TACIT-SWITCH: COST-AWARE MODEL ESCALATION FOR LLM AGENTS FROM CENSORED SUPERVISION

Anonymous authors

ABSTRACT

Agents with smaller language-model backbones are less expensive but can drift into persistent failure modes, whereas those with larger backbones are generally more reliable but more costly. This reliability-cost trade-off motivates routing methods that decide when to invoke an agent with a larger backbone: before execution, after a fixed trajectory prefix, or locally at individual steps. Our method, TACIT-SWITCH, learns permanent handoff policies from accumulated trajectory evidence and *Teacher-Annotated Censored Intervention Times* (TACIT). It represents each annotation as an interval-censored observation on a cumulative-risk scale. The resulting mixture-cure threshold model estimates the probability that the paired Strong rollout succeeds and, conditional on success, the handoff threshold; no teacher is required at deployment. In a mechanism-based multi-step simulation, TACIT-SWITCH improves success by 7.4–11.1 percentage points over task-level, step-level, and fixed-prefix routing baselines at comparable cost. Within that controlled simulation, ablations show that task features and cumulative trajectory risk provide complementary information. With operating points selected on development data, TACIT-SWITCH achieves the highest held-out success among learned policies on both ALFWorld (48.5% with 4B Cheap; 45.5% with 9B Cheap) and DABench (73.1%).

1 INTRODUCTION

Efficient agent deployment requires deciding when to use language-model backbones with different capabilities and inference costs. For a one-shot request, routing is largely a static model-selection problem: choose a model before generation begins. In interactive agents, however, routing becomes a sequential decision problem. Each action alters the state from which later decisions must proceed, while evidence of looping, protocol violations, or stalled progress often emerges only as the trajectory unfolds. Handing off immediately to the larger model sacrifices the cost advantage of the smaller model; handing off too late may leave little opportunity for recovery. Routing therefore becomes a recoverability-aware stopping-time problem: deciding whether and when to hand control to the larger model.

Existing methods make routing decisions at different points in execution (Figure 1). Request-level methods do so without observing an evolving environment trajectory: RouteLLM chooses a model before generation, whereas FrugalGPT uses a learned cascade (Ong et al., 2025; Chen et al., 2024). ReDAct and TRIM make step-local, non-permanent interventions after which control returns to the smaller model (Piatrashyn et al., 2026; Kapoor et al., 2026); SWE-Router makes one continue-or-restart decision after a fixed prefix (Son et al., 2026). EvoRoute instead repeatedly selects models across explicit workflow subtasks (Zhang et al., 2026). We study a different control structure: an adaptively timed, one-way transfer for the remainder of the episode, which we call a *permanent handoff*.

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.

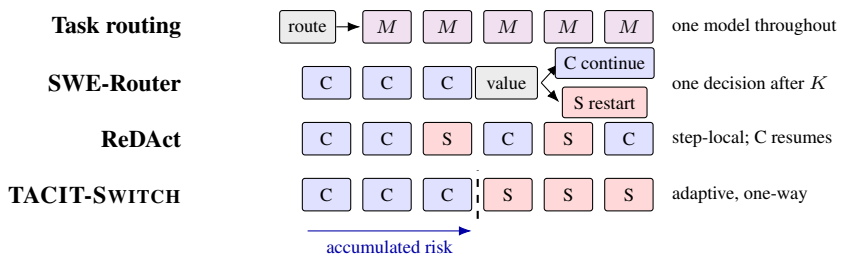

Figure 1: Four model-routing strategies distinguished by when they make routing decisions: task routing (one pre-execution model choice), SWE-Router (one post-prefix decision to continue with C or restart with S), ReDAct (step-local, non-permanent deferral), and TACIT-SWITCH (an adaptive one-way handoff from the current trajectory). Here C and S denote Cheap and Strong, respectively.

TRIM assumes delimited reasoning steps and step-level correctness scores, whereas EvoRoute assumes a role-structured multi-model workflow and an experience base. Neither method maps directly to our fixed two-model setting, so we compare task-, step-, and fixed-prefix routing policies under the same protocol.

Learning when to hand off is difficult because the desired supervision is counterfactual. A "do not hand off" label is ambiguous: the trajectory may not yet require intervention, or the stronger model may be unable to rescue the task at all. We instead collect one full-episode rollout from each model and ask an offline teacher to mark a coarse handoff window. The teacher window provides timing supervision, analogous to expert-intervention labels (Spencer et al., 2020). Records whose paired Strong rollout from the initial state succeeds form the paired-Strong-success stratum ($B_i = 1$). When Cheap fails and Strong succeeds, the teacher marks either a post-action checkpoint or a short range of checkpoints at which handoff appears useful and recovery remains feasible. A single marked checkpoint is treated as a one-step interval, so both forms locate the latent threshold on the cumulative-risk scale.

When both models succeed, $B_i = 1$ is observed but there is no rescue time because Cheap completes the task without handoff. When the paired Strong rollout fails, the data provide no observed evidence that switching would rescue the task. We call this supervision *Teacher-Annotated Censored Intervention Times* (TACIT).

We separate two uncertainties: whether the paired Strong rollout succeeds and, conditional on that outcome, where the handoff threshold lies on the cumulative-risk scale. This incidence–threshold decomposition adapts the logic of mixture-cure models (Farewell, 1982; Kuk & Chen, 1992). Coarse teacher intervals are represented as interval-censored observations (Turnbull, 1976), and a log-normal AFT-style parameterization models the positive, task-dependent threshold κ_i (Wei, 1992). The corresponding handoff step is not modeled directly; it is induced by the first crossing, $\tau_i^\star = \inf\{t : R_i(t) \geq \kappa_i\}$. Under this working factorization,

$$q_i(t) = \pi_\gamma(x_i) F_{\beta,s}(R_i(t) \mid x_i) \tag{1}$$

is the product of the estimated probability that the paired Strong rollout succeeds and the conditional probability that its threshold has been crossed. On the development set, we choose a threshold α to balance success and cost. At deployment, the policy permanently hands control to Strong when $q_i(t) \geq \alpha$. The risk scale is learned, and the teacher's coarse interval may depend on the observed trajectory. We therefore use the survival model as a practical model for switching decisions, not as a literal description of how agent behavior is generated.

Practical implications. Training avoids exhaustive checkpoint-level counterfactual rollouts by using one full-episode rollout from each model plus a coarse offline annotation. The deployed router uses trajectory diagnostics and a fitted statistical score, with no teacher or separate LLM/value head. Since $q_i(t) \leq \pi_\gamma(x_i)$, the incidence term caps the score and can suppress costly handoffs when paired-Strong success is unlikely. A teacher range that brackets the latent threshold retains timing information without an exact checkpoint; Theorem 1 quantifies local stability to limited interval corruption.

We evaluate the method in three settings. A correctly specified model-based simulation examines parameter and score recovery. A separate mechanism-based simulation examines decision utility when trajectories are generated by dynamics outside the fitted model; ablations within this simulation measure the contributions of task features and online risk. Interactive experiments on ALFWorld and DABench compare held-out success and cost with task-, step-, and fixed-prefix routing.

2 Problem Formulation and TACIT Supervision

2.1 Permanent handoff on a cumulative-risk scale

We consider episodes of at most H interaction steps. A permanent-handoff policy h begins each episode with a Cheap policy π_{c}. Based on the trajectory observed so far, h may transfer control once to a Strong policy π_{s}; after transfer, π_{s} produces all remaining actions. Let $Y_i(h) \in \{0, 1\}$ and $\mathrm{Cost}_i(h)$ denote task success and deployment inference cost.

We seek policies that maximize success under an inference budget,

$$\max_h \ \mathbb{E}[Y_i(h)] \qquad \text{s.t.} \qquad \mathbb{E}[\mathrm{Cost}_i(h)] \le C_{\max}. \tag{2}$$

This constraint defines the success–cost trade-off. Here $Y_i(h)$ records task-specific performance, while $\mathrm{Cost}_i(h)$ is defined by the deployment setting and may vary across tasks. TACIT-Switch does not require a particular cost model. It first estimates the handoff score $\widehat{q}_i(t)$; the application-specific cost enters only when choosing its decision threshold α on development data. This threshold is then fixed for evaluation and deployment.

