AGENTIC ON-POLICY DISTILLATION: DISTILLING DECISIONS, NOT TOKENS

Anonymous authors
Paper under double-blind review

ABSTRACT

Distilling capable language-model agents into smaller models requires aligning the student's behavior with the teacher's. This raises a fundamental question: at which states and at what granularity should this alignment be performed? Offline imitation provides dense but token-specific supervision and can suffer from state-distribution mismatch. Token-level on-policy distillation (OPD) addresses this mismatch but typically requires teacher token probabilities and compatible tokenization. We argue that semantic actions provide a natural unit of supervision for agent distillation. We introduce Agentic On-Policy Distillation (AOPD), motivated by maximizing expected teacher-action likelihood at student-encountered states. At each such state, AOPD samples actions from both the student and a black-box teacher, rewarding student actions according to their semantic agreement with the sampled teacher actions. Through policy optimization, AOPD aligns the student's action-level behavior with the teacher's decisions without requiring reproduction of teacher-written reasoning or access to teacher token probabilities. Training Qwen3-4B and Qwen3-8B with two teachers on diverse, curated tool-use tasks yields average gains of 3.4–6.6 percentage points across four out-of-distribution benchmarks. AOPD scales favorably with student size and teacher capability, without requiring the teacher and student to share a model family or tokenizer. On τ^2-bench, AOPD achieves competitive performance against offline distillation and token-level OPD. We further explore Self-AOPD with a same-model teacher and privileged context. Together, these results establish AOPD as a promising approach to agent distillation.

1 INTRODUCTION

Language models are increasingly deployed as agents that communicate with users, maintain state, and act through external tools (Yao et al., 2023; 2025). In these interactions, an agent's decisions shape subsequent observations and the states it encounters. Distilling capable but expensive frontier agents into smaller models therefore raises two questions: *at which states* should the teacher supervise the student, and *at what granularity* should their behavior be aligned?

Distillation methods address these questions differently. Offline distillation provides dense supervision by fitting teacher-generated sequences (Kim & Rush, 2016), but can face teacher–student state-distribution mismatch (Ross et al., 2011). Token-level on-policy distillation (OPD) supervises student-generated sequences using teacher token probabilities, reducing mismatch while typically requiring compatible tokenization (Agarwal et al., 2024; Gu et al., 2024). Both supervise tokens without explicitly matching action semantics (Figure 2).

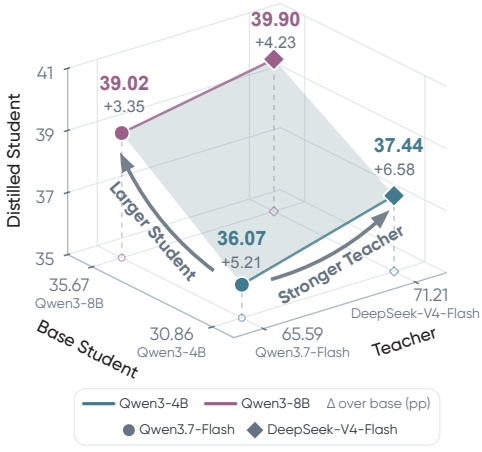


Figure 1: Student–teacher scaling of AOPD. Larger students attain higher final averages. DeepSeek-V4-Flash yields higher student averages than Qwen3.7-Flash.

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.

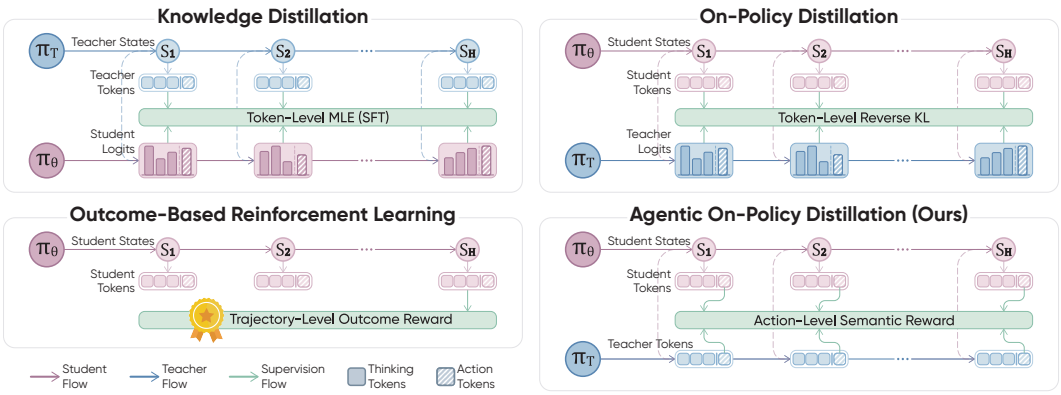


Figure 2: Supervision for agent learning. Offline distillation (KD) imitates teacher tokens at teacher-induced states. Token-level OPD compares token distributions along student-generated interactions. Outcome-based RL uses terminal feedback. AOPD instead rewards semantic agreement with teacher actions at student-encountered states. Here, we sample only one response per state for demonstration. For implementation, we expands the sampled teacher multiset and student state group.

Our key observation is that *semantic actions* provide a natural unit of supervision for agent distillation. A tool call expresses a decision through its tool name and arguments; a natural-language response conveys communicative intent and information. The same decision can be realized through different wording and reasoning. This shifts the objective from reproducing a particular teacher sequence to increasing the likelihood of its underlying action.

We introduce **Agentic On-Policy Distillation (AOPD)**, motivated by maximizing expected teacher-action likelihood at student-encountered states. Here, likelihood is the total probability mass of student outputs that semantically match a teacher action, not the probability of reproducing one teacher-written sequence. This objective yields a REINFORCE-form policy gradient with action-level semantic agreement as reward (Williams, 1992). In practice, AOPD samples actions from the teacher and responses from the current student policy, rewarding student responses according to their semantic agreement with teacher actions. It requires only generation access to a black-box teacher, not teacher token probabilities or tokenizer alignment. The resulting action-level rewards train the student's own reasoning and action tokens without imposing teacher-written reasoning as a token-level imitation target.

We train Qwen3-4B and Qwen3-8B students using teachers from two model families on curated tool-use tasks from Agent World Model (AWM) and EnvScaler (Wang et al., 2026b; Song et al., 2026). All four configurations improve average performance across four out-of-distribution agent benchmarks. AOPD scales favorably with student size and teacher capability: larger students achieve higher final averages, and DeepSeek-V4-Flash yields higher student averages than Qwen3.7-Flash at both sizes (Figure 1). Comparisons on τ^2-bench (Barres et al., 2025) show competitive performance against offline distillation and token-level OPD. Ablation studies support the effectiveness of semantic action supervision. We further explore Self-AOPD, in which a same-model teacher receives privileged task information. This extension improves held-out averages while the student uses only public interaction context during both training and evaluation.

Contributions. Our contributions are threefold:

- We introduce AOPD, an action-level on-policy distillation method that decouples teacher decision supervision from student token realization, enabling distillation from black-box teachers without teacher token probabilities or tokenizer alignment.
- We demonstrate AOPD's applicability across teacher families and empirical scaling with student size and teacher capability. All tested configurations improve the four-benchmark average over their corresponding base models by 3.4–6.6 percentage points.
- We show that AOPD achieves competitive performance against offline distillation and token-level OPD while requiring only generation access to the teacher.

2 AGENTIC ON-POLICY DISTILLATION

A language-model agent alternates between generating responses and receiving environment observations. At interaction step t, its visible state s_t contains the task, available tools, and retained interaction history. Its response y_t realizes a semantic action $a_t = \phi_{s_t}(y_t)$ that captures the meaning of a tool call or a message to the user. Here ϕ_s maps responses to semantic actions at state s. We use π_θ for student probabilities over semantic actions, complete responses, and individual tokens, distinguished by their arguments, and π_{T} for the teacher. Student and teacher receive the same public context and tools through their native prompts.

To align the student's action-level behavior with the teacher, we maximize expected teacher-action likelihood at student-encountered states:

$$\max_\theta \ \mathbb{E}_{s\sim d_{\mathrm{old}},\, a^{\mathrm{T}}\sim\pi_{\mathrm{T}}(\cdot|s)} \left[\pi_\theta(a^{\mathrm{T}} \mid s)\right]. \tag{1}$$

Denote this expectation by $J_{\mathrm{AOPD}}(\theta)$. The collected-state distribution d_{old} is induced by the rollout in Section 2.3 and held fixed during each update.

