LEARNING TO FOLLOW INSTRUCTIONS FROM IMPERFECT GUIDANCE: REFERENCE-GUIDED REINFORCEMENT LEARNING WITH DEMONSTRATION ANCHORING

Anonymous authors
Paper under double-blind review

ABSTRACT

Planners, experts, and language models can guide instruction-following reinforcement learning (RL), but may receive inaccurate state descriptions or be impractical to query at deployment. To address this problem, we initialize policies by imitation and fine-tune them with rewards, temporary action-probability targets from a frozen reference on learner-visited states and continuing demonstration supervision (anchoring). We train policies that learn from imperfect guidance and act without it. In BabyAI, with left/right-swapped inputs to an online language guide, anchoring raises mean final success from 18.3% to 27.7% (five seeds, one initializer). Separately, across five independent BabyAI datasets and model fits, the framework improves time-averaged success by 6.5 percentage points over anchoring alone and 8.1 over the unchanged initial policy. Exchanging the same targets among states sharing an instruction lowers time-averaged success by 4.6 points relative to own-state guidance. Another BabyAI study restricts exchanges to targets sharing the most probable action, reducing this gap from 5.1 to 0.7 points; further benefit from exact own-state probabilities remains unresolved. In CrafText, own-state guidance beats reassignment, uniform targets and anchoring alone, but not the initial policy on main-study time-averaged success, despite higher final success.

1 INTRODUCTION

Planners, trained experts and large language models (LLMs) can guide instruction-following reinforcement learning (RL), but may be costly or use information unavailable to the policy (Hu & Sadigh, 2023; Carta et al., 2023; Zhou et al., 2024). Advice can also mismatch the learner's situation: an inaccurate description of a red key's location may suggest the wrong turn for *pick up the red key*. How can an agent learn from imperfect guidance and then act without it? We study this question in BabyAI-Text (Chevalier-Boisvert et al., 2019; Carta et al., 2023) and CrafText (Volovikova et al., 2025), using *references* that supply instruction-conditioned action distributions during training.

Behavior cloning (BC) initializes the policy by supervised prediction of demonstrated actions from observations and instructions (Ross et al., 2011, Section 2.1). Proximal policy optimization (PPO; Schulman et al., 2017) then learns from rewards with two additional signals: a Kullback–Leibler (KL) divergence loss (Kullback & Leibler, 1951) toward a frozen reference's action distributions on learner-visited states, and a *demonstration anchor* continuing supervised learning on demonstrations (Hester et al., 2018; Rajeswaran et al., 2018). Reference guidance shapes early learning, then ends so it no longer constrains reward optimization; anchoring continues to limit drift on demonstration inputs. Deployment uses only the policy (Figure 1). *Anchor only* omits reference KL; *retained BC* deploys the initializer without PPO.

We evaluate learning under corrupted guide descriptions, then ask what matching advice to states contributes beyond retaining the same targets for an instruction. Which reference supplies useful guidance: expert or language-model fits, their combination, or a frozen copy of the initial BC policy?

Within each batch of interactions collected before an update, we exchange reference outputs between states sharing an instruction. Each instruction keeps exactly the same collection of vectors, hence

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.

the same average action probabilities; vectors are exchanged, not replaced by that average. Only the state–vector pairing changes. *Aligned* uses own-state outputs; *Reassigned* exchanges them: a state requiring a turn may receive the action probabilities computed for another state where moving toward the key is appropriate. Exchanging only vectors with the *same top (most probable) action* tests whether the exact own-state vector helps beyond its recommendation. Each condition trains its own policy, so this controls target assignment within a rollout, not the states that different policies later visit.

We use Qwen3-32B (Qwen; Yang et al., 2025) online in BabyAI and for offline reference construction in CrafText. Our contributions are as follows:

- **Training scheme.** We fine-tune a behavior-cloned policy with PPO, a KL penalty toward a frozen reference on learner-visited states, and supervised learning on demonstrations. The KL weight decays to zero; demonstration supervision continues. Only the policy is deployed.
- **Learning benefits.** With left/right-swapped guide inputs, anchoring raises our InstructPPO-style online-Qwen baseline (Hu & Sadigh, 2023) from 18.3% to 27.7% mean final success (Table 1). Separately, across five BabyAI datasets and model fits, guidance with anchoring improves time-averaged success by 6.5 points over Anchor only and 8.1 over retained BC (Table 2).
- **Guidance properties and reference choice.** Own-state targets outperform reassignment and uniform targets in both suites. In a separate BabyAI study, same-top exchange reduces the gap to Aligned from 5.1 to 0.7 points (Table 4). In CrafText, using the fixed BC initializer as reference (*Frozen-BC*) outperforms expert and expert–Qwen-mixture references fitted on fewer labelled states (Table 6).

2 Related Work

Training-time guidance. Language guides policies or exploration (Luketina et al., 2019; Carta et al., 2023; Du et al., 2023). Distillation transfers teacher distributions; kickstarting combines this with RL (Rusu et al., 2016; Schmitt et al., 2018). Behavior priors regularize toward reusable policies (Tirumala et al., 2022; Galashov et al., 2019; Singh et al., 2021); a frozen initializer supplies targets in KL-regularized language-model RL (Ouyang et al., 2022; Korbak et al., 2022). InstructPPO combines policy-to-LLM KL, pretrained initialization and an annealed coefficient (Hu & Sadigh, 2023). Proximal Policy Distillation weights teacher-to-student KL by a lower-clipped current/rollout-policy action-probability ratio (Spigler, 2025). LLM4Teach anneals guidance before student-only deployment (Zhou et al., 2024); LaGO fits an LLM-backed prior offline (Liu et al., 2026b).

Continuing demonstrations. Demonstration losses also accompany reward learning (Hester et al., 2018). Demonstration-augmented policy gradient (DAPG) initializes with BC and decays an additional demonstration term (Rajeswaran et al., 2018). InstructQ tests corrupted prior outputs (Hu & Sadigh, 2023, Appendix C.3); we corrupt guide inputs and vary anchoring. Our framework retains demonstration supervision after reference guidance ends. Demonstration-derived KL can impede learning (Rudner et al., 2021); we therefore also compare against retaining BC without further training.

Alignment and attribution. State-matched routing selects advice compatible with the reached execution state (Liu et al., 2026a); MemOPD restores the memory and context used to generate each action before scoring it for distillation (Liu et al., 2026c). Self-distillation studies test whether extra teacher-only information explains gains (Shrestha & Tessier, 2026); adaptive composition selects or modifies priors for downstream tasks (Rietz et al., 2026). Our BC-budget test extends BC-to-RL studies (Ramrakhya et al., 2023; Wagenmaker et al., 2025) to state-assignment benefits.

3 Reference-Guided Training Framework

3.1 Policy inputs and deployment

The environment supplies instruction m and partial observation o. A *rollout* collects interactions before an update; a *row* is one time step. At PPO update u, row i has input $x_{u,i} = (o_{u,i}, m_{u,i}, h_{u,i})$,

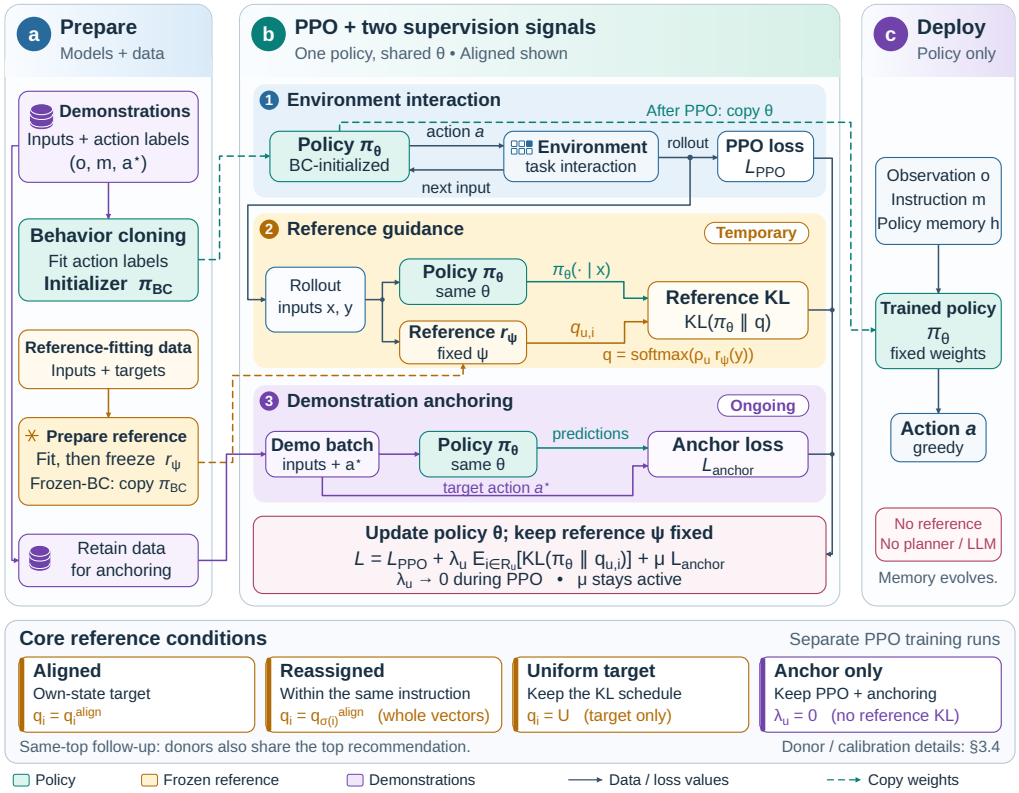


Figure 1: **Training and reference conditions.** Policy boxes share θ; reference weights ψ stay fixed. Solid arrows carry data/losses; dashed arrows copy weights. Inputs x, y include their model's memory where used (Section 3). Lower cards are separate runs: row i receives donor $\sigma(i)$'s vector; U is uniform. Sharpness ρ shapes targets; λ, μ weight reference KL and anchoring. Core targets omit update index u. Reference KL ends; anchoring continues.

including recurrent memory $h_{u,i}$. The policy π_θ maps it to action probabilities using parameters θ. Reference input $y_{u,i}$ is $(o_{u,i}, m_{u,i})$ in BabyAI; CraftText adds separate reference memory, following learner observations and resetting at episode boundaries. Deployment updates policy memory, not parameters.