To determine when to hand off from the observed trajectory, we represent the handoff threshold on a learned cumulative-risk scale rather than at a fixed time step. Specifically, at each decision checkpoint t, we map the observed trajectory prefix $\mathcal{H}_{i,t}$ to a nonnegative diagnostic vector $u_i(t) = \phi(\mathcal{H}_{i,t}) \in \mathbb{R}^d_+$. Its entries capture signals such as repeated or invalid actions, uncertainty, and stalled progress. We learn nonnegative diagnostic weights $w = (w_1, \ldots, w_d)$ satisfying $\sum_{j=1}^d w_j = 1$. The stepwise and cumulative risk scores are

$$r_i(t) = w^\top u_i(t), \qquad R_i(t) = \sum_{s \le t} r_i(s). \tag{3}$$

Because both $u_i(t)$ and w are nonnegative, $R_i(t)$ is nondecreasing as evidence accumulates. For an episode whose paired Strong rollout succeeds, let $\kappa_i > 0$ denote its latent handoff threshold on the cumulative-risk scale. The corresponding first-crossing step is

$$\tau_i^\star = \inf\{t : R_i(t) \ge \kappa_i\}, \qquad \inf \varnothing = \infty. \tag{4}$$

This crossing is a decision-model construct; it does not change either model's ability or the environment dynamics.

2.2 Censored TACIT supervision

Each training task provides paired final outcomes from complete Cheap and Strong rollouts. When Cheap fails but Strong succeeds, an offline teacher additionally marks a coarse interval in which handoff appears useful and still recoverable. Let A_i and B_i indicate Cheap and Strong success, respectively, from the initial state, and abbreviate the outcome pair (A_i, B_i) as AB. Thus B_i records the realized outcome of one paired Strong rollout from the initial state; it is not an intrinsic or state-independent capability label. In the working likelihood, records with $B_i = 0$ enter the cure component. Because the evaluated Strong rollout failed, these records provide no observed evidence that switching would rescue the task. Records with $B_i = 1$ form the paired-Strong-success stratum and use a finite handoff-threshold distribution. Table 1 summarizes the resulting evidence.

Table 1: Mapping from paired full-episode rollout outcomes to handoff evidence. Only $AB = 01$ receives a teacher interval.

Type	Paired outcomes	Likelihood evidence
01	cheap fails, strong succeeds	paired-Strong success; interval (or time missing)
11	both succeed	paired-Strong success; right censoring
10	cheap succeeds, strong fails	operational cure component
00	both fail	operational cure component

For $AB = 01$, an offline teacher reads the complete Cheap trajectory and marks either a post-action checkpoint $\widetilde{\tau}_i$ or a wider range of checkpoints at which handoff first appears useful and recovery remains feasible. A single marked checkpoint does not identify an exact real-valued threshold; it gives the one-step interval

$$R_i(\widetilde{\tau}_i - 1) < \kappa_i \leq R_i(\widetilde{\tau}_i). \tag{5}$$

More generally, a teacher range $[a_i, b_i]$ gives $R_i(a_i - 1) < \kappa_i \leq R_i(b_i)$, with the convention $R_i(-1) = 0$. These interval-censored observations constitute TACIT supervision. The teacher coarsens only the threshold location; membership in the paired-Strong-success stratum already follows from $B_i = 1$. $AB = 11$ belongs to this stratum but is right-censored because Cheap completes the episode without an observed need for rescue.

3 TACIT-Switch: Model, Estimation, and Deployment

3.1 Incidence–threshold model

The incidence–threshold model combines three established survival-analysis tools: mixture-cure models separate susceptibility from the conditional event distribution (Farewell, 1982; Kuk & Chen, 1992), AFT models regress log event times on covariates (Wei, 1992), and interval censoring represents event locations known only through a window (Turnbull, 1976). In our application, B_i is the observed incidence outcome, and the event location is a handoff threshold on learned cumulative risk rather than elapsed time; the first crossing of this threshold triggers a permanent handoff.

Accordingly, a logistic component models the observed Strong-success indicator B_i, while a log-normal AFT component models the threshold location conditional on $B_i = 1$:

$$\pi_i = \mathbb{P}(B_i = 1 \mid x_i) = \mathrm{sigmoid}(x_i^\top \gamma), \qquad \log \kappa_i \mid (B_i = 1) = x_i^\top \beta + s\epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0,1). \tag{6}$$

Here x_i contains compact task features and $F_{\beta,s}(\cdot \mid x_i)$ denotes the conditional CDF of a finite threshold.

3.2 Censored likelihood and estimation

Write $R_{i,w}$ for the cumulative score in Equation (3) to emphasize its dependence on w. For a teacher window $[a_i, b_i]$, define $L_i(w) = R_{i,w}(a_i - 1)$ and $U_i(w) = R_{i,w}(b_i)$, with $R_{i,w}(-1) = 0$. For a right-censored episode ending at H_i, define $C_i(w) = R_{i,w}(H_i)$. Because B_i is observed for every paired rollout, the main working-likelihood contributions are

$$\mathcal{L}_i(\theta) = \begin{cases} 1 - \pi_i, & \text{cured } (B_i = 0), \\ \pi_i[F_{\beta,s}(U_i(w) \mid x_i) - F_{\beta,s}(L_i(w) \mid x_i)], & \text{teacher interval } (B_i = 1), \\ \pi_i[1 - F_{\beta,s}(C_i(w) \mid x_i)], & \text{right-censored } (B_i = 1). \end{cases} \tag{7}$$

If $B_i = 1$ but no teacher time is available, marginalizing over the unobserved threshold location leaves the incidence contribution π_i. We jointly estimate $\theta = (w, \gamma, \beta, s)$ by maximizing the working likelihood with ℓ_2 regularization on the non-intercept coefficients of γ and β. Appendix D gives the objective and inference pseudocode. When the working model and its coarsening assumptions hold, Equation (7) is the observed-data likelihood. For agent data, however, the teacher constructs the interval after inspecting the realized trajectory, so the coarsening mechanism need not be ignorable (Heitjan & Rubin, 1991). We therefore use Equation (7) as a working likelihood for learning the switching policy, without claiming a generative model for the trajectory or teacher annotation.

For finite-sample analysis, we use unconstrained local coordinates for the simplex weight w and positive scale s, and let p be the number of free parameters. Let $\theta^\dagger$ denote a local minimizer of the clean population version of the regularized working objective. Assumption 1 defines the neighborhood radius r, local curvature μ, score-concentration scale σ, and interval-corruption influence bound G_T.

Theorem 1 (Local stability under teacher-interval noise)**.** *Fix $\delta \in (0,1)$. Under Assumption 1, suppose that teacher-provided intervals are incorrect for at most a fraction ρ_T of the n records. Define*

$$a_n(\delta) = 2\sigma\sqrt{\frac{2\{p\log 5 + \log(4/\delta)\}}{n}} + G_T\rho_T. \tag{8}$$

If $a_n(\delta) < \mu r/2$, then, with probability at least $1-\delta$, the corrupted empirical objective has a unique stationary point $\widehat{\theta} \in \mathbb{B}_r(\theta^\dagger)$, and

$$\|\widehat{\theta} - \theta^\dagger\|_2 \le \frac{2a_n(\delta)}{\mu}. \tag{9}$$

For a task with features x and observed diagnostics $u(1{:}t)$, let $q_\theta(x, u(1{:}t)) = \pi_\gamma(x)F_{\beta,s}(R_w(t) \mid x)$ denote the model's handoff score at step t. If this score is uniformly L_q-Lipschitz in θ over all evaluation tasks, decision steps, and $\mathbb{B}_r(\theta^\dagger)$, then on the same event, for every such task and step,

$$|q_{\widehat{\theta}}(x, u(1{:}t)) - q_{\theta^\dagger}(x, u(1{:}t))| \le \frac{2L_q a_n(\delta)}{\mu}. \tag{10}$$

Thus, relative to the clean population working target, parameter and handoff-score errors are $O\left(\sqrt{p/n} + \rho_T\right)$, where the two terms capture finite-sample variation and teacher-interval corruption. Appendix A states the assumptions and gives the full proof.