2.1 FROM ACTION LIKELIHOOD TO POLICY GRADIENTS

To optimize Equation 1, we first express the teacher-action likelihood in terms of student response probabilities. This yields:

$$\pi_\theta(a^{\mathrm{T}} \mid s) = \sum_y \pi_\theta(y \mid s)\, \mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\}. \tag{2}$$

Here, equality is semantic rather than textual. Substituting Equation 2 into Equation 1 and exchanging expectations gives an equivalent expected-reward objective:

$$\begin{aligned} r_{\mathrm{T}}(s,a) &= \mathbb{E}_{a^{\mathrm{T}}\sim\pi_{\mathrm{T}}(\cdot|s)}[\mathbf{1}\{a = a^{\mathrm{T}}\}], \\ J_{\mathrm{AOPD}}(\theta) &= \mathbb{E}_{s\sim d_{\mathrm{old}},\, y\sim\pi_\theta(\cdot|s)}[r_{\mathrm{T}}(s,\phi_s(y))]. \end{aligned} \tag{3}$$

With the state distribution, teacher policy, and semantic mapping held fixed during the update, the score-function identity yields

$$\nabla_\theta J_{\mathrm{AOPD}}(\theta) = \mathbb{E}_{s,a^{\mathrm{T}},y}\left[\mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\}\nabla_\theta \log \pi_\theta(y \mid s)\right], \tag{4}$$

where the variables follow the distributions above. This is precisely the REINFORCE policy-gradient form (Williams, 1992), with semantic action agreement as reward (Appendix A.1).

This gradient can be estimated from sampled teacher actions and student responses using student log-probabilities, without teacher token probabilities or tokenizer alignment. Since $\log \pi_\theta(y \mid s) = \sum_{\ell=1}^{|y|} \log \pi_\theta(y_\ell \mid s, y_{<\ell})$, where ℓ indexes response tokens, a single semantic-agreement reward weights every token's contribution to the policy gradient. The supervision is therefore action-level, while optimization covers the student's own reasoning and action tokens without imposing teacher-written reasoning as a hard target.

2.2 BLACK-BOX SEMANTIC ACTION SUPERVISION

The gradient above requires an agreement reward for each student response. We estimate it using sampled teacher actions and a practical semantic matcher. At state s, we request $K = 3$ teacher responses realizing $a_k^{\mathrm{T}} = \phi_s(y_k^{\mathrm{T}})$ and retain the multiset $\mathcal{M}_{\mathrm{T}}(s) = [y_1^{\mathrm{T}}, \ldots, y_K^{\mathrm{T}}]$. The matcher $m_s(y, y_k^{\mathrm{T}}) \in \{0,1\}$ approximates $\mathbf{1}\{\phi_s(y) = a_k^{\mathrm{T}}\}$. Invalid outputs never match. Define

$$c(s,y) = \sum_{k=1}^{K} m_s(y, y_k^{\mathrm{T}}), \qquad \widehat{r}_{\mathrm{T}}(s,y) = \frac{c(s,y)}{K}. \tag{5}$$

Repeated teacher samples contribute separately to the count. Under independent teacher draws, $c(s,y)/K$ is an unbiased estimate of the probability that the matcher judges y to agree with a sampled teacher response. With accurate matching, this probability equals $r_{\mathrm{T}}(s,\phi_s(y))$. Appendix A.3 details sample retention and caching.

The matcher compares action content, not reasoning. For tool calls, it checks tool names and canonicalized arguments programmatically, using a frozen LLM matcher for remaining argument differences. For communicative responses, normalized exact matches are accepted. Otherwise, the LLM is instructed to assess agreement in intent and factual content. Each candidate is compared with every teacher sample. Tool calls, messages, and invalid outputs remain in one training group. Note that semantic agreement reflects consistency with the teacher, not necessarily action correctness or task success. Appendix A.3 specifies matcher criteria and inputs.

Although c/K estimates agreement, three samples provide only coarse frequency estimates. To limit sensitivity to sampled counts, we assign a unit reward to the first match and a smaller frequency bonus to each additional match. For $K > 1$, we use

$$r_\lambda(s, y) = \begin{cases} -1, & y \text{ contains an invalid action,} \\ 0, & y \text{ contains a valid action and } c(s, y) = 0, \\ 1 + \lambda \dfrac{c(s, y) - 1}{K - 1}, & c(s, y) > 0. \end{cases} \tag{6}$$

With $\lambda = 0.5$ and $[A, A, B]$, a response matching A earns 1.25 and one matching B earns 1: the additional A match contributes 0.25, compared with 1 for the first match. Invalid outputs incur a separate penalty for protocol violations. The reward in Equation 6 modifies the original likelihood objective. Appendix A.2 derives the objective and its single-sample, binary, and raw-count cases.

2.3 State-group rollout and optimization

State-group rollout. We use these rewards both to select actions for environment interaction and to optimize the student from same-state candidate groups (Figure 3). At state s_g, group g contains $N = 4$ responses indexed by i, sampled as $y_{g,i} \sim \mu_{\theta_{\text{old}}}(\cdot \mid s_g)$ with frozen θ_{old}. The decoder's sampling distribution μ includes temperature and truncation controls. All candidates are scored against the same teacher multiset, yielding $r_{g,i} = r_\lambda(s_g, y_{g,i})$. We execute the parsed action in $y_{g,i^\star}$, with $i^\star \in \arg\max_i r_{g,i}$. In AWM/EnvScaler, ties favor actions that differ from the previously selected action, when available, with uniform sampling among the remaining candidates.

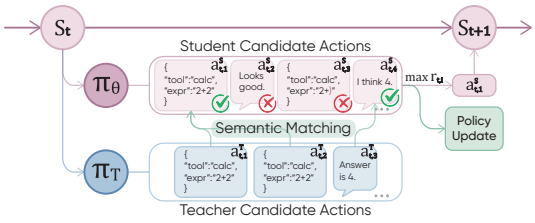


Figure 3: AOPD at one state. Student candidates are scored against a shared teacher multiset. One highest-reward action is committed. All eligible candidates contribute to training.

Only the selected action is executed, but eligible unselected responses still contribute to training. Repeated selection produces one trajectory per task with a teacher-guided visited-state distribution, rather than the decoder's ordinary single-sample rollout distribution.

Policy optimization. Whereas execution selects a highest-reward action, policy updates compare rewards within each state using group-relative advantages and a clipped surrogate (Shao et al., 2024; Schulman et al., 2017). Let $\mathcal{I}_g$ index eligible, non-padding candidates in group g. Eligible candidates have available reward supervision and need not contain valid actions. Groups with fewer than two eligible candidates are masked. Equal-reward groups can advance the environment but have zero centered advantages and are also masked. Let $\mathcal{B}_+$ contain the remaining candidate indices (g, i). We center rewards within each group and whiten across the update batch:

$$\bar{r}_g = \frac{1}{|\mathcal{I}_g|} \sum_{j \in \mathcal{I}_g} r_{g,j}, \qquad \widetilde{A}_{g,i} = r_{g,i} - \bar{r}_g, \qquad A_{g,i} = \frac{\widetilde{A}_{g,i}}{\sigma_{\mathcal{B}} + \epsilon}. \tag{7}$$

Here $\sigma_{\mathcal{B}}$ is the population standard deviation of $\widetilde{A}_{g,i}$ over $(g, i) \in \mathcal{B}_+$, and $\epsilon > 0$ stabilizes division. The shared denominator preserves the relative magnitudes of centered rewards across groups.

Let $\mathcal{T}_+$ contain active generated-token indices (g, i, ℓ) after masking. Using temperature-scaled model probabilities, define the token-level probability ratio

$$\rho_{g,i,\ell}(\theta) = \frac{\pi_\theta(y_{g,i,\ell} \mid s_g, y_{g,i,<\ell})}{\pi_{\theta_{\text{old}}}(y_{g,i,\ell} \mid s_g, y_{g,i,<\ell})}. \tag{8}$$

This ratio does not correct for top-p/top-k truncation in μ (Appendix A.4). Following the token-level expansion in Section 2.1, $A_{g,i}$ weights every active token in $y_{g,i}$.

With clipping width $\eta > 0$, the token-mean surrogate is

$$J_{\text{clip}}(\theta) = \frac{1}{|\mathcal{T}_+|} \sum_{(g,i,\ell)\in\mathcal{T}_+} \min\{\rho_{g,i,\ell}(\theta)A_{g,i},\ \text{clip}(\rho_{g,i,\ell}(\theta), 1-\eta, 1+\eta)A_{g,i}\}. \tag{9}$$

Policy optimization follows collection (Figure 3). With finite-group baselines, whitening, masking, and clipping, the surrogate gradient is not an unbiased estimate of Equation 4. Appendix A.4 provides details on dual clipping and execution. Training uses per-state rewards rather than trajectory returns. Terminal outcomes are used only to monitor success.

3 Experiments

We evaluate AOPD's cross-benchmark generalization, student–teacher scaling, and performance relative to other distillation methods. We then use ablations and case studies to examine its components and decision-making behavior, and explore Self-AOPD with privileged teacher context.