BabyAI uses a simulator-access planner (Chevalier-Boisvert et al., 2019); CraftText uses a PPO-trained expert (Appendix C.4). PPO copies BC parameters, resetting optimizer and memory (Appendix C.1). For CraftText, Qwen annotates text-rendered expert observations, instructions and rules to supply fitting targets or select expert examples to sample more often (Appendices N, C.5). BC and anchoring remain expert-derived; PPO queries the fitted reference, not Qwen.

3.2 Reference construction and guidance sources

For A actions in $\mathcal{A}$, reference parameters ψ map y to logits $r_\psi(y) \in \mathbb{R}^A$, action scores normalized by softmax. BabyAI fits planner actions by weighted negative log-likelihood:

$$L_{\text{ref}}(\psi) = \mathbb{E}_{(o,m,a^\star)\sim D_w}\left[-\log \operatorname{softmax}(r_\psi(o,m))_{a^\star}\right] + \frac{\beta_r}{A}\,\mathbb{E}_{(o,m)\sim D_w}\left[\|r_\psi(o,m)\|_2^2\right]. \quad (1)$$

Here D_w samples weighted offline training rows, $a^\star$ is the action label, and $\beta_r \geq 0$ penalizes mean squared logits. BabyAI has six actions, with capped inverse-frequency sampling upweighting rare labels (Appendix E). CraftText's recurrent 17-action reference minimizes $-\sum_{a\in\mathcal{A}} p^{\text{fit}}(a) \log \operatorname{softmax}(r_\psi(y))_a$, averaged over fitted positions, without the logit penalty. Here p^{fit} is the expert vector or places probability one on the expert/Qwen action for hard-label fits. Fitting and checkpoint rules are in Appendix C.1, with changes in Table 10.

3.3 TRAINING OBJECTIVE

Let $z_{u,i} = r_\psi(y_{u,i}) \in \mathbb{R}^A$ be the suite-specific reference logits. For Aligned, the training-time target is

$$q_{u,i}^{\text{align}} = \text{softmax}(\rho_u z_{u,i}), \tag{2}$$

Sharpness $\rho_u \geq 0$ gives uniform targets at zero, native softmax at one, and sharper targets above one. We write $q_{u,i}$ for a condition's target. L_{PPO} includes policy, value and entropy terms; $\lambda_u \geq 0$ weights reference KL and $\mu \geq 0$ weights anchoring (Appendix C.3). Averaging over rollout rows $\mathcal{R}_u$, PPO minimizes

$$L = L_{\text{PPO}} + \lambda_u \, \mathbb{E}_{i \in \mathcal{R}_u} \left[\text{KL}\left(\pi_\theta(\cdot \mid x_{u,i}) \,\|\, q_{u,i}\right) \right] + \mu L_{\text{anchor}}. \tag{3}$$

For BabyAI, D_w^{anchor} samples individual observations from the full labelled pool with replacement and capped inverse-action-frequency weights; reference fitting restricts D_w to its training split (Appendix E). The anchor minimizes

$$L_{\text{anchor}} = \mathbb{E}_{(o,m,a^\star) \sim D_w^{\text{anchor}}} \left[-\log \pi_\theta(a^\star \mid o, m, h{=}0) \right]. \tag{4}$$

BabyAI anchor examples use zero memory. CrafText instead averages labelled-action cross-entropy over 64-step sequences, propagating recurrent state from zero and respecting recorded resets (Appendix C.1). PPO rollouts carry memory. The direction $\text{KL}(\pi_\theta \| q) = \sum_{a \in \mathcal{A}} \pi_\theta(a \mid x) \log[\pi_\theta(a \mid x)/q(a)]$ penalizes policy mass where targets assign low probability. Targets are fixed during an update; only the policy receives these gradients. Section 4 gives schedules.

3.4 REFERENCE-PROPERTY INTERVENTIONS

A *donor* supplies the vector compared with a *recipient*'s policy. Let $G_{u,m} = \{i_1 < \cdots < i_k : m_{u,i_j} = m\}$ collect the k rows sharing instruction m. The map σ selects each recipient's donor. In BabyAI, Reassigned takes the next row, $\sigma_{u,m}(i_j) = i_{j+1}$, wrapping the last to the first, $\sigma_{u,m}(i_k) = i_1$. A one-row group keeps its own vector:

$$q_{u,i}^{\text{reassign}} = \text{softmax}\big(\rho_u z_{u,\sigma_{u,m}(i)}\big), \quad \sigma_{u,m}(i) = i \text{ if } |G_{u,m}| = 1, \quad m_{u,\sigma_{u,m}(i)} = m_{u,i}. \tag{5}$$

Shared ρ_u preserves each instruction's target collection with repetitions (its *multiset*) after softmax. BabyAI groups steps by environment, so most states receive the next time step's target; a final row takes the next occurrence of its instruction. CrafText and BabyAI's *random-cycle Reassigned* control instead randomize the ordering with a separate generator, forming one cycle in which every row donates exactly once (Sattolo, 1986); only singletons retain self. Mappings stay fixed across PPO passes (Appendices F, L).

BabyAI's *Frozen-BC reference* guides PPO with the initializer at $\rho = 1$ (native probabilities) and separate episode-reset memory. It retains the coefficient schedule and anchor but omits sharpness warm-up (Appendix L).

Preserving the top recommendation. Within a rollout, let $c_i = \arg\max_a z_{u,i,a}$ be row i's top action, breaking ties by lowest action index. A row is *eligible* only if another has the same instruction and top action:

$$\mathcal{E} = \{i : \exists j \neq i,\ m_{u,j} = m_{u,i},\ c_j = c_i\}.$$

Both controls use these same eligible rows. Unrestricted reassignment randomly orders them within each instruction and takes the next vector, wrapping around; *same-top-action reassignment* does this separately for each top action. Other rows keep their own vectors. For example, probability vectors $(.60, .30, .10)$ and $(.50, .49, .01)$ recommend the same action but impose different soft constraints. Targets and mappings stay fixed through PPO passes.

Combining expert and Qwen fitting targets. To test whether Qwen probabilities add useful guidance beyond expert targets, the CrafText mixture study fits four references on matched positions and budgets. Qwen generates one of 17 action-code tokens; normalizing its 17 processed action log probabilities by softmax gives p_i^Q. At fitting position i, expert probabilities p_i^E, Qwen probabilities p_i^Q and uniform $U(a) = 1/A$ give the fitting targets

$$p_i^E, \qquad \frac{p_i^E + p_i^Q}{2}, \qquad \frac{p_i^E + U}{2}, \qquad \frac{p_i^E + p_{\sigma^Q(i)}^Q}{2}. \tag{6}$$

These are expert, expert–Qwen, expert–Uniform (softening only) and expert–reassigned-Qwen targets. In the fourth, σ^Q cycles Qwen vectors within each instruction and data split, randomly once before fitting; singletons retain self. This preserves Qwen vectors, not mixture vectors. Reassigning the fitted reference during PPO instead exchanges its rollout outputs.

Coefficient-scale checks. Calibration rescales the KL coefficient, not targets, to check sensitivity to the mean loss magnitude when targets change. Global calibration rescales BabyAI's coefficient from completed logs. For CrafText, let $\bar{D}_u^{\text{align}}$ and $\bar{D}_u^{\text{reassign}}$ be KL means over all rollout rows, scoring both targets with the collection policy before optimization. Reassigned uses

$$\lambda_u^{\text{reassign}} = \lambda_u \ \text{clip}_{[.01,10]} \left(\frac{\bar{D}_u^{\text{align}}}{\max(\bar{D}_u^{\text{reassign}}, 10^{-8})} \right).$$

The multiplier is fixed through that rollout's PPO passes, without gradients. The guards bound extreme ratios; otherwise weighted mean KL matches, not per-state losses, gradients or later visitation (Appendices M and A).