3.3 Cost-sensitive online handoff

At deployment, the teacher is absent. At each benchmark-defined decision checkpoint, the policy updates $R_i(t)$ from the Cheap trajectory evidence available at that checkpoint and computes the fitted handoff score

$$\widehat{q}_i(t) = \hat{\pi}_i F_{\hat{\beta},\hat{s}}(R_i(t) \mid x_i). \tag{11}$$

It permanently hands control to π_{s} when

$$\widehat{q}_i(t) \ge \alpha. \tag{12}$$

Because the implemented diagnostics are nonnegative, the cumulative score is nondecreasing. The deployed switch time is therefore the first decision checkpoint satisfying Equation (12), with no switch if the set is empty. Lowering α cannot delay this crossing.

Lower values of α switch earlier and more often, exposing a family of success–cost operating points. We select α on development data by maximizing empirical success subject to the prescribed cost budget and freeze it before test evaluation.

4 Controlled Statistical Experiments

We use two complementary simulations. The first generates censored observations directly from the proposed AB–AFT model and tests whether its parameters and predictions can be recovered. The second generates complete multi-step trajectories from an agent-like path-state process and tests whether the fitted switching policy improves success at a fixed cost. This second simulation is not assumed to follow the AB–AFT model and therefore evaluates decision utility rather than parameter recovery. Information ablations and one-factor sensitivity experiments then isolate the roles of task features and online risk. Figures 2 and 3 summarize the four statistical studies. Throughout this section, "$\pm$" denotes the half-width of a Student-t 95% Monte Carlo confidence interval across independent replicates.

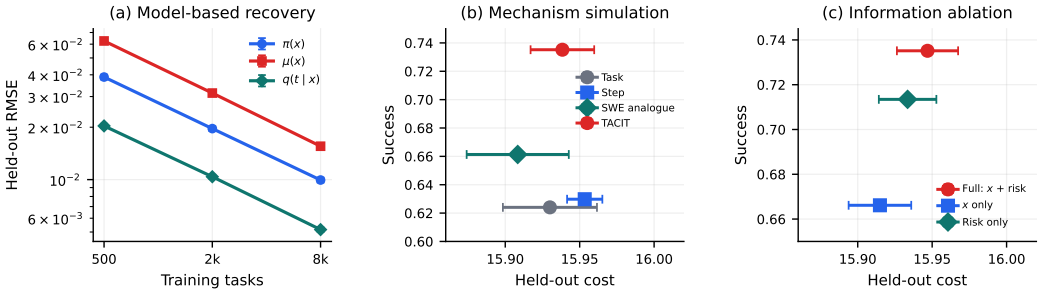

Figure 2: Controlled recovery, policy, and ablation results. (a) Held-out recovery error over 500 independent fits per sample size. (b) The 100-repeat mechanism-based policy comparison, including the development-tuned SWE-Router analogue. (c) The 100-repeat information ablation. Points show replicate means; bars show Student-t 95% Monte Carlo confidence intervals for those means. Candidate policies are fitted on training data, selected on development data, and evaluated once on test after all choices are frozen.

4.1 Model recovery under correct specification

We first isolate the statistical behavior of the estimator under correct model specification. Data are generated according to

$$B_i \mid x_i \sim \text{Bernoulli}(\pi_i), \qquad \log \kappa_i = \widetilde{x}_i^\top \beta^\star + s^\star \varepsilon_i, \quad \varepsilon_i \sim N(0,1).$$

Writing $\mu(x) = \widetilde{x}^\top \beta^\star$, the induced first-crossing time and handoff probability are

$$T_i = \inf\{t : R_i(t) \geq \kappa_i\}, \qquad q(t \mid x_i) = \pi(x_i) \Pr\{\kappa_i \leq R_i(t) \mid x_i, B_i = 1\}.$$

For $B_i = 1$, a crossing by the horizon produces an AB01 interval, whereas $T_i > H$ yields an AB11 right-censored observation. When $B_i = 0$, a separate Bernoulli nuisance model for A_i yields AB00 or AB10. We fit 500 independent samples at each $n \in \{500, 2{,}000, 8{,}000\}$ and evaluate every fit on a fresh sample of 10,000 tasks.

All 1,500 fits converge. Componentwise parameter RMSE approximately halves whenever the sample size quadruples, while held-out $q(t \mid x)$ RMSE decreases from 0.0203 ± 0.0006 at $n = 500$ to 0.0052 ± 0.0001 at $n = 8{,}000$. These results are consistent with an $n^{-1/2}$-like recovery rate. The mean AB00/01/10/11 proportions remain stable at $0.259/0.261/0.197/0.284$.

Appendix C studies two types of corruption at rates up to 20%: moving a one-step teacher interval to an adjacent step and flipping the paired Strong outcome. The estimator changes little under the former but is more sensitive to the latter.

4.2 Mechanism-based multi-step simulation

To test policy utility outside the fitted likelihood, we use a separate ten-step, three-state simulator in which a task succeeds after at least eight correct actions. Cheap and Strong follow fixed state-transition and action-success probabilities, with Strong more likely to recover from degraded states. A noisy post-step signal provides partial evidence about path quality, while x provides a weak task-level prior. The latent task threshold affects only synthetic supervision and diagnostic quantities; it does not enter actor dynamics, risk emissions, or terminal success.

Across 100 independent replicates, policies are fitted on 8,000 training tasks, tuned on 4,000 development tasks under a cost cap of 16, and evaluated on 20,000 test tasks after all choices are frozen. Cheap and Strong steps cost 1 and 3. We compare Task Router, Step Deferral using the latest risk, a fixed-prefix SWE-Router analogue (Son et al., 2026), and TACIT-SWITCH; implementation and teacher details are deferred to Appendix B.

Pure Cheap and Pure Strong succeed on 44.59% and 84.88% of test tasks. At nearly identical realized costs (15.91–15.95), Task Router, Step Deferral, the SWE-Router analogue, and TACIT-SWITCH attain 62.40%, 62.98%, 66.13%, and 73.52% success. The paired gains of TACIT-SWITCH are 11.12, 10.54, and 7.39 percentage points, all positive in every replicate. The gain comes from using trajectory evidence to identify when Strong is more useful.

4.3 Information ablation

The path-state construction makes task information and trajectory information individually imperfect, allowing us to measure their separate contributions. On the same 100-replicate splits, we compare the Full model with x-only and risk-only variants. The x-only variant masks the diagnostic values but retains indicators of which Cheap steps were executed.

At nearly identical realized cost, Full reaches $73.51 \pm 0.09\%$ success, compared with $66.62 \pm 0.10\%$ for x-only and $71.35 \pm 0.09\%$ for risk-only. Its paired gains are 6.90 ± 0.08 and 2.17 ± 0.08 percentage points, respectively. Online risk provides the larger marginal gain, while the improvement over risk-only shows that task features add information beyond online risk.

4.4 Scenario sensitivity

Finally, we vary task-feature and online-risk information one at a time, holding actor behavior and endpoint difficulty fixed. Each axis contains five settings spanning the observed AUC range under the same development-only selection protocol; full results appear in Appendix B.

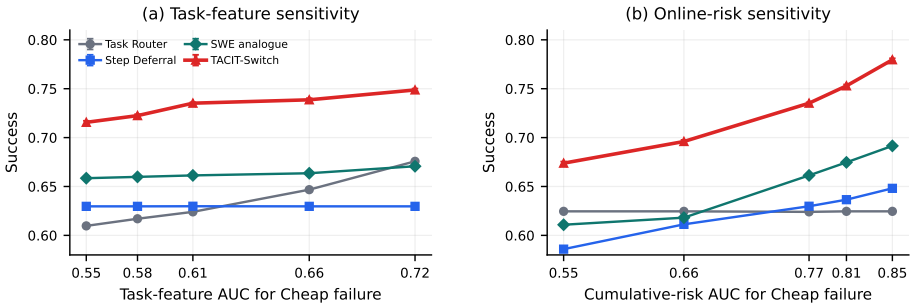

Figure 3: One-factor information sensitivity. (a) Task-feature information is varied while the execution process is fixed; the horizontal axis is the AUC with which task features predict terminal Cheap failure. (b) Online-risk information is varied by changing only risk-emission noise; the horizontal axis is the corresponding AUC for cumulative risk. Both panels compare Task Router, Step Deferral, the SWE-Router analogue, and TACIT-Switch. The baseline uses 100 repeats and each nonbaseline condition uses 50. Points show replicate means; bars show Student-t 95% Monte Carlo confidence intervals for those means.