3.1 Experimental design

Training data. We trained on Agent World Model (AWM; Wang et al., 2026b) and EnvScaler (Song et al., 2026). AWM provides executable tools and persistent backend state; EnvScaler adds a simulated user to the interaction loop. The curated pool comprised **5,942 AWM tasks from 794 environments** and **755 EnvScaler tasks from 49 environments**: **6,697 tasks** in total. Screening assessed environment integrity and static task feasibility, without requiring successful expert rollouts (Appendix B.1).

Models and training. We paired Qwen3-4B and Qwen3-8B students (Yang et al., 2025) with Qwen3.7-Flash (Alibaba Cloud, 2026) and DeepSeek-V4-Flash (DeepSeek-AI, 2026) action teachers in a 2×2 design. Each configuration used 100 training steps and 64 task starts per step (59 AWM, 5 EnvScaler), with $N = 4$ student candidates, $K = 3$ teacher samples per state, and frequency bonus scale $\lambda = 0.5$. Pool size and the number of unique tasks visited are distinct quantities. Appendix B.2 details settings and auxiliary models.

Evaluation. We evaluated transfer on **BFCL Multi-turn** (Patil et al., 2025), **ToolSandbox**'s verified-local-only subset (Lu et al., 2025), **AppWorld** Normal and Challenge (Trivedi et al., 2024), and τ^2**-bench** Airline, Retail, and Telecom (Barres et al., 2025). These tasks were excluded from AWM/EnvScaler training. Tau comparisons, ablations, and Self-AOPD used official train/test splits, reported separately. Appendix B.3 details coverage, trials, and interaction protocols.

Metrics. We report BFCL accuracy, ToolSandbox similarity (not binary success), AppWorld task/scenario goal completion (TGC/SGC), and Tau $\text{pass}^1/\text{pass}^4$: single-trial success and four-trial all-success reliability, not best-of-four success (Yao et al., 2025). The *average* weights the four benchmarks equally, using mean TGC for AppWorld and mean domain pass^1 for Tau. Appendix B.3 gives the estimators.

3.2 Main results

Table 1 compares distilled students with base models and directly evaluated teachers.

Cross-benchmark generalization. After training on AWM and EnvScaler, all four configurations improved the four-benchmark average over their base models by 3.4–6.6 percentage points. Gains extended across BFCL accuracy, ToolSandbox similarity, and AppWorld task completion. These gains on benchmarks excluded from training demonstrate that action-level distillation transfers beyond the training environments, improving average performance across diverse agent tasks.

Table 1: Cross-benchmark transfer after 100 AOPD steps (%). + identifies the teacher. AppWorld: task/scenario success on Normal and Challenge. Scenario success requires success on all task variants. Tau: $\text{pass}^1/\text{pass}^4$. Bold/underlined: best/second-best distinct scores per student-size group, including Base; ties share marks. Rankings and green average gains over Base (percentage points) use unrounded scores. Role-specific generation settings: Table 8.

Model	BFCL	Tool Sandbox	AppWorld		τ^2-Bench			Average
			Normal	Challenge	Airline	Retail	Telecom	
Direct Teacher Evaluation								
Qwen3.7-Flash	58.0	76.2	70.2/55.4	36.5/23.0	78.0/54.0	55.9/18.4	90.6/71.9	65.6
DeepSeek-V4-Flash	57.0	72.3	84.5/75.0	74.8/54.7	77.0/52.0	51.8/15.8	98.9/95.6	71.2
Qwen3-4B Student								
Base	34.3	55.5	9.5/3.6	2.6/0.7	34.4/10.8	27.9/8.8	20.5/4.4	30.9
+ Qwen3.7-Flash	42.3	61.8	**14.3/8.9**	4.8/0.7	34.0/12.0	35.4/9.1	22.6/7.4	36.1 +5.2
+ DeepSeek-V4-Flash	**44.1**	**63.3**	13.1/5.4	**6.5/2.2**	**37.2/18.0**	**36.1/10.0**	**24.2/15.3**	**37.4** +6.6
Qwen3-8B Student								
Base	39.9	57.7	15.5/5.4	4.6/0.7	35.2/10.0	37.7/13.0	32.3/**14.6**	35.7
+ Qwen3.7-Flash	45.0	**62.5**	16.7/8.9	**7.7**/1.4	38.4/**20.8**	36.8/10.2	**33.9**/12.3	39.0 +3.4
+ DeepSeek-V4-Flash	**45.5**	**62.5**	**17.9/10.7**	**7.7/2.9**	**43.2/20.8**	**39.5/13.2**	33.7/12.1	**39.9** +4.2

Scaling with student size. With either teacher, Qwen3-8B achieved a higher final average than Qwen3-4B under the same task-start and training-step budgets. We used the same candidate count, teacher sample count, and reward formulation for both students. This pattern supports empirical scaling across the two tested sizes, with larger students yielding stronger distilled agents.

Scaling with teacher capability. DeepSeek-V4-Flash achieved a higher direct-evaluation average than Qwen3.7-Flash and yielded higher student averages at both sizes (Figure 1), supporting empirical scaling with teacher capability. The DeepSeek-to-Qwen results also demonstrate effective cross-family distillation through an action-level interface that requires neither a shared model family nor tokenizer alignment. This allows teacher selection to prioritize decision quality rather than compatibility with the student's token space.

3.3 Comparison with distillation baselines

We compared AOPD, offline distillation (KD), and sampled-token reverse-KL OPD using a Qwen3-4B student, a Qwen3-32B teacher, and the same 178 Tau training tasks. All methods used 24 task starts per step—eight per domain—and were evaluated at step 50. KD collected four independent teacher trajectories per task start and fine-tuned on all reasoning and action/message tokens from successful trajectories. OPD collected four student trajectories per task start and used teacher log-probabilities to supervise sampled tokens. AOPD used state-group rollout, sampling four candidates at each visited state while advancing one trajectory per task start. It used $K = 3$ teacher samples per state and $\lambda = 0.5$ (Appendix B.4).

KD achieved the highest test average pass^4 (19.2), indicating greater repeated-success reliability on limited tasks and states. Conversely, OPD and AOPD achieved higher test average pass^1 than KD. This pattern demonstrates the advantages of on-policy supervision: reducing mismatch between teacher demonstrations and student-encountered states, and improving performance across a broader range of student tasks. AOPD achieved higher training-set averages on both metrics and exceeded OPD's test average pass^1 by 0.4 percentage points, although its test average pass^4 was lower. While these results do not establish uniform superiority, AOPD offers a competitive alternative with a more flexible teacher interface. Its generation-only interface requires neither teacher token log-probabilities nor tokenizer alignment, and does not require a shared model family. The cross-family transfer demonstrated in Section 3.2 complements this same-family comparison.

Table 2: Tau distillation comparison: $\mathrm{pass}^1/\mathrm{pass}^4$ (%). Qwen3-4B students, step 50; Qwen3-32B teacher. All tasks are evaluated for four trials. The average equally weights domains per split. AOPD uses the same run as in Table 3. Bold/underlined: best/second-best distinct scores (unrounded); teacher excluded.

Model	Airline		Retail		Telecom		Average	
	Train (30)	Test (20)	Train (74)	Test (40)	Train (74)	Test (40)	Train	Test
Qwen3-32B	49.2/16.7	43.8/15.0	46.0/16.2	46.9/15.0	25.7/13.5	41.9/30.0	40.3/15.5	44.2/20.0
Base	35.3/12.7	33.0/8.0	27.8/7.3	28.0/11.5	17.6/4.3	26.0/4.5	26.9/8.1	29.0/8.0
KD	38.3/**20.0**	37.5/**20.0**	37.5/**13.5**	33.8/**12.5**	25.3/**13.5**	**41.3/25.0**	33.7/**15.7**	37.5/**19.2**
OPD	40.0/13.3	**41.3/20.0**	40.2/10.8	37.5/10.0	22.6/10.8	40.0/22.5	34.3/11.7	39.6/17.5
AOPD	**42.5**/16.7	40.0/15.0	**42.9**/10.8	**39.4/12.5**	**28.4/13.5**	40.6/15.0	**37.9**/13.7	**40.0**/14.2

Table 3: Tau ablations: $\mathrm{pass}^1/\mathrm{pass}^4$ (%). All tasks are evaluated for four trials. Parentheses: task counts. The average equally weights domains within each split. Bold/underlined: best/second-best distinct scores (unrounded), including Base; ties share marks.