The *Uniform target* control replaces $q_{u,i}$ by $U(a) = 1/A$ under the same loss schedule. Since $\text{KL}(\pi \| U) = \log A - H(\pi)$, with entropy $H(\pi) = -\sum_a \pi(a) \log \pi(a)$, it adds scheduled entropy pressure up to a constant but does not match realized policy entropy. *Anchor only* sets $\lambda_u \equiv 0$.

4 Experimental Setup

Tasks and metrics. BabyAI-Text (Chevalier-Boisvert et al., 2019; Carta et al., 2023) has six actions and five task families (`goto`, `open`, `pickup`, `putnext`, two-step `seq`); training and evaluation use disjoint object–color combinations. CrafText on Craftax-Classic (Volovikova et al., 2025; Matthews et al., 2024) uses 17 actions and 57 canonical instructions shared across training and evaluation, without paraphrase tests. The policy observes a symbolic local map and inventory; collecting wood requires reaching and interacting with a tree (illustrations: Appendix B). Normalized area under the learning curve (AUC), our primary metric, measures time-averaged success by trapezoidal integration over training frames (environment steps), including initialization. Greedy-policy success is the fraction of episodes completing the instruction, averaged equally across families or goals. BabyAI allows 64 steps for goto/pickup and 128 for open/putnext/seq; CrafText allows 1,000. A timeout without completion is a failure. Evaluation *blocks* fix episode seeds; *checkpoints* save policies at specified steps. Endpoint success uses the final checkpoint. Tables give fractions and percentage-point differences (Appendix A.5).

Replication and decision rules. A *pipeline* comprises a dataset, BC initializer and reference; CrafText identifies it by expert/data origin. A study's policies and evaluation blocks form its *cohort*. Differences are averaged over paired blocks and PPO seeds within each pipeline; BabyAI seed identifiers match but initial task draws need not (Appendix G). Student-t intervals use five pipeline means (four degrees of freedom), not individual seeds or episodes as replications. Primary comparisons govern decisions; BabyAI has a second prespecified assignment contrast. Secondary comparisons add context. Descriptive intervals are unadjusted 95%; Table 1 instead gives means and sample standard deviations. Follow-ups with K primary comparisons use two-sided $100(1-.05/K)\%$ Bonferroni intervals (Guthrie, 2020), within each study. Follow-ups designed after earlier results fix their own new-run rules, not the full research sequence (study map: Appendix A).

BabyAI protocol. Five independently constructed pipelines each collect 1,200 planner-labelled episodes (300 each from pickup, open, putnext and seq) and fit BC and reference models. PPO interaction and evaluation include all five families, including goto. Four PPO seeds per condition train for 1.5M frames; six checkpoints use four seed-disjoint blocks of 100 episodes per family. Main-study decisions require five positive pipeline means and a positive 95% interval. Reference fitting uses a mean-squared-logit penalty with $\beta_r = 10^{-3}$; $\mu = .2$ continues throughout PPO. Cosine schedules warm ρ_u from 0 to 2 over .375M frames and decay λ_u from .05 to zero by .75M. Coefficient sensitivities vary the initial KL weight or calibrate it from logs. Random-cycle and Frozen-BC follow-ups reuse Aligned with two-comparison adjustment (Appendices C, E).

CrafText protocols. CrafText uses an expert chosen before a three-pipeline pilot; preliminary comparisons and the pilot are not pooled into the main estimates (Appendix F). Full-position references enter two five-pipeline studies: *assignment* compares Aligned/Reassigned/calibrated-Reassigned; *reference-control* compares separately trained Aligned/Anchor-only/Uniform policies using the same data and BC initializers. Each uses three paired seeds, 4.19M frames, nine checkpoints and four held-out episode blocks, requiring five positive pipeline means and a positive interval (Appendix F). CrafText uses native probabilities ($\rho_u = 1$); λ_u decreases linearly from .05 to zero over 32 of 64 PPO updates, also for Frozen-BC. Follow-ups use new evaluation blocks; compare scores within studies. Direct-label and top-action references fit eight positions per 64-step window; mixtures fit four. These *sparse* fits match Qwen's annotation budget while reading whole windows for context. Main references, BC and the anchor ($\mu = .01$) retain full-position expert supervision. Appendices H–K specify their frozen protocols and reuse.

Online Qwen and anchoring (Table 1). To test resilience to corrupted guide inputs, this study varies guidance source and anchoring. With probability $p = .9$ per Qwen description string, every `left`/`right` occurrence is swapped; this is not an action-error rate. The environment, policy observations, action definitions and planner labels stay unchanged. Fitted-only and no-reference conditions do not receive this corrupted input. *Online Qwen* queries text descriptions while KL is active. Digits 1–6 name actions; their returned token log-probabilities are softmaxed into a target. Missing digits receive the lowest observed score minus 5; a row with no valid digit contributes no Qwen KL. The unanchored cell adapts InstructPPO's policy-to-LLM KL (Hu & Sadigh, 2023) to BC initialization and decay to zero. *Fitted reference* uses Aligned's own-state outputs. *Fitted reference + online Qwen* adds the two sources' action scores before softmax, giving one KL target: $q_{u,i} = \mathrm{softmax}(\rho_u r_\psi(y_{u,i}) + .25\,\ell^Q_{u,i})$, where $\ell^Q_{u,i} \in \mathbb{R}^6$ contains Qwen's reconstructed action scores. An all-invalid Qwen response also omits the entire combined KL. This combines scores during PPO, unlike Eq. 6's offline probability mixture. Qwen also sees derived action-feasibility and target hints absent from the policy's input (Appendix D.2); sources are not information-matched. All cells, including clean-text PPO, share one planner-labelled BC initializer; sampling favors swapped-text Qwen/planner disagreements on planner trajectories and learner/planner disagreements on mixed trajectories (Appendix D.1). Five seeds use final-checkpoint evaluation. Combined/no-anchor and clean ($p = 0$) cells completed later on different hardware do not give a matched source ranking.

5 Results

5.1 Does guidance with anchoring improve learning?

Learning with corrupted guide inputs. When the online guide receives misleading descriptions, adding anchoring to the InstructPPO-style baseline raises mean success from .183 to .277 (+9.43 points; Table 1). Fitted guidance with anchoring reaches .294 (+11.13 over that baseline), changing both source and anchoring. Every anchored guided recipe exceeds Anchor only (.199). Fitted-only guidance does not receive corrupted online inputs. In the separate clean cohort, anchoring raises online Qwen from .173 to .326; the unmatched cohorts do not isolate a corruption-specific protective effect. Separate accounting across five fresh runs allocates 544.68 NVIDIA L40S GPU-hours to online Qwen plus anchoring and 2.29 to the fitted recipe, excluding offline preparation (Appendix D.3).

Benefits across independently constructed pipelines. In the separate, uncorrupted five-pipeline BabyAI study, Aligned improves AUC over Anchor only and retained BC by 6.53 and 8.06 points; all five pipelines agree (Table 2, Figure 2a).

5.2 Which properties of guidance matter?

State-specific advice. In BabyAI, Aligned beats Uniform by 3.92 points (primary) and Reassigned by 4.62 (second prespecified comparison; Table 2). Reassigned lies 1.91 points above Anchor only ($[-0.27, +4.08]$): exchanging targets removes most of the observed reference benefit. Because the default exchange usually takes the next time step's target, random-cycle exchange checks donor choice; the gap persists at 5.21 points, as do coefficient sensitivities (Appendices E, L). CrafText tests these properties without corrupting guide inputs. Aligned beats reassigned, uniform-target and anchor-only PPO in CrafText (Table 3). The reference-control study starts at success .3118; Aligned

Table 1: **Guidance sources and anchoring in BabyAI.** Final success (mean ± sample standard deviation; five PPO seeds, one initializer). Qwen-containing rows in the first block receive left/right-swapped text ($p = .9$); fitted-only and no-reference inputs are unchanged. Clean prompts are separate. Gain: descriptive with-minus-without points from unrounded means. †: InstructPPO-style baseline. Combined/no-anchor and clean cells completed later on different hardware. Fitted reference with anchoring is Aligned; No reference drops only PPO-time KL. Evaluation uses only the policy (Appendix D.1).

PPO-stage guidance	Without anchor	With anchor	Gain
Online Qwen	$.183 \pm .030^{\dagger}$	$.277 \pm .019$	$+9.43$
Fitted reference	$.195 \pm .041$	$.294 \pm .038$	$+9.92$
Fitted reference + online Qwen	$.122 \pm .050$	$.309 \pm .040$	$+18.63$
No reference	$.081 \pm .069$	$.199 \pm .023$	$+11.71$
Online Qwen, clean ($p = 0$)	$.173 \pm .049$	$.326 \pm .030$	$+15.32$

Table 2: BabyAI: Aligned AUC .23167; control AUCs in fractions, differences in Aligned-minus-control percentage points. Five-pipeline intervals: 95% for rows 1–4, 97.5% for follow-ups reusing Aligned. Bold: interval excludes zero, in either direction.