Figure 3 shows that TACIT-Switch has higher success than all baselines at comparable cost. Success increases monotonically across the five task-feature settings, from 71.56% to 74.87%, and across the five online-risk settings, from 67.38% to 77.96%. Pure-Cheap and Pure-Strong success remain stable, so these trends reflect information quality rather than changes in task difficulty or actor capability.

5 Interactive-Agent Experiments

5.1 Common protocol

Within each benchmark, Cheap and Strong share the scaffold, prompt, action space, and step budget. Paired rollouts define A = Cheap success and B = Strong success from the initial state; B is a realized paired outcome, not an intrinsic capability label. Records with $AB \in \{00, 10\}$ enter the operational cure component. An $AB = 01$ record receives a teacher interval when one is available, while $AB = 11$ is right-censored because Cheap succeeds without a handoff. DeepSeek V4 Pro labels only $AB = 01$ training trajectories; no teacher is used on development or test.

We compare Pure Cheap and Pure Strong, Task Router, a ReDAct-inspired step-local deferral baseline (Piatrashyn et al., 2026), a SWE-Router-style fixed-prefix baseline (Son et al., 2026), and TACIT-Switch. All policy settings are chosen once on development data under a fixed cost cap and

frozen before test. The cost proxy selects the development operating point but does not enter score fitting. Here, costs are expressed in 4B-equivalent units based on model parameter counts. Table 2 and Figure 4 report held-out results; exact likelihood and cost accounting appear in Appendix B.

5.2 ALFWORLD

ALFWorld is a text-based interactive household-task benchmark aligned with embodied environments (Shridhar et al., 2021). We pair Qwen3.5-4B or Qwen3.5-9B with Qwen3.6-27B, allow at most 50 actions, fit on 100 valid-seen episodes, select on 40 held-out seen episodes, and evaluate once on 134 valid-unseen episodes. All operating settings are chosen on the development split; Appendix B gives the outcome counts and candidate grids.

Table 2 and Figure 4 show that TACIT-SWITCH has the highest held-out success among learned policies for both model pairs. With 9B Cheap, it achieves higher success at lower cost than both Task Router and SWE-Router; with 4B Cheap, it approaches pure-27B success at lower cost. Bootstrap intervals based on paired tasks exclude zero for the gain over SWE-Router in both model pairs.

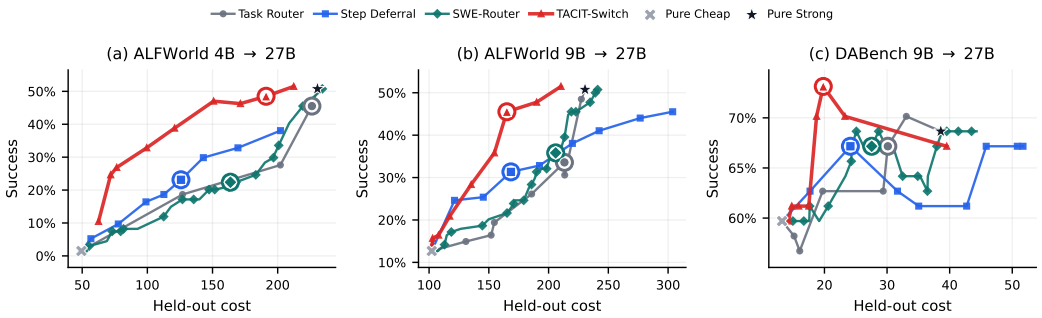

Figure 4: **Held-out success–cost curves for the real-agent experiments.** Lines trace test performance over the prespecified policy grids; rings mark the operating points selected on development data. The remaining test points show curve shape only and were not used for selection.

Table 2: **Frozen interactive-agent operating points.** All models are fitted on training data. Operating points and policy hyperparameters, including the SWE-Router prefix length and threshold, are selected on development data and frozen before test. Strong share is the fraction of deployed model calls made by Strong. † marks the proposed method.

Benchmark	Policy	Success	Avg. cost	Strong share
DABench Test67	Pure 9B	40/67 (59.7%)	13.23	0.0%
	Task router	45/67 (67.2%)	30.12	68.3%
	Step deferral	45/67 (67.2%)	24.15	22.9%
	SWE-Router ($K = 2$)	45/67 (67.2%)	27.55	47.3%
	TACIT-SWITCH†	49/67 (73.1%)	19.81	33.3%
	Pure 27B	46/67 (68.7%)	38.59	100.0%
ALFWorld unseen134 (4B)	Pure 4B	2/134 (1.5%)	49.40	0.0%
	Task router	61/134 (45.5%)	226.26	92.3%
	Step deferral	31/134 (23.1%)	125.79	22.4%
	SWE-Router ($K = 2$)	30/134 (22.4%)	163.72	46.9%
	TACIT-SWITCH†	65/134 (48.5%)	191.10	79.5%
	Pure 27B	68/134 (50.7%)	230.51	100.0%
ALFWorld unseen134 (9B)	Pure 9B	17/134 (12.7%)	102.22	0.0%
	Task router	45/134 (33.6%)	213.26	72.8%
	Step deferral	42/134 (31.3%)	168.46	22.8%
	SWE-Router ($K = 4$)	48/134 (35.8%)	205.95	60.3%
	TACIT-SWITCH†	61/134 (45.5%)	165.11	47.3%
	Pure 27B	68/134 (50.7%)	230.51	100.0%

Because TACIT-SWITCH invokes Strong on most 4B tasks, this setting mainly demonstrates the benefit of delaying handoff even when handoff is frequent, rather than using Strong selectively.

A larger-scale check using 1,000 training, 300 development, and 274 test episodes again places TACIT-SWITCH first among the learned policies for both model pairs. Because it reuses the underlying task pool, we do not treat it as an independent replication; Appendix B.3 reports the full results.

5.3 DABENCH

InfiAgent-DABench evaluates data-analysis agents in an executable tool environment (Hu et al., 2024). We pair Qwen3.5-9B with Qwen3.6-27B, allow at most 12 model calls, and use a fixed Train140/Dev50/Test67 split; full protocol details appear in Appendix B.

TACIT-SWITCH outperforms all three learned routing baselines on held-out success while having the lowest average cost among them. It uses Strong less often than SWE-Router. The paired gain over SWE-Router is positive, but its bootstrap interval includes zero, so this result is suggestive rather than conclusive.

6 CONCLUSION AND LIMITATIONS

We studied when an agent should permanently hand control from a cheaper language model to a stronger one during a multi-step task. TACIT-SWITCH learns this policy from paired Cheap–Strong episode outcomes and coarse offline teacher intervals. It combines task features with cumulative trajectory risk through a mixture-cure threshold model and requires no teacher at deployment. The model-based study supports recovery under correct specification, while controlled experiments show that task and trajectory information are complementary. In the mechanism-based and interactive-agent experiments, TACIT-SWITCH achieves better held-out success–cost trade-offs than task-, step-, and fixed-prefix routing. A larger ALFWorld scale check likewise places TACIT-SWITCH first among the learned policies for both model pairs.

A permanent one-way handoff can be suboptimal when brief Strong assistance followed by a return to Cheap would suffice, or when different task phases favor different models. The operating point depends on an application-specified cost function; here we use parameter-weighted call count as a deployment-cost proxy. The method also relies on paired offline rollouts and teacher-provided intervals, so systematic annotation bias or distribution shift can affect the learned policy.