Model	Airline		Retail		Telecom		Average	
	Train (30)	Test (20)	Train (74)	Test (40)	Train (74)	Test (40)	Train	Test
Base	35.3/12.7	33.0/8.0	27.8/7.3	28.0/11.5	17.6/4.3	26.0/4.5	26.9/8.1	29.0/8.0
Validity Reward	**42.5/16.7**	30.0/10.0	23.7/8.1	26.3/10.0	25.3/**14.9**	36.3/10.0	30.5/13.2	30.8/10.0
Binary Reward	35.8/10.0	33.8/10.0	35.8/8.1	36.3/**17.5**	**28.4**/13.5	40.6/**15.0**	33.3/10.5	36.9/**14.2**
Random Commit	34.2/3.3	27.5/10.0	41.6/**16.2**	**40.6**/15.0	24.0/8.1	**44.4**/12.5	33.2/9.2	37.5/12.5
AOPD	**42.5/16.7**	**40.0/15.0**	**42.9**/10.8	39.4/12.5	**28.4**/13.5	40.6/**15.0**	**37.9/13.7**	**40.0/14.2**

3.4 Ablation studies

Ablations shared the settings in Section 3.3, with $N = 4$ and $K = 3$. Validity Reward replaced the semantic training reward with action validity, while retaining semantic scores for action selection. Binary Reward removed the frequency bonus from Equation 6, using binary agreement for both training and action selection. Random Commit retained AOPD's reward but selected uniformly from all candidates, including invalid or zero-reward actions. Appendix B.4 details the variants and their effects on trainable state groups.

Table 3 reports results on the training set and 100 held-out tasks. AOPD outperforms all three variants in average single-trial success on both splits, supporting the combined design. The largest drop occurs with Validity Reward, which assigns identical rewards to all valid candidates and thus provides no learning signal in all-valid groups. Semantic rewards can instead distinguish these candidates by their agreement with teacher actions, highlighting the value of supervising action meaning beyond protocol compliance.

3.5 Pattern analysis

We compared Qwen3-4B Base with the same student after 100 AOPD steps using DeepSeek-V4-Flash as teacher. Among tasks evaluated over four trials, we selected those with at least two additional successes after training. Table 4 summarizes one Base-failure/AOPD-success pair matched by task and trial index in each Tau domain, using replies, tool calls, and environment results.

In these selected cases, the contrasts go beyond well-formed tool calls: the trained student connected policy constraints and task goals to the actions needed for an appropriate final state. These behavioral examples establish neither benchmark-wide prevalence nor internal reasoning mechanisms.

Table 4: Selected positive cases: Qwen3-4B Base versus its AOPD counterpart after 100 steps with DeepSeek-V4-Flash. Fractions: successes in four trials (Base → AOPD); text: paired trajectories. Outcomes denote evaluation results, not training rewards.

Pattern / Task Context	Qwen3-4B Base	Qwen3-4B + AOPD
Following Policy Constraints Airline ($2/4 \rightarrow 4/4$). The user requests a flight change, but basic economy bookings cannot be modified.	Reads that the booking is basic economy, yet modifies the flight through the API, violating the policy. **Outcome: Failure.**	Explains the restriction and calls the transfer tool when the user requests escalation. Success means keeping the booking unchanged, not rebooking. Prohibited modifications: $2/4 \rightarrow 0/4$. **Outcome: Success.**
Tracking the Correct Target Retail ($0/4 \rightarrow 3/4$). Change the target order's address and speaker. Retrieve the new address in New York from another order.	Treats the source order's New York address as if it already belonged to the target. Changes only the speaker. The target's address remains in Los Angeles. **Outcome: Failure.**	Retrieves the source address, then updates the target's address followed by its speaker. The returned order contains both the New York address and the requested green speaker. **Outcome: Success.**
Completing and Verifying Actions Telecom ($0/4 \rightarrow 3/4$). Restore service after an overdue-bill suspension and an unseated SIM. Payment alone does not reactivate the line.	Promises restoration after payment but never calls `resume_line`. The user pays and reboots, but service remains disconnected and the agent transfers the case. **Outcome: Failure.**	Recovers from an invalid tool call, resumes the line with `resume_line`, and guides the *user* to reseat the SIM. User-side tool output confirms connection, and the evaluator verifies that the debt is cleared. **Outcome: Success.**

Table 5: Self-AOPD on Tau: $\text{pass}^1/\text{pass}^4$ (%), using Qwen3-4B. Privileged Teacher: initial Base with privileged prompting; Self-AOPD: public context only at step 50. All tasks are evaluated for four trials. The average equally weights domains within each split. Bold/underlined: best/second-best of Base and Self-AOPD; privileged reference excluded.

Model	Airline		Retail		Telecom		Average	
	Train (30)	Test (20)	Train (74)	Test (40)	Train (74)	Test (40)	Train	Test
Privileged Teacher	51.7/33.3	35.0/15.0	38.2/6.8	38.1/17.5	20.6/8.1	37.5/20.0	36.8/16.1	36.9/17.5
Base	<u>35.3</u>/<u>12.7</u>	<u>33.0</u>/<u>8.0</u>	<u>27.8</u>/<u>7.3</u>	28.0/**11.5**	<u>17.6</u>/<u>4.3</u>	<u>26.0</u>/<u>4.5</u>	<u>26.9</u>/<u>8.1</u>	<u>29.0</u>/<u>8.0</u>
Self-AOPD	**44.2/26.7**	**35.0/10.0**	**35.1/8.1**	**35.6**/<u>10.0</u>	**19.3/5.4**	**31.3/10.0**	**32.9/13.4**	**34.0/10.0**

3.6 SELF-AOPD

We further examine whether a model can provide useful action-level supervision for itself when given privileged task information. Self-AOPD shared current Qwen3-4B weights between teacher and student, as in OPSD (Zhao et al., 2026), but independently sampled their responses. Teacher responses were refreshed from the updated shared model at each collection step and held fixed during subsequent policy optimization. The student received only public interaction context, whereas the teacher additionally received action guidance, summaries of private user goals and constraints, and reference actions and target outcomes. Training used action-agreement rewards on student-generated candidates only. Appendix B.5 provides implementation details.

After 50 steps, Self-AOPD improved test average $\text{pass}^1/\text{pass}^4$ over Base by 5.0/2.0 percentage points (Table 5). Single-trial success also improved across all three domains. Unlike the privileged-teacher reference, the trained student was evaluated using only public context, showing that the student retained benefits from the privileged guidance after training. Together, these results support the feasibility of using a same-model teacher given privileged context for action-level distillation.

4 Related work

Distillation and self-distillation. Classical distillation transfers predictive distributions (Hinton et al., 2015), while sequence distillation imitates generated outputs (Kim & Rush, 2016). Offline imitation uses token-level losses on fixed demonstrations that may not cover student-induced prefixes. GKD and MiniLLM instead supervise student-generated sequences through token-level divergences (Agarwal et al., 2024; Gu et al., 2024). These objectives require teacher distributions or sampled-token log-probabilities. Guidance can become unreliable under prefix drift or special-token mismatch (Fu et al., 2026). Cross-tokenizer OPD transfers probabilistic supervision through explicit alignment (Niu et al., 2026; Wang et al., 2026a). Privileged self-distillation augments teacher context: On-Policy Self-Distillation (OPSD) minimizes token-level divergence on student rollouts using a shared-weight teacher conditioned on reference solutions (Zhao et al., 2026). SDPO distills feedback-conditioned predictions (Hübotter et al., 2026), while privileged-information distillation studies shared-parameter training, including on-policy reverse KL (Penaloza et al., 2026). Self-AOPD instead uses sampled-action agreement.

Black-box and semantic distillation. GAD jointly trains a discriminator to reward teacher-like responses (Ye et al., 2025), whereas ROPD evaluates responses against rubrics derived from teacher–student contrasts (Fang et al., 2026). On-policy Verbal Distillation (OVD) uses verbal teacher scores to evaluate reasoning quality at step and trajectory levels (Xiong et al., 2026). OmniOPD compares selected student reasoning chunks with teacher continuations sampled at student prefixes (Zhou et al., 2026). MENTOR rewards trajectory-wide tool-set agreement, final correctness, and execution validity (Choi et al., 2025). Action Semantics Learning (ASL) derives a semantic-agreement policy gradient and uses a world model to compare UI transitions induced by generated and reference actions. Its RL variant combines teacher-action agreement with environment rewards (Tang et al., 2026). AOPD instead matches immediate tool calls or communicative responses against teacher samples at the same state. Its matcher is instructed to assess agreement, not correctness. AOPD requires no reasoning-text matching, complete reference trajectories, or terminal rewards.

Interactive agent learning and process supervision. DAgger queries experts at learner-visited states (Ross et al., 2011). SCoRe provides local teacher corrections before supervised and short-horizon RL refinement (Lyu et al., 2026). Guided-OPD gradually withdraws teacher turns through a rollout curriculum; ATOD combines OPD and RL with annealed coefficients and turn-level weighting (Li et al., 2026; Tan et al., 2026). GiGPO retrospectively groups repeated states across trajectories and derives relative advantages from returns (Feng et al., 2025). AOPD scores directly sampled same-state candidates against a shared teacher multiset, without estimating returns.