Control	Control AUC	Difference	Pipeline interval
Uniform target (primary)	.19244	$\mathbf{+3.92}$	$[+2.00, +5.85]$
Reassigned (prespecified)	.18544	$\mathbf{+4.62}$	$[+2.65, +6.60]$
Anchor only (secondary)	.16638	$\mathbf{+6.53}$	$[+3.71, +9.35]$
Retained BC (secondary)	.15110	$\mathbf{+8.06}$	$[+5.87, +10.25]$
Reassigned (random cycle)	.17960	$\mathbf{+5.21}$	$[+0.98, +9.43]$
Frozen-BC reference	.20636	+2.53	$[-0.34, +5.40]$

averages .2956 but finishes at .3241. A higher endpoint therefore coexists with lower time-averaged success than retaining BC (Figure 2b).

Only calibrated Reassigned rescales its KL coefficient to Aligned's mean loss; the uncalibrated CrafText gap is also positive, $+2.07$ points $[0.88, 3.27]$ (Appendix F).

Top recommendation versus the full vector. Same-top-action reassignment improves BabyAI AUC over unrestricted reassignment by $+4.41$ points $[2.93, 5.90]$ (descriptive 95% interval). Aligned exceeds unrestricted Reassigned by $+5.13$ $[3.21, 7.06]$, but exceeds same-top exchange by only 0.72 (Table 4). Much of the benefit survives recommendation preservation; the intervals still permit an additional full-vector benefit of 1.82 points in BabyAI and 2.03 in expert-reference CrafText.

Changed donors and later visitation prevent additive causal decomposition of these gaps (Appendix G).

5.3 Which reference should supply guidance?

Direct Qwen labels. To test label source, Qwen and expert hard-label fits share positions and budgets. Qwen uses no history; the expert is recurrent, so information is not matched. Qwen Aligned is 1.26 points below expert Aligned in primary AUC (Table 5), despite a $+1.62$-point endpoint difference ($[.04, 3.20]$, descriptive 95%). Its assignment benefit is uncertain; Qwen Aligned also averages below Anchor only (-0.5 points, $[-2.4, +1.3]$, 95%). To test example selection instead of label replacement, upweighting Qwen–expert disagreements while retaining expert targets changed AUC by -0.08 points relative to expert-only Aligned ($[-2.18, +2.01]$, 97.5%); its Aligned-minus-calibrated-Reassigned comparison also did not establish a benefit (Appendix N).

Expert–Qwen mixture. Probability mixing tests whether Qwen adds information beyond the expert or merely softens its targets. AUC is .30183 for the mixture, .30742 for expert and .31020 for expert–Uniform. The mixture misses the minimum gains fixed before the runs: 1.0 point over expert and .5 over expert–Uniform. With 99.1667% intervals for six primary comparisons, mixture minus

Table 3: CrafText: AUC fractions and Aligned-minus-control percentage points. Studies share five upstream pipelines but train separate Aligned policies. Reassigned/Anchor contrasts are primary; Uniform is secondary. All intervals are 95%; all five pipeline means are positive. Bold differences have intervals excluding zero.

Study / control	Aligned	Control	Difference	95% interval
Assignment: calibrated Reassigned	.29232	.27478	$+\mathbf{1.75}$	$[+0.62, +2.89]$
Reference-control: Anchor only	.29556	.27799	$+\mathbf{1.76}$	$[+0.43, +3.09]$
Reference-control: Uniform	.29556	.26500	$+\mathbf{3.06}$	$[+0.79, +5.32]$

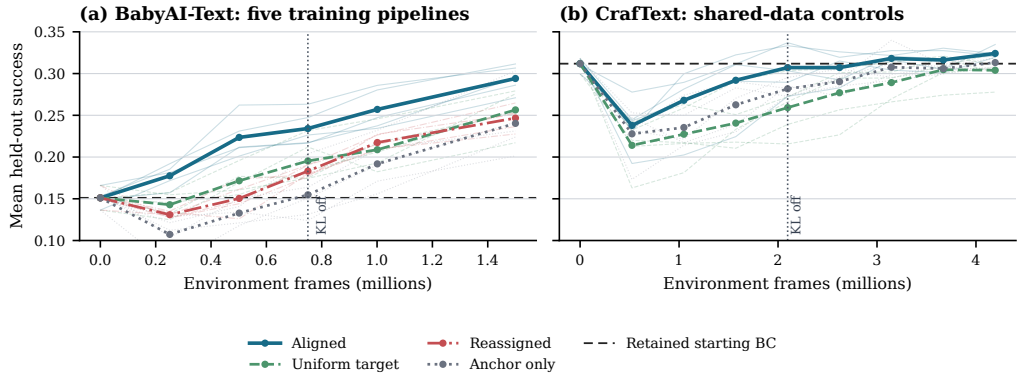


Figure 2: **Held-out learning curves.** Thick lines average five pipelines; faint lines show each pipeline. CrafText uses the reference-control study's full-position fits. Horizontal lines show retained BC; vertical lines mark KL cutoff. Vertical axes are success fractions; horizontal axes count environment frames. Curves are unsmoothed.

Table 4: **Own-state, unrestricted and same-top guidance.** AUC fractions; Aligned-minus-control gaps in points. BabyAI rows share Aligned. Five-pipeline intervals: descriptive 95% in row one, $98\frac{1}{3}\%$ Bonferroni for three primary contrasts. Bold: interval excludes zero. Controls share eligible rows, without calibration. CrafText reuses Table 5's hard-label references, not policies.

Setting / reassignment	Aligned	Control	Gap	Interval
BabyAI / unrestricted	.22247	.17117	$+\mathbf{5.13}$	$[+3.21, +7.06]$
BabyAI / same top	.22247	.21530	$+0.72$	$[-0.39, +1.82]$
CrafText, Qwen / same top	.26996	.26957	$+0.04$	$[-0.61, +0.69]$
CrafText, expert / same top	.28853	.28747	$+0.11$	$[-1.82, +2.03]$

Table 5: **Qwen versus expert labels at matched positions.** Five reused pipelines, three new paired seeds. Qwen Aligned AUC .27056; comparator fractions and Qwen-minus-comparator points. Two-test Bonferroni intervals (97.5%). Reassignment is unrestricted and calibrated, unlike Table 4. Bold: interval excludes zero, in either direction.

Comparator	Comparator AUC	Difference	Adjusted interval
Expert Aligned	.28320	$-\mathbf{1.26}$	$[-1.75, -0.77]$
Qwen calibrated Reassigned	.26536	$+0.52$	$[-0.08, +1.12]$

expert is -0.56 points $[-1.62, +0.51]$ and minus expert–Uniform -0.84 $[-2.10, +0.42]$. Both upper bounds fall below those minimum gains (Appendix I).

Using the initializer as reference. On CrafText's mixture-study cohort, the Frozen-BC reference beats sparse expert, sparse expert–Qwen mixture and Anchor only in all five pipeline means (Table 6).

Table 6: **CrafText mixture-study cohort.** Five pipelines, three paired seeds; 15 new/45 reused policies. Frozen-BC-minus-comparator AUC points; $98\frac{1}{3}\%$ intervals. Full-data BC versus sparse fits compares recipes, not label source alone. Bold: interval excludes zero.

PPO recipe	AUC ↑	Frozen-BC minus row	Interval
Frozen-BC reference	.32432	–	–
Sparse expert reference	.30742	$\mathbf{+1.69}$	$[+0.48, +2.90]$
Sparse expert–Qwen mixture	.30183	$\mathbf{+2.25}$	$[+1.10, +3.40]$
Anchor only	.30429	$\mathbf{+2.00}$	$[+0.24, +3.76]$

Designed after the mixture results, this study omits full-position expert and reassigned Frozen-BC controls. All four recipes exceed retained BC (.29123); Frozen-BC gains 3.31 points (descriptive; Appendix J). All four zero the critic's output layer, so its initial return predictions are zero; main studies copy it unchanged. Cross-cohort scores cannot isolate reference choice.

5.4 Boundaries: initialization and other environments

BC fitting budget. Does weaker initialization increase the assignment benefit? One- versus ten-epoch BC on five pipelines, with fixed Qwen references and anchoring, does not establish this. Weak-BC Aligned AUC (.0475) exceeds its initializer (.0254) but stays below retained full BC (.3097) (Appendix K).

Other suites. Assignment contrasts are small or mixed in HighwayEnv (return), MiniHack (success) and object-centric Pong (return) (Leurent, 2018; Samvelyan et al., 2021; Küttler et al., 2020; Delfosse et al., 2024). Pong's benefit shrinks with more BC updates. Zeroing MiniHack's value head improves mean AUC, but Aligned stays below retained BC. A HighwayEnv expansion stopped after its expert failed the performance gate (Appendix O.1).

6 Discussion

Anchoring improves learning under corrupted guide inputs. Separately, own-state targets outperform exchange, and preserving the top recommendation reduces BabyAI's gap. Guidance content and state assignment both matter.

Frozen-BC guidance beats sparse CrafText fits; its comparison with BabyAI's fitted reference remains unresolved. Relative to retained BC, all four mixture-cohort recipes improve AUC, whereas fitted-reference AUC falls in CrafText's reference-control, direct-label, top-action and sampling-priority studies. Endpoint improvements need not improve time-averaged learning.