REPRODUCIBILITY STATEMENT

The appendices document the experimental protocols, model fitting, cost accounting, and development-only selection procedure. The accompanying artifact contains the manuscript source, exact configurations and seeds, task manifests, prompt definitions, fitted parameter files, frozen scores, numerical results, and scripts used to regenerate the reported tables and figures.

AI USE STATEMENT

OpenAI Codex assisted with code, simulation and data analysis, proof and reference checking, and manuscript editing. DeepSeek V4 Pro served as the offline teacher in Section 5. The authors verified all AI-assisted outputs and take responsibility for the final content.

REFERENCES

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=cSimKw5p6R.

V. T. Farewell. The use of mixture models for the analysis of survival data with long-term survivors. *Biometrics*, 38(4):1041–1046, 1982. doi: 10.2307/2529885.

Daniel F. Heitjan and Donald B. Rubin. Ignorability and coarse data. *The Annals of Statistics*, 19(4): 2244–2253, 1991. doi: 10.1214/aos/1176348396.

Xueyu Hu, Ziyu Zhao, Shuang Wei, Ziwei Chai, Qianli Ma, Guoyin Wang, Xuwu Wang, Jing Su, Jingjing Xu, Ming Zhu, Yao Cheng, Jianbo Yuan, Jiwei Li, Kun Kuang, Yang Yang, Hongxia Yang, and Fei Wu. InfiAgent-DABench: Evaluating agents on data analysis tasks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 19544–19572. PMLR, 2024.

Vansh Kapoor, Aman Gupta, Hao Chen, Anurag Beniwal, Jing Huang, and Aviral Kumar. TRIM: Hybrid inference via targeted stepwise routing in multi-step reasoning tasks. In *International Conference on Learning Representations*, 2026.

Anthony Y. C. Kuk and Chen-Hsin Chen. A mixture model combining logistic regression with proportional hazards regression. *Biometrika*, 79(3):531–541, 1992. doi: 10.1093/biomet/79.3.531.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *International Conference on Learning Representations*, 2025.

Dzianis Piatrashyn, Nikita Kotelevskii, Kirill Grishchenkov, Nikita Glazkov, Ivan Nasonov, Ilya Makarov, Timothy Baldwin, Preslav Nakov, Roman Vashurin, and Maxim Panov. ReDAct: Uncertainty-aware deferral for LLM agents, 2026. URL https://arxiv.org/abs/2604.07036.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*, 2021.

Seongho Son, Sangwoong Yoon, Jiahua Tang, Shuhan Wang, Lorenz Wolf, and Ilija Bogunovic. SWE-Router: Routing in multi-turn agentic software engineering tasks, 2026. URL https://arxiv.org/abs/2607.00053.

Jonathan Spencer, Sanjiban Choudhury, Matt Barnes, Matthew Schmittle, Mung Chiang, Peter Ramadge, and Siddhartha Srinivasa. Learning from interventions: Human-robot interaction as both explicit and implicit feedback. In *Proceedings of Robotics: Science and Systems*, 2020. doi: 10.15607/RSS.2020.XVI.055.

Bruce W. Turnbull. The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B*, 38(3):290–295, 1976. doi: 10.1111/j.2517-6161.1976.tb01597.x.

Lee-Jen Wei. The accelerated failure time model: A useful alternative to the Cox regression model in survival analysis. *Statistics in Medicine*, 11(14–15):1871–1879, 1992. doi: 10.1002/sim.4780111409.

Guibin Zhang, Haiyang Yu, Kaiming Yang, Bingli Wu, Fei Huang, Yongbin Li, and Shuicheng Yan. EvoRoute: Experience-driven self-routing LLM agent systems. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 38213–38225, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.1771.

A Finite-Sample Theory

A.1 Clean target and local regularity

Let O_i^0 denote the clean observed TACIT record for task i, including its task features, risk-diagnostic sequence, paired-Strong outcome, and interval or right-censoring information. Let O_i be the record actually used for fitting after possible teacher-interval corruption; all other fields agree with O_i^0. Here, "clean" includes a valid coarsening whose interval contains the latent crossing; replacing it with an invalid interval is corruption. Using a locally identifiable parameterization of (w, γ, β, s), let $\ell(\theta; O)$

denote the negative log-likelihood contribution in Equation (7) and let $\mathcal{P}(\theta)$ denote the deterministic regularizer. Define

$$M(\theta) = \mathbb{E}[\ell(\theta; O^0)] + \mathcal{P}(\theta),$$
$$\widehat{M}_n^0(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; O_i^0) + \mathcal{P}(\theta), \qquad \widehat{M}_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; O_i) + \mathcal{P}(\theta). \tag{13}$$

The simplex is represented with $d-1$ independent coordinates rather than all d softmax logits, which removes the common-shift nonidentifiability. Any smooth local coordinate for the positive scale may be used.

Assumption 1 (Local regularity and bounded teacher-interval influence)**.** The clean records $O_1^0, \ldots, O_n^0$ are i.i.d., and all recordwise losses below are twice continuously differentiable on the closed ball $\mathbb{B}_r(\theta^\dagger)$, which is contained in the interior of the local coordinate chart. Differentiation and expectation may be interchanged through order two on this ball. There are constants $r, \mu, \sigma, G_T, H_T > 0$ such that:

(i) $\theta^\dagger$ is an interior stationary point of M, and M is twice continuously differentiable with

$$\nabla^2 M(\theta) \succeq \mu I_p \quad \text{for every } \theta \in \mathbb{B}_r(\theta^\dagger).$$

(ii) The centered clean score

$$\xi_i = \nabla \ell(\theta^\dagger; O_i^0) - \mathbb{E}[\nabla \ell(\theta^\dagger; O_i^0)]$$

is directionally σ-sub-Gaussian: for every unit vector v and $\lambda \in \mathbb{R}$,

$$\mathbb{E}\exp\{\lambda v^\top \xi_i\} \le \exp(\sigma^2\lambda^2/2).$$

(iii) For the confidence level δ under consideration, with probability at least $1 - \delta/2$,

$$\sup_{\theta \in \mathbb{B}_r(\theta^\dagger)} \|\nabla^2 \widehat{M}_n^0(\theta) - \nabla^2 M(\theta)\|_{\mathrm{op}} \le \mu/4.$$

(iv) Let $\mathcal{I}_T$ denote the records with a replaced teacher interval, where $|\mathcal{I}_T| \le \rho_T n$. For every record,

$$\|\nabla \ell(\theta^\dagger; O_i) - \nabla \ell(\theta^\dagger; O_i^0)\|_2 \le G_T \mathbb{I}\{i \in \mathcal{I}_T\},$$
$$\sup_{\theta \in \mathbb{B}_r(\theta^\dagger)} \|\nabla^2 \ell(\theta; O_i) - \nabla^2 \ell(\theta; O_i^0)\|_{\mathrm{op}} \le H_T \mathbb{I}\{i \in \mathcal{I}_T\}. \tag{14}$$

Moreover, $H_T \rho_T \le \mu/4$.

For the log-normal working likelihood, sufficient local conditions for item (iv) are bounded covariates and diagnostics, a scale s bounded away from zero, and interval and right-censor likelihood masses bounded away from zero on $\mathbb{B}_r(\theta^\dagger)$. The local smoothness assumption also requires the numerical likelihood floor to remain inactive on this ball.