5 Conclusion and limitations

AOPD decouples teacher decision supervision from student token realization through semantic agreement at student-encountered states, requiring only generation access to the teacher. Across tested configurations, it improves average performance on unseen benchmarks, with higher final averages from larger students and stronger teachers. AOPD also achieves competitive performance against offline distillation and token-level OPD. Ablations support semantic rewards over validity rewards. Self-AOPD improves held-out performance through same-model guidance with privileged task information. Together, these findings support semantic actions as a unit of supervision for agent distillation.

Three limitations warrant attention. First, agreement does not guarantee correctness: teacher errors or suboptimal decisions can become learning targets. Second, LLM matching of communicative responses and tool-argument differences can produce false positives or negatives, misassigning rewards. Third, fixed-size teacher and student sample sets may contain few semantically matching actions when plausible decisions are dispersed, as may occur in long-horizon interactions. Increasing K and N could improve coverage at greater generation and matching cost, but the benefit in this regime remains untested.

AI USE STATEMENT

The authors developed the method and designed all experiments. Generative AI tools assisted with manuscript preparation, literature review, implementation, debugging, derivation checks, and result analysis, and provided feedback on methodological details. The use of language models for task screening, teacher supervision, semantic matching, and user simulation is documented in the experimental protocol. All final research decisions were made by the authors, who take responsibility for the manuscript and its findings.

REPRODUCIBILITY STATEMENT

Appendices A and B document the derivations, algorithm, data preparation, and experimental protocols. Code, data, and model checkpoints for the core results will be made available at `https://anonymous.4open.science/r/AOPD`.

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*, 2024. URL `https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html`.

Alibaba Cloud. Qwen3.7-Flash model documentation. Model Studio documentation, 2026. URL `https://www.alibabacloud.com/help/en/model-studio/qwen3-7-flash`. Accessed September 24, 2026.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating conversational agents in a dual-control environment. arXiv:2506.07982, 2025. URL `https://arxiv.org/abs/2506.07982`.

ChangSu Choi, Hoyun Song, Dongyeon Kim, Minkyung Cho, WooHyeon Jung, Sunjin Park, NohHyeob Bae, Seona Yu, and KyungTae Lim. MENTOR: Reinforcement learning via flexible teacher-optimized rewards for tool-use distillation. arXiv:2510.18383v4, 2025. URL `https://arxiv.org/abs/2510.18383v4`.

DeepSeek-AI. DeepSeek-V4-Flash. Model card, 2026. URL `https://huggingface.co/deepseek-ai/DeepSeek-V4-Flash`. Accessed September 24, 2026.

Junfeng Fang, Zhepei Hong, Mao Zheng, Mingyang Song, Gengsheng Li, Houcheng Jiang, Dan Zhang, Haiyun Guo, Xiang Wang, and Tat-Seng Chua. Rubric-based on-policy distillation. arXiv:2605.07396v1, 2026. URL `https://arxiv.org/abs/2605.07396v1`.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for LLM agent training. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL `https://proceedings.neurips.cc/paper_files/paper/2025/hash/420c9f777c0b4f78d515e53cf74d58b2-Abstract-Conference.html`.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Jiacai Liu, Zhuo Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. arXiv:2603.25562v2, 2026. URL `https://arxiv.org/abs/2603.25562v2`.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024. URL `https://proceedings.iclr.cc/paper_files/paper/2024/hash/8ac015d409635f196f9e3e9dcfb9a94e-Abstract-Conference.html`.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv:1503.02531, 2015. URL `https://arxiv.org/abs/1503.02531`.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. arXiv:2601.20802v2, 2026. URL https://arxiv.org/abs/2601.20802v2.

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. doi: 10.18653/v1/D16-1139. URL https://aclanthology.org/D16-1139/.

Gengsheng Li, Mao Zheng, Mingyang Song, Ruiqi Liu, Tianyu Yang, Jie Sun, Qiyong Zhong, Haiyun Guo, Junfeng Fang, Dan Zhang, and Jinqiao Wang. On-policy distillation with curriculum turn-level guidance for multi-turn agents. arXiv:2606.15912v2, 2026. URL https://arxiv.org/abs/2606.15912v2.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1160–1183, 2025. doi: 10.18653/v1/2025.findings-naacl.65. URL https://aclanthology.org/2025.findings-naacl.65/.

Yuanjie Lyu, Chengyu Wang, Jun Huang, and Tong Xu. Student-centered distillation narrows the agentic gap between small and large LLMs. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2509.14257v3.

Yifan Niu, Han Xiao, Dongyi Liu, Zelong Wang, Dihong Gong, Yasheng Wang, and Jia Li. Breaking the tokenizer barrier: On-policy distillation across model families. arXiv:2606.09456, 2026. URL https://arxiv.org/abs/2606.09456.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392, 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. arXiv:2602.04942v3, 2026. URL https://arxiv.org/abs/2602.04942v3.

Qwen Team. Qwen3.5-9B. Model card, 2026a. URL https://huggingface.co/Qwen/Qwen3.5-9B. Accessed September 24, 2026.

Qwen Team. Qwen3.8-27B. Model card, 2026b. URL https://huggingface.co/Qwen/Qwen3.8-27B. Accessed September 24, 2026.

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, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. arXiv:1707.06347, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. arXiv:2402.03300, 2024. URL https://arxiv.org/abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025. doi: 10.1145/3689031.3696075. URL https://doi.org/10.1145/3689031.3696075.

Xiaoshuai Song, Haofei Chang, Guanting Dong, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. EnvScaler: Scaling tool-interactive environments for LLM agent via programmatic synthesis. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 8326–8357, 2026. doi: 10.18653/v1/2026.findings-acl.407. URL https://aclanthology.org/2026.findings-acl.407/.

Qitai Tan, Zefang Zong, Mo Li, Yipeng Shi, Yang Li, and Peng Chen. ATOD: Annealed turn-aware on-policy distillation for multi-turn agentic tasks. arXiv:2606.27814v6, 2026. URL https://arxiv.org/abs/2606.27814v6.

Bohan Tang, Dezhao Luo, Jianheng Liu, Jingxuan Chen, Shaogang Gong, Jianye Hao, Jun Wang, and Kun Shao. Beyond syntax: Action semantics learning for app agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 9444–9454, 2026. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Tang_Beyond_Syntax_Action_Semantics_Learning_for_App_Agents_CVPRF_2026_paper.html.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16022–16076, 2024. doi: 10.18653/v1/2024.acl-long.850. URL https://aclanthology.org/2024.acl-long.850/.

Hao Wang, Kun Yuan, Wenlin Zhong, Minglei Zhang, Han Xiao, Ming Sun, and Honggang Qi. Cross-tokenizer on-policy distillation via byte-prefix marginalization. arXiv:2607.22334, 2026a. URL https://arxiv.org/abs/2607.22334.

Zhaoyang Wang, Canwen Xu, Boyi Liu, Yite Wang, Siwei Han, Zhewei Yao, Huaxiu Yao, and Yuxiong He. Agent world model: Infinity synthetic environments for agentic reinforcement learning. In *International Conference on Machine Learning*, 2026b. URL https://arxiv.org/abs/2602.10090v3.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

Jing Xiong, Hui Shen, Shansan Gong, Yuxin Cheng, Jianghan Shen, Chaofan Tao, Haochen Tan, Haoli Bai, Lifeng Shang, and Ngai Wong. OVD: On-policy verbal distillation. arXiv:2601.21968, 2026. URL https://arxiv.org/abs/2601.21968.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. arXiv:2505.09388, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=roNSXZpUDN.

Deheng Ye, Zhao Liu, Mingfei Sun, Bei Shi, Peilin Zhao, Hao Wu, Hongsheng Yu, Shaojie Yang, Xipeng Wu, Qingwei Guo, Qiaobo Chen, Yinyuting Yin, Hao Zhang, Tengfei Shi, Liang Wang,

Qiang Fu, Wei Yang, and Lanxiao Huang. Mastering complex control in MOBA games with deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6672–6679, 2020. doi: 10.1609/aaai.v34i04.6144. URL https://ojs.aaai.org/index.php/AAAI/article/view/6144.

Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-box on-policy distillation of large language models. arXiv:2511.10643v3, 2025. URL https://arxiv.org/abs/2511.10643v3.

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:2601.18734v3, 2026. URL https://arxiv.org/abs/2601.18734v3.

Yuhang Zhou, Lizhu Zhang, Yifan Wu, Mingyi Wang, Bo Peng, Jiayi Liu, Xiangjun Fan, and Zhuokai Zhao. OmniOPD: Logit-free on-policy distillation via speculative verification. arXiv:2606.01476v2, 2026. URL https://arxiv.org/abs/2606.01476v2.

A Derivations and implementation details

A.1 From teacher-action likelihood to REINFORCE

Assumptions and induced actions. Let $\mathcal{Y}(s)$ contain student response sequences, including termination tokens. At each state, semantic action classes partition these sequences: $\phi_s(y)$ maps equivalent realizations to the same action, and protocol-invalid outputs to an invalid outcome $\perp$. The induced $\pi_\theta(a \mid s)$ is normalized over the complete action space, including $\perp$. Teacher targets from $\pi_{\mathrm{T}}(a \mid s)$ are valid actions. Summation over actions counts each semantic class once, not each expression of that class.