7 Limitations

Privileged labels, symbolic inputs, discrete actions and one selected CrafText expert configuration limit generalization. Teacher memory and fitting differ across sources. Anchoring ablations use one initializer; the clean cohort does not establish reduced corruption sensitivity. Controls do not distinguish teaching better actions from retaining demonstrated behavior. Frozen-BC was not reassigned, leaving the state-specificity of its benefit untested. Fixed CrafText instructions leave language necessity untested.

8 Conclusion

Reference guidance with demonstration anchoring trains standalone instruction-following policies, improves BabyAI learning and beats the main CrafText PPO controls. Controlled studies favor own-state targets and recommendation-preserving exchange; CrafText's mixture cohort favors Frozen-BC guidance over sparse fits.

ETHICS STATEMENT

This simulation-only study uses privileged planner or learned-expert labels and no human subjects or private data. Risks include transferring unsafe LLM priors, over-trusting them under shift, and training compute. Evaluation of trained PPO policies calls neither Qwen, the planner, nor the fitted reference.

REPRODUCIBILITY STATEMENT

The supplement provides frozen protocols, code, training logs, saved evaluation records, and a file-level inventory. Appendix A maps the principal studies and reconstruction scopes; Appendix O covers auxiliary and development procedures. Appendices G, H, I, K and J report these five follow-up studies, including failed confirmations and reused comparators.

The full supplement includes the hash-listed training payloads and fitting annotations omitted from the compact download. Anonymous reviewer access: `https://osf.io/d8ayv/overview?view_only=7b32a0b5f0fd4128a48f9858c4a8a7ed`. The compact package contains the current manuscript; the full archive retains its earlier manuscript snapshot with unchanged experimental records. The five-study verifier reconstructs the five studies above; the synchronized verifier additionally checks current online-guidance results and MiniHack follow-ups. Neither reruns training or validates every study. The README states each package's contents, commands and verification scope. For the online-Qwen auxiliary study, replay starts from the preserved mixed demonstration corpus; the earlier collection learner's complete sequence of saved training models is not fully documented (Appendix D.1).

AI USE STATEMENT

Separately from Qwen3-32B, which supplies experimental action priors, reference-fitting priorities and direct action labels, generative-AI assistants including OpenAI Codex and external LLM review systems were used to critique and refine hypotheses, contribution framing, methodology, experimental controls, implementation, debugging, mathematical equivalence checks, result interpretation, manuscript organization and prose. They were not authors or independent sources of empirical evidence. The authors take responsibility for all text, claims, code, experiments and released artifacts; these tools do not replace human verification of results, equations, citations or release materials.

REFERENCES

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization, 2016. URL `https://arxiv.org/abs/1607.06450`.

Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. Grounding Large Language Models in Interactive Environments with Online Reinforcement Learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 3676–3713. PMLR, 23–29 Jul 2023. URL `https://proceedings.mlr.press/v202/carta23a.html`.

Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. BabyAI: First Steps Towards Grounded Language Learning With a Human In the Loop. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=rJeXCo0cYX`.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL `https://aclanthology.org/D14-1179/`.

Quentin Delfosse, Jannis Blüml, Bjarne Gregori, Sebastian Sztwiertnia, and Kristian Kersting. OCAtari: Object-Centric Atari 2600 Reinforcement Learning Environments, 2024. URL https://arxiv.org/abs/2306.08649.

Yuqing Du, Olivia Watkins, Zihan Wang, Cédric Colas, Trevor Darrell, Pieter Abbeel, Abhishek Gupta, and Jacob Andreas. Guiding Pretraining in Reinforcement Learning with Large Language Models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8657–8677. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/du23f.html.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. ISSN 0893-6080. doi: 10.1016/j.neunet.2017.12.012. URL https://www.sciencedirect.com/science/article/pii/S0893608017302976. Special issue on deep reinforcement learning.

Alexandre Galashov, Siddhant Jayakumar, Leonard Hasenclever, Dhruva Tirumala, Jonathan Schwarz, Guillaume Desjardins, Wojtek M. Czarnecki, Yee Whye Teh, Razvan Pascanu, and Nicolas Heess. Information asymmetry in KL-regularized RL. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=S1lqMn05Ym.

William F Guthrie. NIST/SEMATECH e-Handbook of Statistical Methods (NIST Handbook 151), 2020. URL https://www.itl.nist.gov/div898/handbook/prc/section4/prc463.htm.

Todd Hester, Matej Vecerik, Olivier Pietquin, Marc Lanctot, Tom Schaul, Bilal Piot, Dan Horgan, John Quan, Andrew Sendonaris, Ian Osband, et al. Deep q-learning from demonstrations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. URL https://ojs.aaai.org/index.php/AAAI/article/view/11757.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8): 1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

Hengyuan Hu and Dorsa Sadigh. Language Instructed Reinforcement Learning for Human-AI Coordination. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 13584–13598. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/hu23e.html.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 448–456, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/ioffe15.html.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Tomasz Korbak, Ethan Perez, and Christopher Buckley. RL with KL penalties is better viewed as Bayesian inference. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1083–1091, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.77. URL https://aclanthology.org/2022.findings-emnlp.77/.

S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951. ISSN 00034851. URL http://www.jstor.org/stable/2236703.

Heinrich Küttler, Nantas Nardelli, Alexander Miller, Roberta Raileanu, Marco Selvatici, Edward Grefenstette, and Tim Rocktäschel. The NetHack Learning Environment. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 7671–7684. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/569ff987c643b4bedf504efda8f786c2-Paper.pdf.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, pp. 611–626, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400702297. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

Edouard Leurent. An Environment for Autonomous Driving Decision-Making. https://github.com/eleurent/highway-env, 2018.

Junzhuo Liu, Weiwei Li, Jun Ling, and Peng Wang. When Privileged Guidance Misaligns: State-Matched Routing and Contextualized Self-Distillation for Multi-Turn Agents, 2026a. URL https://arxiv.org/abs/2608.05219.

Kuan-Yen Liu, Ren-Jyun Huang, and Ti-Rong Wu. LaGO: Latent Action Guidance for Online Reinforcement Learning, 2026b. URL https://arxiv.org/abs/2606.24669.

Zhiyuan Liu, Tinghong Ye, Chenghao Liu, Yizhuo Li, and Songfang Huang. MemOPD: On-Policy Distillation through Memory State Alignment for Long-Horizon Agents, 2026c. URL https://arxiv.org/abs/2608.07068.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Jelena Luketina, Nantas Nardelli, Gregory Farquhar, Jakob Foerster, Jacob Andreas, Edward Grefenstette, Shimon Whiteson, and Tim Rocktäschel. A Survey of Reinforcement Learning Informed by Natural Language, 2019. URL https://arxiv.org/abs/1906.03926.

Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Thomas Jackson, Samuel Coward, and Jakob Nicolaus Foerster. Craftax: A Lightning-Fast Benchmark for Open-Ended Reinforcement Learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35104–35137. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/matthews24a.html.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Art B. Owen and Dean Eckles. Bootstrapping data arrays of arbitrary order. *The Annals of Applied Statistics*, 6(3):895–927, 2012. doi: 10.1214/12-AOAS547. URL https://doi.org/10.1214/12-AOAS547.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual Reasoning with a General Conditioning Layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. URL https://ojs.aaai.org/index.php/AAAI/article/view/11671.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.

Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations. In *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018. doi: 10.15607/RSS.2018.XIV.049.

Ram Ramrakhya, Dhruv Batra, Erik Wijmans, and Abhishek Das. PIRLNav: Pretraining with Imitation and RL Finetuning for OBJECTNAV. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17896–17906, 2023. doi: 10.1109/CVPR52729.2023.01716.

Finn Rietz, Pedro Zuidberg Dos Martires, and Johannes A. Stork. APC-RL: Exceeding data-driven behavior priors with adaptive policy composition. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 125236–125263, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/cb22704ebc9bc4e0ef875d1959651012-Paper-Conference.pdf.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/ross11a.html.

Tim G. J. Rudner, Cong Lu, Michael A Osborne, Yarin Gal, and Yee Teh. On pathologies in kl-regularized reinforcement learning from expert demonstrations. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 28376–28389. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/eecca5b6365d9607ee5a9d336962c534-Paper.pdf.

Andrei A. Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy Distillation, 2016. URL https://arxiv.org/abs/1511.06295.

Mikayel Samvelyan, Robert Kirk, Vitaly Kurin, Jack Parker-Holder, Minqi Jiang, Eric Hambro, Fabio Petroni, Heinrich Kuttler, Edward Grefenstette, and Tim Rocktäschel. MiniHack the Planet: A Sandbox for Open-Ended Reinforcement Learning Research. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. URL https://openreview.net/forum?id=skFwlyefkWJ.

Sandra Sattolo. An algorithm to generate a random cyclic permutation. *Information Processing Letters*, 22(6):315–317, 1986. ISSN 0020-0190. doi: 10.1016/0020-0190(86)90073-6. URL https://www.sciencedirect.com/science/article/pii/0020019086900736.