A.2 Proof of Theorem 1

Let

$$Z_n = \frac{1}{n}\sum_{i=1}^{n} \xi_i.$$

Choose a $1/2$-net $\mathcal{N}$ of the Euclidean unit sphere in $\mathbb{R}^p$ with $|\mathcal{N}| \le 5^p$. For any fixed $v \in \mathcal{N}$, the directional sub-Gaussian assumption gives

$$\mathbb{P}\big(|v^\top Z_n| > t\big) \le 2\exp\left(-\frac{nt^2}{2\sigma^2}\right).$$

Taking

$$t = \sigma\sqrt{\frac{2\{p\log 5 + \log(4/\delta)\}}{n}}$$

and applying a union bound over $\mathcal{N}$ shows that, with probability at least $1-\delta/2$,

$$\max_{v\in\mathcal{N}} |v^\top Z_n| \le t.$$

For any vector z, the $1/2$-net property implies $\|z\|_2 \le 2\max_{v\in\mathcal{N}} |v^\top z|$. Hence

$$\|Z_n\|_2 \le 2\sigma\sqrt{\frac{2\{p\log 5+\log(4/\delta)\}}{n}}. \tag{15}$$

By Assumption 1(i), $\nabla M(\theta^\dagger)=0$. The deterministic regularizer appears identically in M and $\widehat{M}_n^0$, so it cancels after centering and

$$\nabla\widehat{M}_n^0(\theta^\dagger) = Z_n.$$

Let $\mathcal{I}_T$ be the set of teacher-interval replacements. The recordwise gradient bound and the triangle inequality give

$$\begin{aligned}\|\nabla\widehat{M}_n(\theta^\dagger)\|_2 &\le \|\nabla\widehat{M}_n^0(\theta^\dagger)\|_2 + \frac{G_T|\mathcal{I}_T|}{n}\\ &\le 2\sigma\sqrt{\frac{2\{p\log 5+\log(4/\delta)\}}{n}} + G_T\rho_T = a_n(\delta).\end{aligned} \tag{16}$$

On the clean empirical-Hessian event in Assumption 1(iii),

$$\nabla^2\widehat{M}_n^0(\theta) \succeq (3\mu/4)I_p \quad \text{throughout } \mathbb{B}_r(\theta^\dagger).$$

The recordwise Hessian bounds and item (iv) imply

$$\sup_{\theta\in\mathbb{B}_r(\theta^\dagger)} \|\nabla^2\widehat{M}_n(\theta) - \nabla^2\widehat{M}_n^0(\theta)\|_{\mathrm{op}} \le H_T\rho_T \le \mu/4.$$

Weyl's inequality therefore yields

$$\nabla^2\widehat{M}_n(\theta) \succeq (\mu/2)I_p \quad \text{for all } \theta\in\mathbb{B}_r(\theta^\dagger). \tag{17}$$

By continuity and compactness, $\widehat{M}_n$ attains a minimum on the closed ball. For any u with $\|u\|_2=r$, the fundamental theorem of calculus and Equation (17) give

$$\begin{aligned}\left\langle \nabla\widehat{M}_n(\theta^\dagger+u), u\right\rangle &= \left\langle \nabla\widehat{M}_n(\theta^\dagger), u\right\rangle + \int_0^1 u^\top\nabla^2\widehat{M}_n(\theta^\dagger+tu)u\,dt\\ &\ge -a_n(\delta)r + \frac{\mu r^2}{2} > 0.\end{aligned} \tag{18}$$

Equation (18) rules out boundary minimizers, so the minimizer is interior and stationary. Equation (17) makes the objective $\mu/2$-strongly convex on the ball, hence the minimizer is unique there.

Let $\Delta = \widehat{\theta}-\theta^\dagger$. Strong monotonicity of the gradient and stationarity of $\widehat{\theta}$ yield

$$\begin{aligned}\frac{\mu}{2}\|\Delta\|_2^2 &\le \left\langle \nabla\widehat{M}_n(\widehat{\theta}) - \nabla\widehat{M}_n(\theta^\dagger), \Delta\right\rangle\\ &= -\left\langle \nabla\widehat{M}_n(\theta^\dagger), \Delta\right\rangle \le a_n(\delta)\|\Delta\|_2.\end{aligned} \tag{19}$$

If $\Delta\neq 0$, division by $\|\Delta\|_2$ proves Equation (9); the case $\Delta=0$ is immediate.

Finally, fix any evaluation task and decision step with features x and observed diagnostics $u(1{:}t)$. The line segment between $\theta^\dagger$ and $\widehat{\theta}$ remains in the ball, so the uniform Lipschitz condition in the theorem gives

$$|q_{\widehat{\theta}}(x,u(1{:}t)) - q_{\theta^\dagger}(x,u(1{:}t))| \le L_q\|\widehat{\theta}-\theta^\dagger\|_2 \le \frac{2L_q a_n(\delta)}{\mu}.$$

Because the task and step were arbitrary, this proves Equation (10). The score-concentration and Hessian events each fail with probability at most $\delta/2$; their intersection therefore has probability at least $1-\delta$, completing the proof.

B Protocol Details

B.1 Controlled Statistical Protocols

Model-based AB–AFT study. The recovery experiment uses 500 independent fits at each $n \in \{500, 2{,}000, 8{,}000\}$, three optimizer restarts, at most 500 iterations, and a fresh 10,000-task evaluation sample. The matched-noise experiment uses 100 paired replicates, each with 4,000 training and 4,000 test tasks, two restarts, and at most 350 iterations. Both experiments fix the scalar risk weight at one and use no regularization. For $AB = 11$, they use the right-censoring contribution with observed $B_i = 1$. Pilot calibration and evaluation use disjoint random streams; every fit trained with corrupted supervision is evaluated on clean held-out data.

Mechanism-based data generation and supervision. Each task has ten opportunities and succeeds after at least eight correct actions. Its execution state is on-track, recoverable, or off-track. Cheap and Strong follow fixed state-transition and action-success probabilities, with Strong more likely to recover from degraded states. A post-step Cheap risk signal provides partial evidence about the realized path and can affect routing only from the following step onward.

On each Pure-Cheap trajectory, the simulator searches post-action checkpoints 1–5 and records the earliest candidate handoff checkpoint at which the task remains unfinished, cumulative Cheap risk is at or above the task-specific threshold, the resulting state is recoverable or off-track, and Strong's probability of a correct next action exceeds Cheap's by at least 0.08. Under operational AB supervision, this candidate is retained only for $AB = 01$. A retained candidate becomes a one-step interval at that post-action checkpoint. An $AB = 01$ record without a candidate contributes only its observed $B_i = 1$ status; $AB = 11$ is right-censored at the horizon, and $AB \in \{00, 10\}$ enters the cure component. The latent threshold and candidate checkpoint do not enter actor dynamics, risk emissions, or terminal success.

Each reported replicate contains 8,000 training, 4,000 development, and 20,000 test tasks. The main comparison and feature ablation use 100 replicates, while each nonbaseline sensitivity setting uses 50. All methods use costs $c_C = 1, c_S = 3$ and the common development cap 16. Pilot calibration and reported evaluation use disjoint random streams. The sensitivity experiments vary only learner-visible task features or risk-emission noise, leaving actor dynamics and coupled endpoint outcomes unchanged.

Compared policies. We compare four learned policies on the same coupled tasks. Task Router estimates the probability of Pure-Cheap success from task features and chooses Cheap or Strong before execution. Step Deferral uses the latest observed Cheap risk to assign only the following step to Strong, after which control returns to Cheap. The SWE-Router analogue predicts Pure-Cheap success from task features and the first $K \in \{1, 2, 3, 4\}$ Cheap risk observations, then decides either to continue with Cheap or to restart with Strong. TACIT-SWITCH combines task features with cumulative Cheap risk and, at its first score crossing, hands control permanently to Strong from the current state. At deployment, none of these policies observes latent states, teacher labels, Strong outcomes, or terminal rewards. We use 101 quantiles of the training scores as candidate thresholds; each policy is selected on development under the common cost cap and frozen before test, with K and the threshold selected jointly for the fixed-prefix policy. The SWE-Router baseline is a structural analogue because the simulator provides numerical features rather than language-model trajectories. Its fixed-K diagnostics use the same train/development/test splits.

Diagnostic checks show that cumulative risk predicts terminal Pure-Cheap failure better than task features alone or the latest risk value (AUC .774 versus .605 and .691). Across the 100 held-out replicates, the mean fraction of $AB = 01$ tasks with a candidate handoff checkpoint is 80.1%. Among those tasks, permanently handing off immediately after the marked Cheap step produces terminal success in 85.6% of cases.

Sensitivity results. Across all sensitivity settings, TACIT-SWITCH has higher held-out success than the best routing baseline selected on development data at comparable cost, with larger gains as online-risk information becomes more informative.