Assume a normalized, differentiable response policy on a fixed support, with a finite vocabulary and bounded response length. These conditions permit exchanging differentiation and summation. The semantic mapping ϕ_s, teacher distribution π_{T}, and collected-state distribution d_{old} remain fixed during each update. Teacher action samples suffice; teacher token probabilities are not needed.

Expanding the likelihood. For a teacher semantic action, all its student realizations contribute to its likelihood,

$$\pi_\theta(a^{\mathrm{T}} \mid s) = \sum_{y \in \mathcal{Y}(s)} \pi_\theta(y \mid s) \mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\}. \tag{10}$$

Consequently,

$$\begin{aligned} J_{\mathrm{AOPD}}(\theta) &= \mathbb{E}_{s \sim d_{\mathrm{old}}} \sum_{a^{\mathrm{T}}} \pi_{\mathrm{T}}(a^{\mathrm{T}} \mid s) \sum_{y} \pi_\theta(y \mid s) \mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\} \\ &= \mathbb{E}_{s \sim d_{\mathrm{old}}} \sum_{y} \pi_\theta(y \mid s) r_{\mathrm{T}}(s, \phi_s(y)). \end{aligned} \tag{11}$$

Differentiating the student factor gives

$$\begin{aligned} \nabla_\theta J_{\mathrm{AOPD}}(\theta) &= \mathbb{E}_s \sum_{y} r_{\mathrm{T}}(s, \phi_s(y)) \nabla_\theta \pi_\theta(y \mid s) \\ &= \mathbb{E}_{s,y}[r_{\mathrm{T}}(s, \phi_s(y)) \nabla_\theta \log \pi_\theta(y \mid s)] \\ &= \mathbb{E}_{s,a^{\mathrm{T}},y} \left[\mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\} \sum_{\ell=1}^{|y|} \nabla_\theta \log \pi_\theta(y_\ell \mid s, y_{<\ell}) \right]. \end{aligned} \tag{12}$$

This score-function estimator (Williams, 1992) uses semantic action agreement without evaluating the marginal $\pi_\theta(a^{\mathrm{T}} \mid s)$ explicitly. The likelihood and agreement objectives are identical under this semantic action definition.

An action-independent baseline $b(s)$ leaves this gradient unchanged because $\mathbb{E}_y[\nabla_\theta \log \pi_\theta(y \mid s)] = 0$. A sample group mean instead includes each candidate's own reward. For independent candidates and an otherwise unmodified estimator, subtracting that mean scales the expected gradient by $(N-1)/N$. Whitening, masking, token normalization, and clipping further distinguish the practical surrogate from the ideal gradient.

Expected likelihood versus expected log-likelihood. Where $\pi_\theta(a^{\mathrm{T}} \mid s) > 0$, replacing the expected likelihood in Equation 1 with $\mathbb{E}_{s,a^{\mathrm{T}}}[\log \pi_\theta(a^{\mathrm{T}} \mid s)]$ yields the following gradient:

$$\mathbb{E}_{s,a^{\mathrm{T}},y} \left[\frac{\mathbf{1}\{\phi_s(y) = a^{\mathrm{T}}\}}{\pi_\theta(a^{\mathrm{T}} \mid s)} \nabla_\theta \log \pi_\theta(y \mid s) \right]. \tag{13}$$

The inverse-likelihood factor is absent from AOPD. Expected agreement also need not preserve teacher modes: for disjoint actions A, B with teacher probabilities $0.7, 0.3$, a student placing all mass on A maximizes the unregularized likelihood objective.

Algorithm 1 Agentic On-Policy Distillation (AOPD)

Require: Student π_θ, teacher π_{T}, matcher m_s, tasks, $K = 3$, $N = 4$, $\lambda = 0.5$

for each training iteration **do**
  Freeze $\theta_{\mathrm{old}} \leftarrow \theta$; initialize state-group buffer $\mathcal{B} \leftarrow \emptyset$
  for each sampled task **do**
    Reset the environment and obtain public state s
    while task is active and interaction budget remains **do**
      Set group index $g \leftarrow |\mathcal{B}| + 1$ and state $s_g \leftarrow s$
      Sample teacher responses $\mathcal{M}_{\mathrm{T}}(s_g) = [y_1^{\mathrm{T}}, \ldots, y_K^{\mathrm{T}}]$ using π_{T}
      Sample $y_{g,i} \sim \mu_{\theta_{\mathrm{old}}}(\cdot \mid s_g)$ and retain parsed actions, for $i = 1, \ldots, N$
      Compute $c(s_g, y_{g,i})$ and $r_{g,i} = r_\lambda(s_g, y_{g,i})$ (Equation 5; Equation 6)
      Add the complete state group $(s_g, \{y_{g,i}, r_{g,i}\}_{i=1}^N)$ to $\mathcal{B}$
      Select $i^\star \in \arg\max_i r_{g,i}$; execute the parsed action in $y_{g,i^\star}$ and update s
    end while
  end for
  Exclude equal-reward groups; center rewards per group and whiten across the batch (Equation 7)
  Broadcast advantages to generated tokens and update θ with the clipped objective (Equation 9)
end for

A.2 Objective induced by the frequency bonus

Fix a state s, a nominal teacher sample count $K > 1$, and the retained teacher-response multiset $\mathcal{M} = \mathcal{M}_{\mathrm{T}}(s)$ with $1 \le |\mathcal{M}| \le K$. Write $c(y) = c(s, y) = \sum_{k=1}^{|\mathcal{M}|} m_s(y, y_k^{\mathrm{T}})$, suppressing the fixed state and multiset. Let

$$\begin{aligned} C_\theta(s; \mathcal{M}) &= \mathbb{E}_{y \sim \pi_\theta}[c(y)], \\ U_\theta(s; \mathcal{M}) &= \mathbb{P}_{y \sim \pi_\theta}[c(y) > 0], \\ V_\theta(s) &= \mathbb{P}_{y \sim \pi_\theta}[y \text{ contains an invalid action}]. \end{aligned} \tag{14}$$

Here C_θ is the expected match count, U_θ the any-match probability, and V_θ the invalid-action probability. With accurate matching, $C_\theta = \sum_k \pi_\theta(a_k^{\mathrm{T}} \mid s)$, but the decomposition below does not require this equality. Invalid actions have $c(y) = 0$. Any-match probability does not measure coverage of distinct teacher actions. Writing $\alpha = \lambda/(K-1)$, Equation 6 is exactly

$$r_\lambda(s, y) = \alpha c(y) + (1 - \alpha)\mathbb{I}\{c(y) > 0\} - \mathbb{I}\{y \text{ contains an invalid action}\}. \tag{15}$$

Its expected-reward objective is therefore

$$J_\lambda(\theta \mid s, \mathcal{M}) = \alpha C_\theta(s; \mathcal{M}) + (1 - \alpha) U_\theta(s; \mathcal{M}) - V_\theta(s). \tag{16}$$

The same score-function derivation applies to r_λ. For $0 \le \lambda \le K - 1$, the positive terms combine expected match count and any-match probability with nonnegative weights. Beyond this range, the identity still holds but the any-match coefficient is negative. For $0 < \lambda < K - 1$, each additional match contributes $\alpha < 1$, whereas the first match earns one. This downweights additional matches relative to the first match, but does not imply reduced variance of the normalized policy update. This objective is conditional on the retained samples and does not calibrate teacher probabilities or confidence.

At $\lambda = 0$, valid actions receive binary agreement rewards. At $\lambda = K - 1$, their reward is $c = K\widehat{r}_{\mathrm{T}}$ when all K samples are retained; the fixed invalid penalty is not rescaled. With $K = 3, \lambda = 0.5$, $\alpha = 0.25$, and $[A, A, B]$, rewards are $1.25, 1, 0, -1$ for an A match, a B match, a valid non-match, and an invalid action. One retained teacher action gives unit reward for a match and objective $U_\theta - V_\theta$, as does an explicit $K = 1$ configuration. An empty multiset supplies no supervision, not negative labels.

A.3 Teacher supervision and semantic matching

Visible information and native tools. In standard AOPD, teacher and student receive the same retained public context and tool schemas through their native chat templates. Each teacher sample contains one immediate action. Hidden simulator state, verifier code, task solutions, and private user-simulator instructions are excluded. Self-AOPD adds the inputs in Appendix B.5. Tools are exposed directly, without `list_tools`/`call_tool` wrappers. Multiple student tool calls violate the single-action protocol; only the teacher's first call is retained when its output contains several.