Simon Schmitt, Jonathan J. Hudson, Augustin Zidek, Simon Osindero, Carl Doersch, Wojciech M. Czarnecki, Joel Z. Leibo, Heinrich Kuttler, Andrew Zisserman, Karen Simonyan, and S. M. Ali Eslami. Kickstarting Deep Reinforcement Learning, 2018. URL https://arxiv.org/abs/1803.03835.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-Dimensional Continuous Control Using Generalized Advantage Estimation. In Yoshua Bengio and Yann LeCun (eds.), *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1506.02438.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Samyak Shrestha and Alexander Tessier. Rethinking Privileged Information in On-Policy Self-Distillation, 2026. URL https://arxiv.org/abs/2608.18271.

Avi Singh, Huihan Liu, Gaoyue Zhou, Albert Yu, Nicholas Rhinehart, and Sergey Levine. Parrot: Data-Driven Behavioral Priors for Reinforcement Learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Ysuv-WOFeKR.

Giacomo Spigler. Proximal Policy Distillation. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=WfVXe88oMh.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. URL https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Szegedy_Rethinking_the_Inception_CVPR_2016_paper.html.

Dhruva Tirumala, Alexandre Galashov, Hyeonwoo Noh, Leonard Hasenclever, Razvan Pascanu, Jonathan Schwarz, Guillaume Desjardins, Wojciech Marian Czarnecki, Arun Ahuja, Yee Whye Teh, and Nicolas Heess. Behavior Priors for Efficient Reinforcement Learning. *Journal of Machine Learning Research*, 23(221):1–68, 2022. URL http://jmlr.org/papers/v23/20-1038.html.

Zoya Volovikova, Gregory Gorbov, Petr Kuderov, Aleksandr Panov, and Alexey Skrynnik. CrafText Benchmark: Advancing Instruction Following in Complex Multimodal Open-Ended World. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26131–26151, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1267. URL https://aclanthology.org/2025.acl-long.1267/.

Andrew Wagenmaker, Perry Dong, Raymond Tsao, Chelsea Finn, and Sergey Levine. Posterior Behavioral Cloning: Pretraining BC Policies for Efficient RL Finetuning, 2025. URL https://arxiv.org/abs/2512.16911.

Yuxin Wu and Kaiming He. Group Normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

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, 2025. URL https://arxiv.org/abs/2505.09388.

Zihao Zhou, Bin Hu, Chenyang Zhao, Pu Zhang, and Bin Liu. Large Language Model as a Policy Teacher for Training Reinforcement Learning Agents. In Kate Larson (ed.), *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 5671–5679. International Joint Conferences on Artificial Intelligence Organization, 8 2024. doi: 10.24963/ijcai.2024/627. URL https://doi.org/10.24963/ijcai.2024/627. Main Track.

A Reading Map and Operational Glossary

This appendix expands the main paper's procedures; the following appendices give current protocols and results before secondary and development evidence. Table 7 maps the principal studies. Appendix O.1 gives the auxiliary-suite protocols, including MiniHack critic initialization and Pong's BC factorial; Appendix O.2 gives development studies. Saved records, reconstruction programs and payload inventories are supplied separately.

Table 7: Evidence and verification map. A pipeline is an upstream data/BC/reference construction, not a PPO seed. CrafText tables call its expert/data origin a *source*; saved record identifiers use *lineage*. These names do not add independent replications. The supplement's README maps studies to saved records and reconstruction commands.

Study and appendix	Replication and reuse	Evidence location / entry point
Main BabyAI (E)	Five constructed pipelines, four paired PPO seeds; four main conditions	Assignment, reference controls and robustness
Online guidance (D.1)	Five PPO seeds per recipe; separate data and prompting configuration	Guidance sources and anchoring, Table 1
Main CrafText (F)	Five pipelines, three seeds; selected configuration and reference-control study	Assignment and reference-control studies
Qwen sampling (N)	60 new policies on five reused sources; two matched reference fits per source	Two primary tests; expert labels retained
Additional BabyAI (L)	Two new 20-policy arms; main-study Aligned reused	Random donors and Frozen-BC reference, two 97.5% tests
Top-action study (G)	165 new policies on reused sources	Three primary tests
Direct labels (H)	90 new policies, five reused sources	Two primary tests
Mixture confirmation (I)	105 policies, four reused sources and one new; development excluded	Six primary tests
BC-budget confirmation (K)	60 policies, five reused sources; development excluded	Three primary tests
Frozen-BC reference (J)	15 new policies, 45 designated comparators reused	Three primary tests

The five-study verifier executes the reconstruction for the last five studies in Table 7: 435 unique policy records and 480 policy-record uses. Reusing 45 comparator records does not create new policies; neither quantity is the inferential sample size. Intervals in these five studies use five source means. The synchronized-release verifier also checks paper-file synchronization, current online-guidance results and MiniHack follow-ups. These checks reconstruct saved results, not training or evaluation. The supplement's README identifies each edition's payloads and check scope.

A.1 What is fitted, trained and deployed?

Table 8 distinguishes the main training and evaluation objects.

Time and data units. A training *frame* is one environment step. A *rollout* contains steps collected before one PPO update; a row indexes one step in one parallel environment. An *epoch* is a pass through a specified training set or collected rollout, not another environment rollout. With replacement sampling, an epoch need not visit every example once. A *checkpoint* saves model parameters at a training step. Evaluation *blocks* are sets of episode seeds shared by compared policies within a study. New blocks need not contain new instructions. A *cohort* is the specified collection of policy runs and evaluation blocks, not a new domain.

1. **BC initialization.** Behavior cloning fits action predictions on labelled examples. Its saved parameters initialize the trainable PPO policy. An initialization checkpoint is a saved model, not another source of labels.

Table 8: Quick lookup. Operational steps and mathematical definitions follow below.

Term	Meaning and stage
Aligned	Reference output computed for the learner's current rollout state.
Anchor	Demonstration-action loss applied throughout PPO, not at deployment.
Assignment study	Compares Aligned with reassigned targets; CrafText's primary control uses online calibration (Section 4).
Auxiliary suites	HighwayEnv, MiniHack and object-centric Pong; additional scope checks, not language-effect tests (Appendix O.1).
Block	Episode-seed set shared by compared policies during evaluation.
Checkpoint	Model parameters saved at a specified training step.
Cohort	One study's trained policies and evaluation blocks.
Donor / recipient	Rollout row supplying / receiving a reference vector.
Eligible row (top-action study)	Has another row with the same instruction and top action.
Frozen-BC reference	A frozen copy of the initializer supplies PPO targets.
Hard / soft label	One selected action / a distribution used during reference fitting.
Mixture	Expert and Qwen probabilities averaged before reference fitting.
Multiset	Collection retaining each vector's number of occurrences.
Online combination	Qwen scores added to reference logits during PPO.
Own-state	Aligned guidance: the recipient uses its own reference vector, not a donor's (Section 3).
Pipeline / source	Upstream dataset and fitted models; CrafText identifies it by expert/data origin.
Reference-control study	Separately trained CrafText Aligned, Anchor-only and Uniform policies sharing upstream data and BC (Section 4).
Retained BC	Initializer deployed directly, without PPO.
Selected configuration	CrafText expert recipe chosen in preliminary comparisons, followed by a three-pipeline pilot and fixed five-pipeline studies (Appendix F).
Sparse fitting	Only specified positions in each recurrent window enter the fitting loss.
Uniform target	Action-uniform distribution under the reference-loss schedule.

2. **Separately fitted reference.** Another model fits planner, expert, Qwen or mixture targets. Its parameters are frozen before PPO. A frozen recurrent model still updates its own memory as observations arrive.
3. **Reference KL.** On learner rollout states, the reference supplies a categorical action distribution q. PPO minimizes $\lambda_u \mathrm{KL}(\pi_\theta \| q)$ in addition to its RL objective. λ_u weights this loss; ρ_u changes target sharpness. Increasing one is not the same intervention as increasing the other.
4. **Demonstration anchor.** A continuing labelled-action loss limits drift on the supervised data. BabyAI uses static rows with $h = 0$ (Eq. 4); CrafText uses 64-step recurrent windows, starting memory at zero and applying the recorded episode resets. The anchor weights are .2 and .01, respectively.
5. **Standalone deployment.** Only the trained policy acts at evaluation. The reference, external labeler and anchor dataset are not queried. Annealing reference KL to zero during training does not imply that the anchor also stopped.

A.2 Assignment controls, step by step

For a collected rollout, compute each row's reference vector before reassigning it. The row supplying a vector is its *donor*; the row receiving it is its *recipient*. A row records one step in one parallel environment. Group by the exact encoded instruction, not a broad task family. A cycle assigns each row the next donor in an ordering and wraps the last row to the first. Consequently each vector is donated exactly once. A singleton retains itself. Uniformly randomizing the ordering gives a uniform random *single cycle*: following successive assignments visits every row in that group before returning to the start. This differs from drawing among all permutations without self-assignments, which can contain several cycles.

In the main BabyAI study the order is stored rollout order; in the CrafText assignment study it is a seeded random cycle. The BabyAI random-donor arm changes the ordering but not the within-

instruction vector multiset. Targets remain fixed through the PPO epochs for that rollout. Identical multisets do not imply equal per-state losses, gradients or subsequent states.

