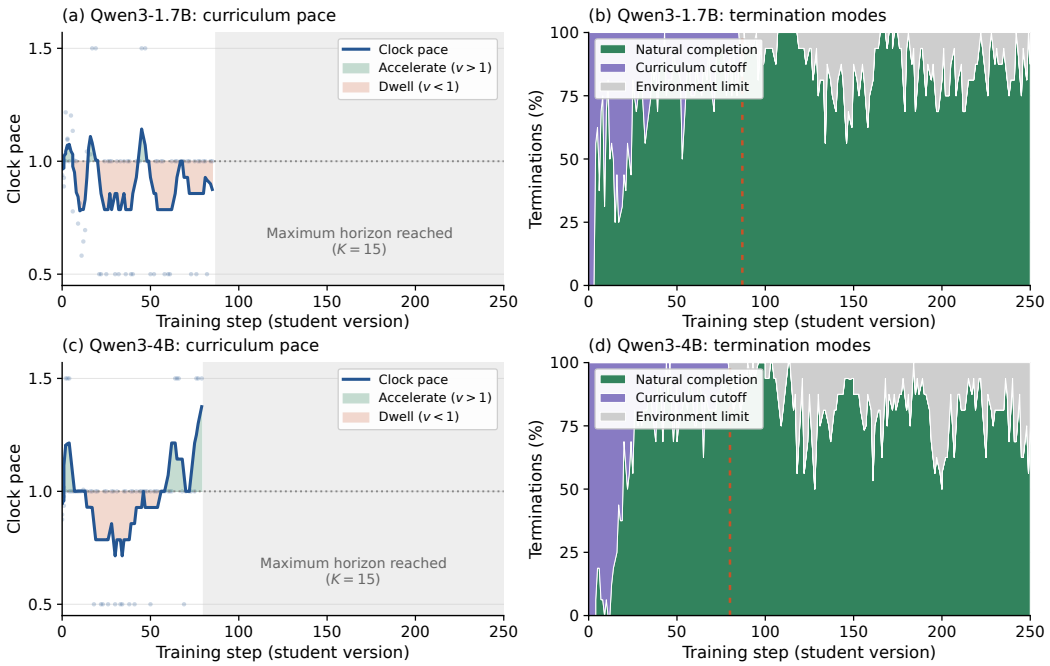


Figure 5: **WebShop trajectory curriculum.** Top: Qwen3-1.7B; bottom: Qwen3-4B. Left: recorded clock pace (dots) and its moving average, relative to fixed F2B ($v = 1$). Right: trajectory-termination proportions; dashed lines mark the first rollouts using the full horizon.

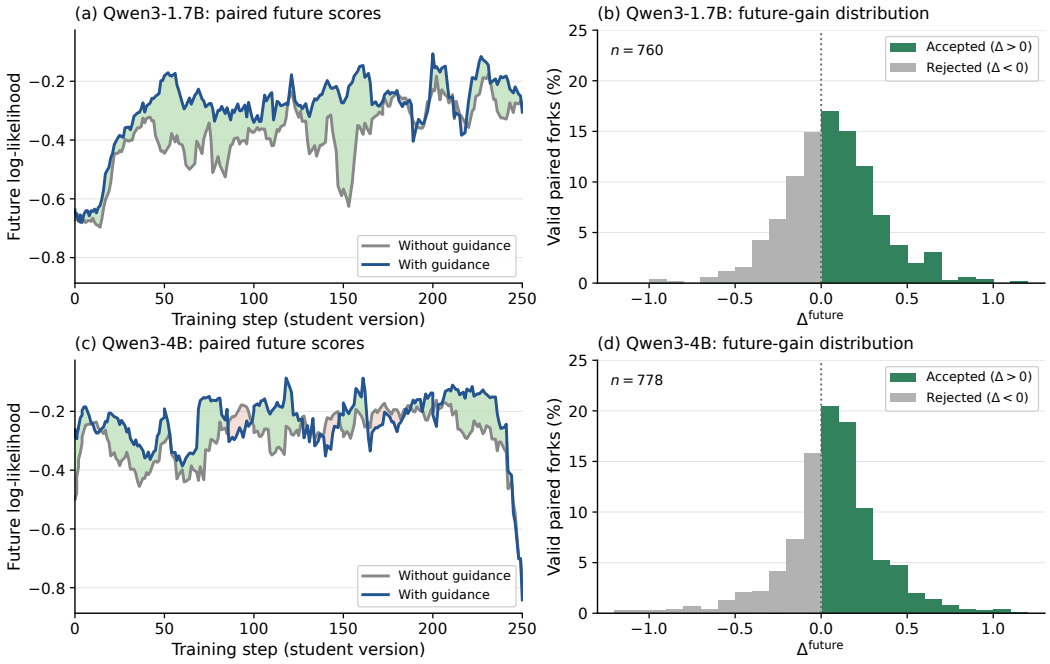


Figure 6: **WebShop future validation.** Top: Qwen3-1.7B; bottom: Qwen3-4B. Left: guided and unguided future scores over the same complete pairs. Right: unsmoothed future-gain distributions, normalized by the number of complete pairs in each run. Green and gray denote positive and negative gains, respectively.