B.2 INTERACTIVE-AGENT PROTOCOLS

Interactive-agent cost accounting. For reproducibility, the reported proxy uses common 4B-equivalent units. If Cheap model $m \in \{4\text{B}, 9\text{B}\}$ makes $n_{m,i}$ calls, the 27B Strong model makes $n_{27,i}$ calls, and the 7B value head is invoked $h_i \in \{0, 1\}$ times, then

$$\text{Cost}_i = c_m n_{m,i} + 6.75 n_{27,i} + 1.75 h_i, \qquad c_{4\text{B}} = 1,\ c_{9\text{B}} = 2.25. \tag{20}$$

The proxy selects the development operating point and reports realized test cost; it does not enter score fitting. SWE-Router includes its Cheap prefix, one head call when invoked, and the full Strong restart. Offline teacher labeling is excluded from deployment cost.

ALFWorld. The fit/select/test split is SEEN100/SEEN40/ UNSEEN134. For the 4B–27B pair, the training counts in $(AB = 00, 01, 10, 11)$ order are $(34, 58, 1, 7)$; for 9B–27B they are $(35, 43, 0, 22)$. Every $AB = 01$ training episode receives a teacher interval. The task router evaluates thresholds $\{0, 0.05, 0.10, \ldots, 0.95, 1.01\}$; selected-action-NLL deferral targets an average of $k \in \{0, 5, 10, 15, 20, 25, 30, 35, 40\}$ Strong-deferred steps per training task; and TACIT-SWITCH uses $\alpha \in \{0.1, 0.2, \ldots, 0.9\}$. Selection maximizes SEEN40 success subject to at most 90% of pure-27B proxy cost, breaking ties by lower cost. Development selects $\alpha = 0.2$ for 4B and $\alpha = 0.3$ for 9B; both values are frozen before UNSEEN134 evaluation.

DABench. We use a fixed custom split of a 257-task DAEval subset into Train140, Dev50, and Test67, with Test67 held out from all fitting and development selection. We do not present this subset or split as an official benchmark partition. The task-router target rates are $\{0, 0.125, \ldots, 0.875, 1\}$, where each rate targets that fraction of training tasks for Strong execution. Selected-action-NLL deferral targets an average of $k \in \{0, 0.75, 1.5, 2.25, 3, 3.75, 4.5, 5.25, 6\}$ deferred steps per training task. For TACIT-SWITCH, a deterministic grid combines target train-switch rates with train-score quantiles in $\{0, 0.1, \ldots, 1\}$. Using training data only, we deduplicate candidates by their induced policies and thin them to nine points, including both endpoints. Dev50 then selects the highest-success candidate using at most 80% of pure-27B proxy cost; remaining ties are broken by lower cost, intervention rate, and target rate. The selected settings are a 62.5% Task Router target, $k = 1.5$, and a 40% TACIT-SWITCH train-switch target corresponding to $\alpha \approx 0.163$. The nine prespecified TACIT-SWITCH candidates are shown on Test67 only as a diagnostic curve; the reported result uses the development-selected 40% candidate.

Compared policy implementations. Task Router fits a logistic model for Pure-Cheap success from task-level features and chooses Cheap or Strong before execution. Step Deferral thresholds the Cheap model's selected-action negative log-likelihood and delegates only the current action to Strong. The SWE-Router-style baseline fine-tunes a Qwen2.5-Coder-7B-Instruct value head with LoRA to predict Pure-Cheap success from the task and the visible Cheap prefix at $K \in \{1, 2, 3, 4\}$; it then decides either to continue with Cheap or to restart with Strong. TACIT-SWITCH fits the censored threshold model described in Appendix D and permanently hands off from the current state. All model fitting uses training data, and every policy parameter is selected on development under the benchmark cost cap and frozen before test. The fixed-prefix policy never receives Strong outcomes, teacher labels, or TACIT-SWITCH scores as inputs, and its Cheap prefix, value-head call, and Strong restart are all charged by Equation (20). Because the available ALFWorld cache omits hidden Cheap reasoning and the original paraphrase augmentation is unavailable, we call it a SWE-Router-style reproduction.

For the selected operating points, paired task-level bootstrap intervals use 10,000 with-replacement resamples and percentile endpoints. The 95% intervals for the success-rate difference between TACIT-SWITCH and SWE-Router are $[17.2, 35.1]$ percentage points for ALFWorld 4B–27B, $[2.2, 17.9]$ for ALFWorld 9B–27B, and $[-3.0, 14.9]$ for DABench 9B–27B.

Offline teacher annotation. For each $AB = 01$ training episode, an offline teacher observes the task, the failed Cheap trajectory, and the successful Strong trajectory. It returns the earliest range of post-action checkpoints at which failure risk is clear and recovery remains feasible, together with a recommended checkpoint in that range. A mark at step t denotes handoff after observing Cheap's action and result at step t and before step $t + 1$. The primary experiments treat the recommended checkpoint as the one-step interval $[t, t]$, whereas the larger ALFWorld study retains the complete range. Teacher explanations are not used for fitting.

Decision timing. ALFWorld executes the Cheap action, observes the resulting state, updates risk, and, if the threshold is crossed, gives the next action to the Strong agent. DABench instead scores an unexecuted Cheap proposal. If the threshold is crossed, it discards that proposal and asks the Strong agent to act from the prior tool history; otherwise it executes the proposal. For DABench, a teacher mark after step t is aligned with the proposal at step $t + 1$ to match this pre-execution decision point.

B.3 Larger-Sample ALFWorld Scale-Up Evaluation

We use this study as a scale check rather than as a separate benchmark. The fit and development sets contain 1,000 and 300 episodes from the official training split from disjoint parent-task groups; groups overlapping the official validation sets are excluded. Test274 is the union of all 140 valid-seen and 134 valid-unseen tasks. In the 4B–27B training set, the counts for $AB = 00, 01, 10, 11$ are $371/576/2/51$; in the 9B–27B set, they are $351/444/22/183$. DeepSeek V4 Pro labels every AB01 training trajectory. Unlike the primary ALFWorld fit, the scale-up fit uses the complete teacher interval $[\tau_{\text{low}}, \tau_{\text{high}}]$; most of these intervals span multiple steps.

All candidates are selected on Dev300 under the same cost cap, 180.77 4B-equivalent units, and then frozen. Development selects $\alpha = 0.3$ for 4B–27B and $\alpha = 0.2$ for 9B–27B; the SWE-Router-style baseline selects $K = 4$ in both pairs. Test costs are computed from actual model-call counts under Equation (20).

Figure 5 shows that TACIT-Switch has the highest success point estimate among learned policies in both model pairs. It is also less costly than Task Router and SWE-Router. Relative to Step Deferral, it costs 8.8 additional units and gains 11.7 percentage points for 4B–27B. For 9B–27B, the corresponding differences are 6.1 units and 17.9 percentage points.

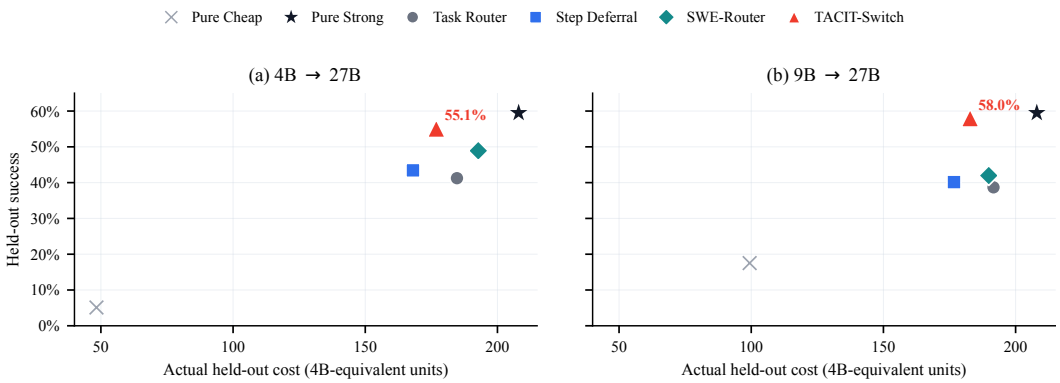

Figure 5: **Larger-sample ALFWorld success–cost operating points.** All policies are frozen on Dev300 and evaluated once on the same Test274 tasks. Costs follow Equation (20) using realized executed calls.