Teacher sampling and validation. The teacher multiset is fixed before student generation. Each of three requested samples permits two retries for parser/schema-invalid output; valid samples are retained without regeneration. This filters protocol validity, not task success. A nonempty partial multiset can supervise a group, retaining nominal $K = 3$ in the bonus denominator. With two retained copies of A, a matching student response receives 1.25 at $\lambda = 0.5$. With no valid samples, collection ends without a student group or policy-failure label. Retries and partial retention differ from independent sampling in Section 2.2.

Caching. Teacher-cache keys include task/state context, tools, teacher/provider, prompt and protocol versions, decoding settings, and history-dependent state. Retained samples preserve multiplicity. Reuse does not provide fresh independent draws. Matcher-cache keys include the action pair, evidence, model, and prompt.

Context management. Native chat templates retain fixed task/system content and complete tool schemas, dropping the oldest removable exchanges until the prompt fits its token budget. The standard teacher uses the same retained public context; Self-AOPD has an additional teacher-only prompt allowance (Appendix B.5). If protected content cannot fit, the rollout terminates without truncating tool definitions or fabricating an action. Prompt construction leaves the environment's full interaction history intact.

Semantic matching. The implemented $m_s(y, y^{\mathrm{T}})$ compares retained tool-call fields or message text, not reasoning. It approximates $\mathbf{1}\{\phi_s(y) = \phi_s(y^{\mathrm{T}})\}$ and can judge equivalent realizations differently. Thus counts and rewards depend on sampled responses, not only their semantic classes. For fixed y, independent teacher draws make c/K unbiased for $\mathbb{E}_{y^{\mathrm{T}}}[m_s(y, y^{\mathrm{T}})]$. Accurate matching recovers $r_{\mathrm{T}}(s, \phi_s(y))$. With any fixed matcher, the score-function identity still applies to its expected reward, but errors can make that objective differ from J_{AOPD}.

The matcher has fixed parameters and prompts. It is instructed to assess agreement in the immediate operation or communicative act, including facts, identifiers, numbers, negation, commitments, and exact-copy requirements. Shared usefulness or outcomes do not establish equivalence. Ambiguous judgments are negative; API or response-format failures mean unavailable supervision, not negative matches.

Tool calls must name the same tool. Canonical checks ignore dictionary-key order and normalize integral floating-point values, but distinguish booleans from numbers. Canonicalization generally preserves distinctions between missing and null values, as well as list order and multiplicity; tool-source evidence can justify specific defaults or normalization. Remaining differences are judged using the tool schema, both argument objects, differing paths, public context, and tool-implementation evidence. Every teacher sample contributes a separate binary match, even when identical pairs reuse cached judgments. No transitive closure is applied.

Tool-implementation evidence is available to the *matcher*, not the teacher policy. It supports argument interpretation, not inspection of hidden database contents, task answers, or action quality. Two calls do not match merely because both fail or produce no change. Missing required source evidence makes supervision unavailable. Message matching receives both messages, public context, and tools, but no hidden solution. Normalized exact matches bypass the LLM. Cross-kind pairs do not match, but tool, message, and invalid candidates remain in the same group.

A.4 Optimization and execution details

Active groups and tokens. Eligible candidates are non-padding responses with available, reliable supervision. Eligibility does not require a valid action: invalid outputs can train with reward -1. Each group needs at least two eligible candidates and a reward range greater than 10^{-8}. Rewards are centered over these candidates and divided by the population standard deviation of centered rewards across active groups, with $\epsilon = 10^{-8}$. No additional mean subtraction is needed. A positive-reward action can have negative advantage, and an all-matching group remains informative if match counts differ. A vanishing standard deviation gives zero advantages. An empty active batch skips optimization. Padding contributes neither statistics nor gradients.

Advantages apply to all active student-generated tokens, including reasoning, but not to prompts, tool observations, or user-simulator outputs. The actor uses active-token mean aggregation. Compact execution removes candidates with no training tokens while retaining rollout records and metrics. Unselected eligible candidates still contribute to training.

Clipped optimization. Let $h(\rho, A) = \min\{\rho A, \mathrm{clip}(\rho, 1-\eta, 1+\eta)A\}$. The implementation additionally uses dual clipping (Ye et al., 2020),

$$h_{\mathrm{dual}}(\rho, A) = \begin{cases} \max\{h(\rho, A), c_{\mathrm{dual}} A\}, & A < 0, \\ h(\rho, A), & A \geq 0, \end{cases} \qquad \eta = 0.2, \quad c_{\mathrm{dual}} = 3. \tag{17}$$

The actor maximizes the mean of h_{dual} over active tokens. Teacher and matcher outputs, old-policy scores, and advantages remain fixed during the update. No reference-policy KL penalty or entropy bonus is added.

Sampling and asynchronous collection. Table 8 specifies the decoder distribution $\mu_{\theta_{\mathrm{old}}}$ and teacher generation settings. In Equation 8, old and updated π scores use the full softmax at the rollout temperature, without top-p/top-k truncation. Their ratio is not an exact importance correction from μ to π, the distribution assumed by the ideal derivation.

Workers overlap teacher sampling and tool/API waiting with generation at other ready states. Within each group, teacher samples are fixed before student generation. Collection uses one behavior-policy snapshot, followed by optimization. Algorithm 1 specifies these dependencies without requiring sequential workers.

B Experimental details

B.1 Training data and task selection

Table 6 records the transfer-training pools. AWM context selection retained tasks whose initial native prompt, including tool schemas, contained at most 16,000 tokens under the Qwen3-4B tokenizer. This selection cutoff is distinct from the training context limit. A deterministic audit checked scenario, database, and verifier integrity. Static feasibility screening then assessed each task using its tools, implementation, initial data, and verification evidence. It tested whether the environment supported the requested goal and success conditions, not whether a particular policy could succeed. EnvScaler used the same feasibility criterion after deterministic checks. Judge confidence was diagnostic, not an admission threshold.

Both static audits used DeepSeek-V4-Flash, with manifests linking retained tasks to source data and screening records. Admission did not require expert rollout success. Static screening does not rule out runtime defects.

Each 100-step transfer run contains 5,900 AWM and 500 EnvScaler task starts, potentially revisiting tasks from the 6,697-task pool. Tau comparisons, ablations, and Self-AOPD use the official 178-task training and 100-task test splits.

Table 6: Task-selection funnel for the pools used in this study. Counts are unique tasks remaining after each stage, not rollout attempts. Context selection is an AWM-specific stage. The dash denotes that no separate corresponding stage was used for EnvScaler.

Stage	**AWM**	**EnvScaler**
Source tasks	10,000	2,550
Context-eligible	9,380	—
Deterministic integrity pass	7,562	2,495
Static feasibility: healthy	5,942	755
Environments represented in the final pool	794	49

Table 7: Standard AOPD and ablation training configurations. Each configuration corresponds to one training run. The number of samples refers to responses at one state, not independent complete trajectories.

Setting	Transfer experiment	Tau ablations
Student	Qwen3-4B or Qwen3-8B	Qwen3-4B
Action teacher	Qwen3.7-Flash or DeepSeek-V4-Flash	Qwen3-32B
Training tasks	AWM/EnvScaler static-feasibility pools	Official train splits
Collection steps reported	100	50
Tasks per step	64: 59 AWM + 5 EnvScaler	24: 8 per Tau domain
Student samples N	4	4
Teacher samples K	3	3
Frequency bonus scale λ	0.5	0.5 for AOPD/random commit; 0 for binary reward
Maximum agent decisions	20 AWM; 40 EnvScaler	20
Optimizer	AdamW	AdamW
Learning rate / schedule	10^{-6} / constant, no warmup	10^{-6} / constant, no warmup
PPO epochs / minibatch size	1 / 32	1 / 32
PPO clip / dual clip	0.2 / 3	0.2 / 3
Advantage normalization	Group centering, batch whitening	Group centering, batch whitening
Loss aggregation	Active-token mean	Active-token mean
KL penalty / entropy bonus	Neither	Neither
Outcome reward in loss	No	No

B.2 Training configurations and auxiliary models

Tables 7 and 8 summarize training and generation settings. Teacher/matcher inputs follow Appendix A.3. User simulators generate interaction turns, not distillation targets. Our implementation extends `verl-agent` (Feng et al., 2025), built on veRL/HybridFlow (Sheng et al., 2025).

One collection step can involve multiple optimizer updates: one PPO epoch traverses the collected candidates in minibatches rather than updating once per task batch. Tau's 20-decision training limit is distinct from the native environment's 200-internal-transition budget. KD, OPD, and Self-AOPD share AOPD's optimizer, learning-rate, update, and sampling settings, but retain their respective objectives.

Main-experiment matcher and EnvScaler user-simulator settings are shared across all four student–teacher configurations. Tau training instead uses Qwen3.8-27B (Qwen Team, 2026b) for matching and user simulation; Tau evaluation uses Qwen3.5-9B (Qwen Team, 2026a). Self-AOPD's private user summaries were prepared offline with DeepSeek-V4-Flash and reviewed, so auxiliary models remain part of its training setup.