For action $a \in \mathcal{A}$, the instruction group's average target probability is

$$\bar{q}_{u,m}(a) = \frac{1}{|G_{u,m}|} \sum_{i \in G_{u,m}} q_{u,i}^{\text{align}}(a) = \frac{1}{|G_{u,m}|} \sum_{i \in G_{u,m}} q_{u,i}^{\text{reassign}}(a).$$

Here $G_{u,m}$ is the nonempty set of rows with instruction m in rollout u; the average weights those rows equally. This equality follows because every vector is used once. It makes one preserved instruction-level tendency explicit: how much probability the reference assigns to each action across these rows. The experiment supplies each recipient a donor's complete vector, not $\bar{q}_{u,m}$. The equality is within a collected rollout, not between the different rollouts later visited by separately trained policies.

For the same-top-action study:

1. Compute $c_i = \arg\max_a z_{u,i,a}$: among action indices a, select the one with the largest reference score for row i at update u, breaking ties by the lowest action index. This uses logits before sharpness scaling; when $\rho_u = 0$, the supplied probabilities are uniform. Mark a recipient eligible only if another row shares both its instruction and c_i. In $\mathcal{E}$, $\exists j \neq i$ means that such a different row j exists. The set contains precisely the rows allowed to exchange vectors.
2. Use exactly these eligible recipients in both donor arms. The unrestricted arm cycles within instructions; the constrained arm cycles within instruction–top-action groups. Ineligible rows retain self.
3. Exchange the complete vectors. For example, $(.60, .30, .10)$ and $(.50, .49, .01)$ have the same top action but different soft constraints. They are illustrative vectors, not observations from an experiment.
4. Preserve the original row's top action only in the constrained arm. Both controls preserve the within-instruction collection of targets.
5. Compare separately trained policies on the specified evaluation blocks to estimate the additional benefit of the own-state vector. Section 5.2 interprets the uncertainty of this comparison.

Changed logits need not imply changed probabilities: a common additive logit shift disappears under softmax. The activity audit therefore does not establish matched probability perturbations or gradients.

A.3 Reference construction controls

Labels versus predictions. A *hard label* supplies one action; a *soft label* supplies probabilities over all actions. A model fitted to hard labels still predicts a probability vector during PPO. Logits are unnormalized scores: $\text{softmax}(z)_a = \exp(z_a)/\sum_b \exp(z_b)$. Here a, b index action coordinates. In contrast to supervised fitting, the policy-to-reference loss is $\text{KL}(\pi \| q) = \sum_a \pi(a) \log[\pi(a)/q(a)]$. An expert label is a teacher prediction, not a guarantee that the action is optimal. *Expert-visited* means the original expert generated the trajectory; PPO later visits its own states.

Direct-label fitting pairs expert and Qwen hard actions on identical labelled positions and uses the same fitting program. It does not match teacher information: stateless Qwen annotations and a recurrent expert have different context. Expert trajectories still supply coverage, BC and anchoring.

For mixture fitting, compute the four targets in Eq. 6 at each fitting position. Expert–Uniform tests whether mixing merely softens the expert probabilities. For the component-reassigned condition, first permute only the Qwen distributions within the same instruction and train/validation split, then mix with the recipient's original expert distribution. This retains the Qwen collection but generally changes the collection of resulting mixtures. Online target reassignment is applied later to a frozen fitted reference on PPO states.

Frozen-BC reference versus retained BC. Make an exact frozen copy of the actual starting policy and query its native probabilities at temperature one. Keep its recurrent memory separate from the PPO policy's memory and reset both at the appropriate episode boundaries. A new policy

receives PPO updates; the frozen copy does not. In contrast, retained BC is the unchanged initializer evaluated without PPO. The CrafText Frozen-BC-reference study uses full-position BC fitting, whereas designated expert and mixture references used four positions per 64-step sequence. This difference is part of the compared recipe, not a matched source-information experiment.

A.4 Calibration and inference

Calibration changes a loss coefficient, not the reference's action coordinates. If the two mean KL values at the scoring policy are .12 and .20, an unguarded ratio $\kappa = .12/.20 = .6$ makes their coefficient-weighted values agree at that point. Detached means that differentiation treats the multiplier as a constant, rather than changing it through the policy gradient. In CrafText it is fixed during optimization on the same scored rollout; after parameters change the two means need not remain in that ratio. Appendix M specifies numerical guards.

A.5 Evaluation success, learning AUC and aggregation

Learning AUC measures time-averaged success, not area under a receiver operating characteristic (ROC) curve. For one policy, let $Y_{j,b,g,e} \in \{0, 1\}$ indicate episode success at checkpoint j, evaluation block b, task family or goal g, and episode e. BabyAI uses the terminal information field `success` when present, otherwise the indicator of positive terminal reward. The mission verifier supplies that reward on completion. The environment horizon is $8^2 n_{\text{rooms}} n_{\text{nav}}$ steps: n_{rooms} is two for open and one otherwise; n_{nav} is two for putnext and the two-step sequence and one otherwise. Thus goto and pickup have 64 steps, and open, putnext and seq have 128. Evaluation stops at termination or truncation, without an additional collector cap. CrafText records whether `instruction_done` becomes true on any active transition before termination or the 1,000-step evaluation limit. A timeout without instruction completion is a failure; completion on the last permitted transition still counts. These are binary task outcomes, not thresholds on cumulative return. With G equally weighted task groups, B_{eval} blocks and N_g episodes per group in each block, compute

$$s_{j,b} = \frac{1}{G} \sum_{g=1}^{G} \frac{1}{N_g} \sum_{e=1}^{N_g} Y_{j,b,g,e}, \qquad s_j = \frac{1}{B_{\text{eval}}} \sum_{b=1}^{B_{\text{eval}}} s_{j,b}.$$

At checkpoints $0 = f_0 < \cdots < f_J$ actual training frames, use these $s_j \in [0, 1]$ in the frame-normalized trapezoidal sum

$$\text{AUC} = \frac{1}{f_J} \sum_{j=1}^{J} (f_j - f_{j-1}) \frac{s_{j-1} + s_j}{2}. \tag{7}$$

Here J is the number of intervals, and f_J is the final training frame. The formula linearly interpolates between scheduled evaluations and includes the initializer at $f_0 = 0$. The main success studies have $\text{AUC} \in [0, 1]$; unequal checkpoint spacing receives unequal frame weight. For illustration only, success $.2, .6, .8$ at $0, .25, 1$ million frames gives $\text{AUC} = .25(.2 + .6)/2 + .75(.6 + .8)/2 = .625$. This is not an experimental trajectory or the unweighted mean of three scores. One percentage point is .01 of normalized success; endpoint success is the last score, .8 in this example. Return-based auxiliary suites substitute their stated return metric and normalization, not a success probability.

For two conditions C, D with R paired policy seeds in source ℓ, let $A^C_{\ell,r}$ be seed r's normalized AUC after block averaging. The source effect is $d_\ell = R^{-1} \sum_{r=1}^{R} (A^C_{\ell,r} - A^D_{\ell,r})$. The same linear averaging gives the same result if block-level AUCs are averaged first, provided their checkpoint grids agree. No test checkpoint or favorable episode subset is selected. A missing scheduled evaluation requires the study's stated technical recovery, not silently dropping it. A study with K designated primary comparisons uses the protocol's $100(1 - .05/K)\%$ two-sided intervals. These study-wise rules do not provide error control across the outcome-informed sequence of studies. Original contrast-specific 95% intervals are not retroactively adjusted. The *crossed bootstrap* (Owen & Eckles, 2012) assesses sensitivity to pipeline, paired-seed and evaluation-block sampling. For the main BabyAI analysis, each of 100,000 draws samples four blocks and five pipelines with replacement. For each selected pipeline occurrence it independently samples four paired seed indices with replacement, averages the paired effects over those seeds and the shared sampled blocks, then averages the five pipeline values.

The 2.5th and 97.5th percentiles of these draw means, using linear interpolation between ordered values, form the interval; the random seed is 20260902. It does not generate new policies or replace the governing source-level interval.

From runs to an interval. With n sources, mean $\bar{d} = n^{-1} \sum_{\ell=1}^{n} d_\ell$ and sample standard deviation $s_d = [\sum_{\ell=1}^{n} (d_\ell - \bar{d})^2/(n-1)]^{1/2}$, the interval for one of K primary comparisons is

$$\bar{d} \pm t_{1-.05/(2K),\, n-1}\, s_d/\sqrt{n}.$$

The main studies use $n = 5$. The t subscript specifies a quantile and degrees of freedom. For an unadjusted 95% interval use $K = 1$; Bonferroni adjustment allocates the study's .05 error budget across K comparisons (Guthrie, 2020); the normal-source-mean model is an assumption, not established by five observations. A *macro* score weights task families or canonical goals equally. Frame-normalized AUC averages this score over training time, including initialization, rather than counting checkpoints as independent runs. Appendix tables retain fractions unless their captions say percentage points. Descriptive mean $\pm$ sample standard deviation across policy seeds uses the same mean and $n-1$ variance denominator on seed scores, not the standard error $s/\sqrt{n}$ or a source-level confidence interval.