Table 3: **Larger-sample ALFWorld operating points on Test274.** Policies are selected on Dev300. Intervention is the fraction of tasks that invoke Strong; Strong share is the fraction of all model calls made by Strong. † marks the proposed method.

Pair	Policy	Success	Avg. cost	Strong share	Intervention
4B–27B	Pure 4B	14/274 (5.1%)	48.32	0.0%	0.0%
	Task Router	113/274 (41.2%)	184.69	70.0%	75.5%
	Step Deferral	119/274 (43.4%)	168.05	35.0%	100.0%
	SWE-Router ($K = 4$)	134/274 (48.9%)	192.75	70.5%	81.8%
	TACIT-Switch†	151/274 (55.1%)	176.87	79.1%	95.6%
	Pure 27B	163/274 (59.5%)	208.02	100.0%	100.0%
9B–27B	Pure 9B	48/274 (17.5%)	99.40	0.0%	0.0%
	Task Router	106/274 (38.7%)	191.64	63.3%	63.5%
	Step Deferral	110/274 (40.1%)	176.69	26.9%	98.9%
	SWE-Router ($K = 4$)	115/274 (42.0%)	189.85	55.9%	61.3%
	TACIT-Switch†	159/274 (58.0%)	182.79	71.0%	92.7%
	Pure 27B	163/274 (59.5%)	208.02	100.0%	100.0%

B.4 Feature Definitions and Optimization

Task features are standardized using training-split means and scales; risk diagnostics are used directly as nonnegative values. No development or test outcome enters feature scaling or fitting.

Baseline features and fitting. The ALFWorld task router uses five binary descriptors: whether the final receptacle is openable, whether it is a surface, whether the task needs an external tool, whether the requested state is cold, and whether the target is food. It fits an L2-penalized logistic model by Newton updates ($\lambda = 1$, at most 100 iterations, tolerance 10^{-9}). The DABench task router instead uses operation breadth and a transformation-workflow indicator, fitted by L2-logistic regression with $C = 0.1$, LBFGS, and at most 2000 iterations. The step-deferral baseline uses only selected-action NLL and the benchmark-specific k grids stated above.

ALFWorld TACIT-SWITCH features. The 12-dimensional task vector encodes requested state, task form, target category, receptacle type, and the visible-entity count. Five step-level diagnostics capture repeated actions, short action cycles, selected-action uncertainty $1 - \exp(-\mathrm{NLL})$, invalid generations or fallbacks, and unchanged observations with zero reward. Candidate margin is not used, and missing selected-action uncertainty is mapped to zero. Joint fitting uses Adam with five restarts, $\lambda = 0.01$, and early stopping.

DABench TACIT-SWITCH features. The two task features are normalized answer arity and an indicator that the required output specifies decimal precision. Eight diagnostics computed from each proposal capture format or protocol violations, execution failures, repetition, stalled answers, pressure from the remaining tool budget, lack of recent progress, and semantic mismatch with the task. Joint fitting uses Adam with six restarts, $\lambda = 0.03$, and early stopping.

C Robustness to Noisy Teacher Supervision

Theorem 1 covers incorrect teacher intervals only. As an additional empirical stress test, we also corrupt the observed paired Strong outcome while keeping evaluation clean. This study fixes $n = 4{,}000$ and uses 100 paired replicates per condition. Among $AB = 01$ records with a one-step interval, teacher-interval noise moves a fraction ρ_T^{timed} to an adjacent step. Strong-outcome noise flips B_i for a fraction ρ_C of tasks. The matched joint conditions set $\rho_T^{\mathrm{timed}} = \rho_C$, and no test measurement is corrupted. If n_T of the n records are timed, the record-level corruption fraction in Theorem 1 is $\rho_T = (n_T/n)\rho_T^{\mathrm{timed}}$.

Clean $q(t \mid x)$ RMSE is 0.0075 ± 0.0005. At 20% corruption, it is 0.0077 ± 0.0005 under teacher-interval noise, 0.0208 ± 0.0007 under Strong-outcome noise, and 0.0209 ± 0.0007 under matched joint noise. Figure 6 plots these errors against the direct corruption rate; for the joint condition, both ρ_T^{timed} and ρ_C equal the displayed rate. Under this design, corrupting the paired Strong outcome has a larger effect than shifting the teacher interval by one step.

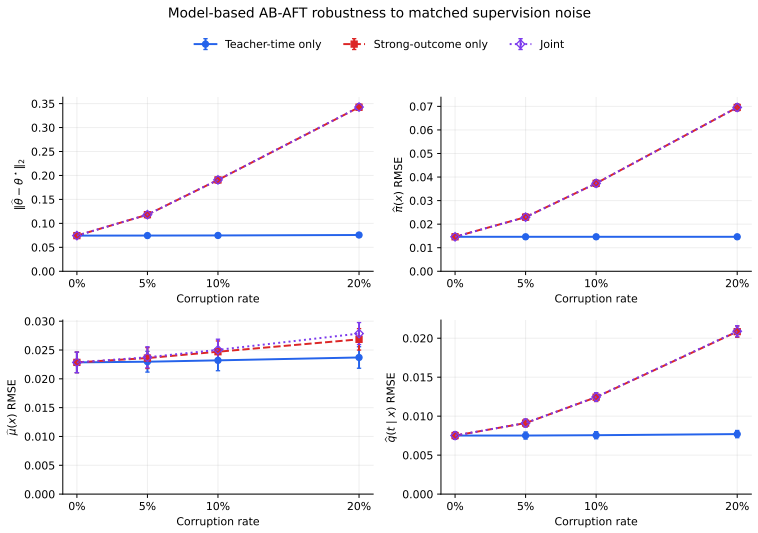

Figure 6: Matched teacher-interval and Strong-outcome noise under the correctly specified model-based DGP. Points show mean estimation or held-out prediction error over 100 paired replicates; bars show Student-t 95% Monte Carlo confidence intervals for the replicate means. The horizontal coordinate is the direct corruption rate; for the joint condition, both noise rates equal the displayed value.

D Working Objective and Inference

The implementation parameterizes $w = \mathrm{softmax}(\widetilde{w})$ and uses a softplus transform to enforce $s > 0$. With the per-episode contributions in Equation (7), it applies a hard numerical floor $\varepsilon = 10^{-12}$ to interval masses and final per-record likelihoods. Let $\overline{\mathcal{L}}_i$ denote the resulting implemented contribution. The optimized objective is

$$-\frac{1}{n}\sum_{i=1}^{n}\log \overline{\mathcal{L}}_i + \lambda\left(\|\gamma_{-0}\|_2^2 + \|\beta_{-0}\|_2^2\right). \tag{21}$$

The intercepts, scale, and risk-weight logits are unpenalized. We optimize all parameters jointly and retain the restart with the lowest objective value. For the parameterization used in Theorem 1, we fix one logit to remove the softmax common-shift nonidentifiability and assume that the numerical floors remain inactive in the local neighborhood.

At inference, both benchmarks update cumulative risk and evaluate Equation (12) once per cheap-agent checkpoint; Algorithm 1 specifies the benchmark-dependent checkpoint semantics.

Algorithm 1 Benchmark-aware permanent handoff

Require: fitted parameters, threshold α, cheap policy π_c, strong policy π_s
$R \leftarrow 0$; controller $\leftarrow \pi_c$
while episode is unfinished and budget remains **do**
  if controller is π_s **then**
    execute one strong-agent action and continue
  end if
  obtain cheap checkpoint evidence e_t ▷ post-action in ALFWorld; proposal in DABench
  $R \leftarrow R + \hat{w}^\top u(e_t)$; $q \leftarrow \hat{\pi} F_{\hat{\beta},\hat{s}}(R \mid x)$
  if $q \geq \alpha$ **then**
    controller $\leftarrow \pi_s$ permanently
    discard e_t and execute the Strong action immediately if the proposal is unexecuted
  else
    execute e_t if it is an unexecuted proposal
  end if
end while