B.3 Evaluation protocol and score definitions

BFCL and ToolSandbox. Students and directly evaluated teachers use the same task subsets and interaction limits. BFCL includes 800 multi-turn tasks, 200 per category, with one trial and the native per-turn step limit of 20. ToolSandbox uses 490 strictly local scenarios, four scheduled trials each, and at most 30 messages. Similarity is averaged over all 1,960 scheduled trials.

AppWorld. AppWorld evaluation covers 168 Normal and 417 Challenge tasks for students and directly evaluated teachers, with one trial, 50 steps, and a 100,000-output-token budget per task. TGC is the fraction passing all task-goal tests; SGC measures completion of all variants within a scenario. Normal and Challenge scores are reported separately. The average uses TGC only. Our harness runs the benchmark tests. Shared task and interaction budgets do not imply matched generation budgets (Table 8).

Table 8: Model roles and generation settings.

Generation Settings				
Experiment	**Role and Model**	**Thinking**	**Sampling**	**Maximum Response**
AWM/ EnvScaler Training	Student: Qwen3-4B/8B	Enabled	$T = 0.6$; $p = 0.95$; $k = 20$	4k
	Teacher: DeepSeek-V4-Flash	Enabled; effort max	—	4k
	Teacher: Qwen3.7-Flash	Enabled	$T = 0.6$; $p = 0.95$	8k
	Matcher: DeepSeek-V4-Flash	Disabled	$T = 0$	128
	User: DeepSeek-V4-Flash	Disabled	$T = 1$	—
Tau/Self-AOPD Training	Student: Qwen3-4B	Enabled	$T = 0.6$; $p = 0.95$; $k = 20$	4k
	Teacher: Qwen3-32B	Enabled	$T = 0.6$; $p = 0.95$; $k = 20$	8k
	Self Teacher: Qwen3-4B	Enabled	$T = 0.6$; $p = 0.95$; $k = 20$	4k
	Matcher: Qwen3.8-27B	Disabled	$T = 0.7$; $p = 0.8$; $k = 20$	32k
	User: Qwen3.8-27B	Disabled	$T = 0.7$; $p = 0.8$; $k = 20$	8k
Evaluation	Student: Qwen3-4B/8B	Enabled	Greedy ($T = 0$)	4k
	Teacher: Qwen3.7-Flash	Enabled	$T = 0.6$; $p = 0.95$	32k
	Teacher: DeepSeek-V4-Flash	Enabled; effort max	—	32k
	Tau User: Qwen3.5-9B	Enabled	$T = 1$	8k

Tau transfer evaluation. Transfer evaluation covers all 50 Airline, 114 Retail, and 114 Telecom tasks in the official base splits, including the subsets used for Tau train/test comparisons. Transfer students were trained on AWM/EnvScaler, not these tasks. Evaluation uses native function calling and the generation settings in Table 8. Each model has four trials per task.

All Tau evaluations in this paper disable natural-language assertions, including directly evaluated teachers, method comparisons, ablations, Self-AOPD, and the Privileged Teacher reference. Tau's evaluator supplies success labels under this scoring configuration, not the semantic matcher. Success therefore does not certify every utterance.

Tau method comparison, ablation, and Self-AOPD evaluation. Each model has four trials per task. Repeated Base and AOPD entries reuse the same results, not independent runs. Airline, Retail, and Telecom train/test splits contain 30/20, 74/40, and 74/40 tasks. Training-set evaluation measures in-domain fit, not held-out generalization. Its episodes are distinct from training rollouts. Ordinary student evaluation excludes reference actions and private task criteria. The Privileged Teacher instead receives additional information and is not a same-input reference.

Success and reliability. Following Tau's repeated-success metric (Yao et al., 2025), for task j with n scheduled trials and c_j successes, we compute

$$\widehat{p}_j^{\,1} = \frac{c_j}{n}, \qquad \widehat{p}_j^{\,4} = \frac{\binom{c_j}{4}}{\binom{n}{4}}, \tag{18}$$

where $\binom{c_j}{4} = 0$ when $c_j < 4$. Each domain score averages these values over its tasks.

For the main table, let B be BFCL accuracy, S ToolSandbox similarity, G_{N} and G_{C} AppWorld Normal and Challenge TGC, and P_d Tau pass^1 in domain d. On a common percentage scale,

$$\begin{aligned} G &= \tfrac{1}{2}(G_{\mathrm{N}} + G_{\mathrm{C}}), \\ P &= \tfrac{1}{3}(P_{\mathrm{Airline}} + P_{\mathrm{Retail}} + P_{\mathrm{Telecom}}), \\ \mathrm{Avg} &= \tfrac{1}{4}(B + S + G + P). \end{aligned} \tag{19}$$

This study-specific summary weights benchmark families equally, without counting AppWorld SGC or Tau reliability again. Scores use one-decimal half-up rounding. Average gains are computed before display rounding, so subtracting displayed values can differ by 0.1 percentage points. Tau-only tables instead average the three domains separately for each metric and split; train and test are never pooled. Each configuration was trained once; repeated evaluation trials measure interaction variability, not variation across independent training runs.

B.4 Tau baselines, ablations, and collection budgets

All configurations use the same 178-task Tau training set and Qwen3-4B initialization. KD, OPD, and standard AOPD use Qwen3-32B as action teacher. Each step starts eight tasks per domain, and Tables 2 and 3 report step 50. Their AOPD rows reuse one run.

AOPD. The reference configuration uses Equation 6 with $K = 3$, $\lambda = 0.5$, and uniform tie-breaking among maximum-reward candidates.

Validity Reward. Training assigns $+1$ to valid actions and -1 to invalid output or transfer-protocol violations. Teacher sampling and semantic matching remain active for AOPD's commitment rule, not training rewards. All-valid groups are masked as equal-reward groups, so this variant changes the number and composition of trainable groups as well as removing semantic reward distinctions.

Binary Reward. With the same three teacher samples, $\lambda = 0$ gives $+1$ for any match, zero for valid non-matches, and -1 for invalid actions. This reward also governs commitment, changing both learning signals and ties relative to AOPD.

Random Commit. Rewards follow AOPD, but commitment is uniform over all four candidates, including invalid and zero-reward actions. There is no success-based resampling. Unselected candidates remain eligible for training.

A transfer notification without the preceding transfer call is treated as invalid output with reward -1. Equal-reward groups produce no learning signal.

KD: successful teacher-trajectory imitation. For each task start, Qwen3-32B independently generated four complete trajectories; only successful ones entered SFT. Following output imitation (Kim & Rush, 2016), loss covers all teacher-generated reasoning and action/message tokens, while user turns and tool observations provide context only. Thus outcome labels filter demonstrations, but do not supply a policy-gradient reward.

OPD: sampled-token reverse-KL. OPD collected four student trajectories per task start. The teacher scored student-sampled tokens at their corresponding prefixes for sampled-token reverse-KL supervision (Fu et al., 2026). It requires teacher log-probabilities and a compatible token space, but not full-vocabulary distributions.

Collection budgets. KD and OPD collect four trajectories per task start; AOPD samples four candidates per state and executes one, while all eligible candidates contribute to training. Trajectory lengths, teacher generation/scoring, matching costs, and KD filtering differ. Equal task starts and collection steps therefore do not equalize retained examples, tokens, teacher queries, or compute.

B.5 SELF-AOPD: TEACHER INPUTS AND PARAMETER UPDATES

Current-weight teacher, fixed update targets. Self-AOPD used Qwen3-4B in both roles, with eight task starts per Tau domain per step. We report results at step 50. Four student candidates and three teacher samples were independently generated from the same current parameters. Teacher samples and matching results remained fixed during optimization. Subsequent collection used updated shared parameters. Self-teacher caches were cleared at each collection step, without importing external-teacher samples.

Table 8 specifies generation settings. The Self-AOPD teacher has an additional 4,096-token prompt allowance within the shared context window, but this allowance itself does not imply privileged content. Teacher outputs and private prompt tokens were excluded from the student loss. Only student-generated candidate tokens received the action-level signal.

Teacher guidance and privileged information. The student received the domain policy, available tools, and observed interaction history. Additional teacher guidance covered immediate actions, preconditions, authorization, clarification, and avoiding repetition. Telecom device operations were rendered as user-facing instructions rather than user-tool syntax. This conversion added no private facts.

Privileged task information comprised reviewed summaries of private goals, constraints, preferences, and known/unknown information, including facts not yet disclosed in dialogue. It also supplied reference tool names and arguments, customer-side operations, target states, and required messages from evaluation criteria. These inputs did not replace required public confirmation or execution evidence. They were withheld from the student and matcher, although they could affect which student actions were rewarded. The trained student was evaluated without privileged inputs.