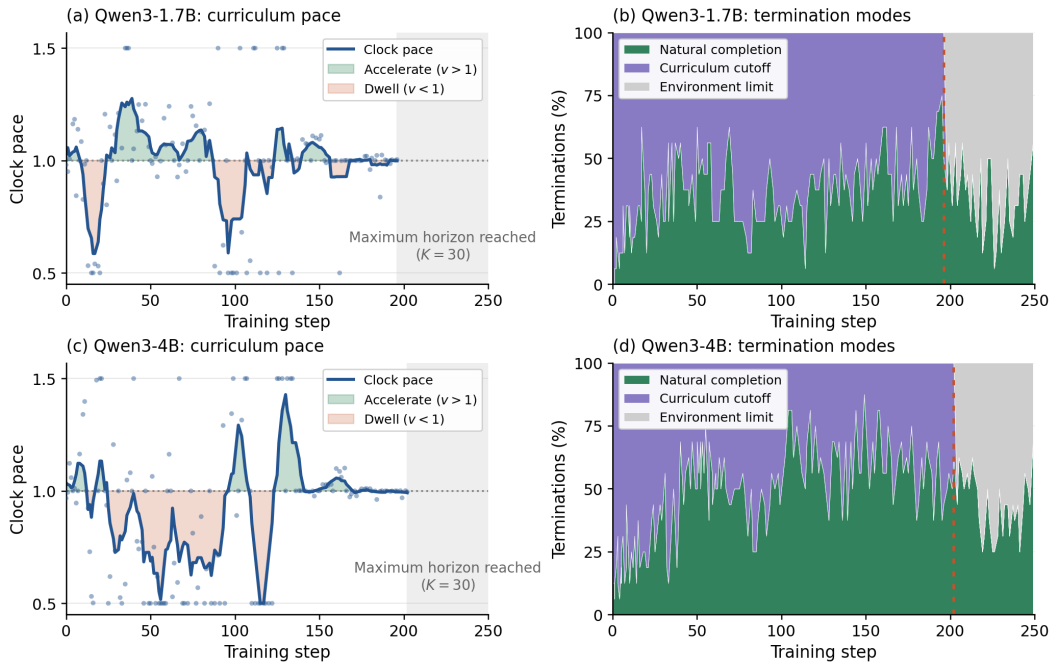


Figure 7: **ScienceWorld trajectory curriculum.** Top: Qwen3-1.7B; bottom: Qwen3-4B. Left: clock pace relative to fixed F2B ($v = 1$). Right: trajectory-termination proportions. Dashed lines mark the first rollouts using the full $K = 30$ horizon.

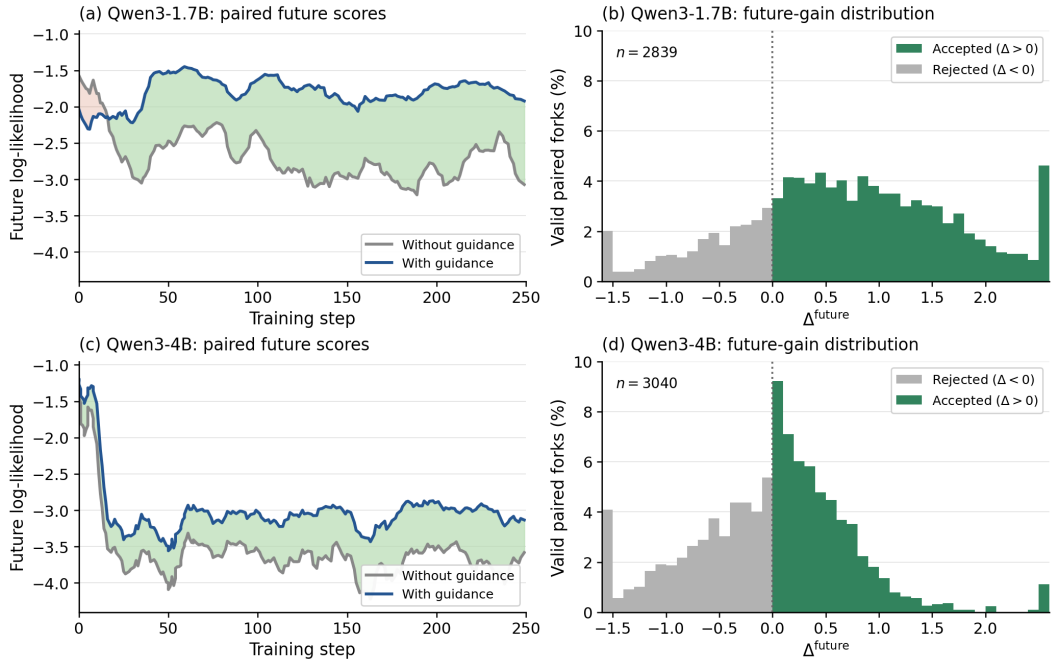


Figure 8: **ScienceWorld future validation.** Top: Qwen3-1.7B; bottom: Qwen3-4B. Left: guided and unguided future scores. Right: future-gain distributions over valid paired forks, normalized separately for each run. Green and gray denote positive and negative gains.