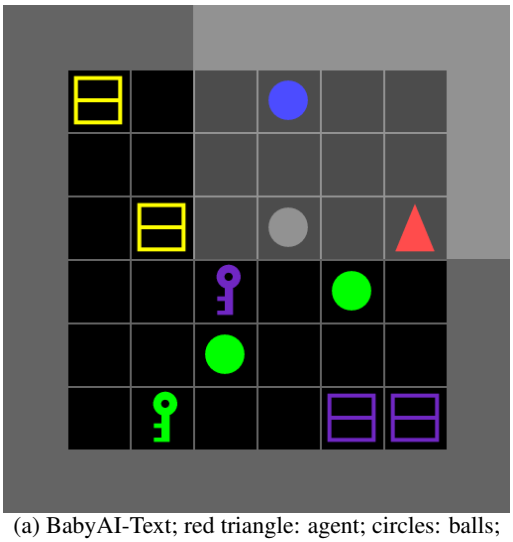
Why Tables 4 and 5 differ. The direct-label study first fits paired expert and Qwen hard-label references, then trains new PPO policies with Aligned or unrestricted calibrated Reassigned targets. The top-action study reuses those CrafText reference fits, but trains different policies on new evaluation blocks and removes online calibration. It compares Aligned with donors restricted to the same top action. Its Qwen and expert rows are separate assignment effects, not a Qwen-minus-expert contrast. Neither table measures the effect of toggling calibration while holding all other study choices fixed.

Design status and saved records. A *primary* comparison governs the stated study decision; a *secondary/descriptive* comparison adds context without replacing that rule. *Development* studies select or assess a proposed recipe; *confirmation* tests its specified rule on the designated follow-up policies, not necessarily new upstream sources. *Outcome-informed* means earlier results motivated the study. A *frozen protocol* fixes settings before its new outcomes; unlike frozen model parameters, it is a design decision, not a training operation or independent preregistration. Validity checks test required model, record and evaluation properties before accepting a study as complete. A reconstruction program recomputes statistics from saved results, not new training runs. The supplement maps immutable code and configuration identifiers to the scientific names used here.

B Environment and Policy Interfaces

Figure 3 shows the two instruction-following environments; Table 9 specifies inputs and action spaces. Figure 4 shows the auxiliary suites.

BabyAI split and action coordinates. Training uses `BabyAI-MixedTrainLocal-v0`; evaluation uses `BabyAI-MixedTestLocal-v0`. Held-out lexical combinations are yellow box, red key, red door, green ball, and grey door. Open uses a two-room locked-door layout; the other families use one room with distractors. Action coordinates are `turn_left`, `turn_right`, `go_forward`, `pick_up`, `drop`, and `toggle`.



(a) BabyAI-Text; red triangle: agent; circles: balls; outlined rectangles: boxes; key shapes: keys; mission: "put a green ball next to a yellow box."



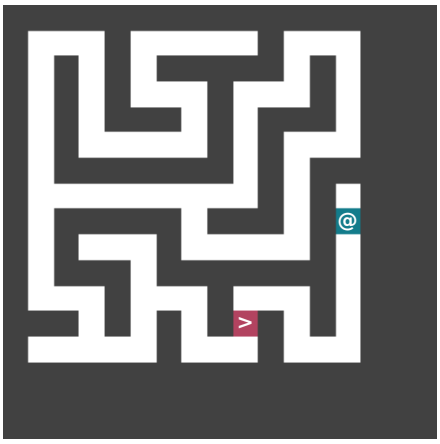
(b) CrafText / Craftax-Classic; the agent stands right of a tree and the inventory appears below; goal: "Collect wood."

Figure 3: **BabyAI and CrafText environments and policy inputs.** These are actual simulator renders generated from the released environment versions. (a) The highlighted BabyAI field is the agent's 7×7 partial view; the policy receives its symbolic encoding and mission tokens, while the planner labeler can access full simulator state. The action space has six controls. (b) CrafText policies and references receive a 1,345-dimensional symbolic Craftax-Classic state and canonical instruction tokens rather than pixels; the action space has 17 controls. In both suites, privileged labelers and frozen references are absent from policy-only evaluation.

Table 9: Environment interfaces. MiniHack's assignment and initialization studies use binary-map model inputs derived from the environment state. The rendered screen is contextual only: every policy and fitted reference receives the structured input listed here rather than screenshot pixels.

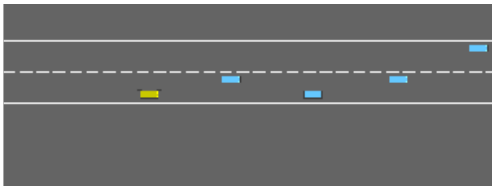
Suite	Policy/reference input	Reassignment group	Actions	Metric
BabyAI-Text	7×7×3 symbolic partial view and mission tokens; policy also carries memory	Exact mission within rollout	6	Success
CrafText	1,345-dimensional symbolic state and canonical instruction; policy and reference carry memory	Exact instruction within rollout	17	Success
HighwayEnv	Ten vehicle slots × seven normalized kinematic features, plus nine all-valid action bits (79 values)	Whole rollout	9	Return
MiniHack	Four 17×17 binary maps (reachable, blocked, player, goal) and task identity	Exact task within rollout	8	Success
OCAtari Pong	Four-frame history of player, ball, and opponent presence/position	Whole rollout	6	Return

(a) MiniHack: actual policy map



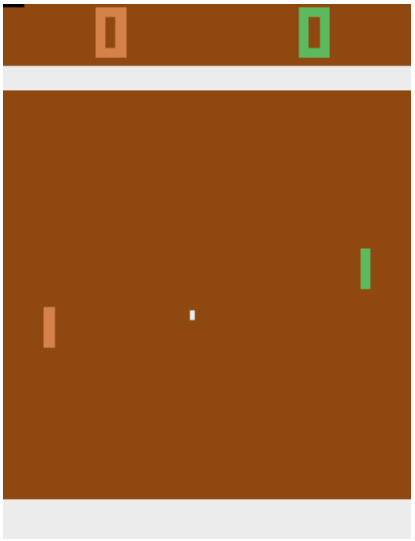
White: reachable; gray: blocked/padding
@: player; >: goal

(b) HighwayEnv: simulator screen



Road scene; policy receives kinematics

(c) Pong: emulator screen



Policy receives object positions,
not pixels or the displayed score

Figure 4: **Recorded interfaces for the other three suites.** These fixed examples are illustrations, not selected successes or evaluation episodes. (a) MiniHack is a color rendering of the exact binary-map policy input at reset in the larger mapped maze, not a screenshot or the underlying glyph input. Reachable cells and their complement form two channels; player and goal locations form the other two. The model also receives task identity. (b) HighwayEnv's native screen at reset shows the road and vehicles. Its policy receives normalized vehicle kinematics and action-availability bits, not pixels. (c) Pong's native emulator screen after 16 fixed no-operation decisions shows the paddles and ball. Its policy receives four frames of object presence and position, not the displayed score. The rendering seed, fixed actions, observed arrays, source hashes, and software versions are retained in the artifact.

Reading the structured observations. In MiniHack, first locate the player and down-stair goal, then inspect connected walkable cells; eight compass actions move the player toward the goal. The recovery labeler chooses a cardinal shortest-path step on this map, which is not necessarily optimal over the eight-action interface. In HighwayEnv, ten slots contain presence, position, velocity, and heading cosine/sine. Ego position and velocity are absolute; nearby-vehicle position and velocity are ego-relative. Fixed-range normalization clips these values, so they are not unbounded physical coordinates. Nine actions combine negative, zero, or positive acceleration with negative, zero, or positive steering; these are low-level controls, not discrete lane-change commands. In Pong, the three slots identify player, ball, and opponent. A four-frame position history lets a model infer motion; it does not provide a privileged future trajectory.

C Implementation Details

C.1 End-to-end fitting, initialization and evaluation

This subsection specifies the common training stages and their transfers. Appendices E and F give data collection and evaluation for the main BabyAI and CrafText studies; Table 10 identifies the follow-up changes. BC pretraining fits the policy before interaction, reference fitting produces a separate frozen predictor, and anchoring continues to update the policy during PPO. Sharing demonstrations does not make these three operations interchangeable.

BC objective and model inputs. Let D_{BC} be the training distribution specified below, and let θ_{BC} denote the policy parameters fitted from a new random initialization. BabyAI samples individual labelled observations and minimizes

$$L_{\mathrm{BC}}^{\mathrm{BabyAI}}(\theta_{\mathrm{BC}}) = \mathbb{E}_{(o,m,a^\star)\sim D_{\mathrm{BC}}}[-\log \pi_{\theta_{\mathrm{BC}}}(a^\star \mid o, m, h = 0)]. \tag{8}$$

There is no entropy, value, reference-KL or logit-penalty term in this BC loss. Its optimizer applies weight decay separately. Memory is zero for each sampled row, not propagated between demonstrations. The actor predicts action probabilities; the critic predicts expected discounted return. The actor and shared observation, instruction and recurrent parameters receive action-loss gradients; the critic-specific branch is not supervised by BC.

CrafText BC instead samples 64-step demonstration windows s uniformly from the training streams and minimizes

$$L_{\mathrm{BC}}^{\mathrm{CrafText}}(\theta_{\mathrm{BC}}) = \mathbb{E}_s\left[-\frac{1}{64}\sum_{t=0}^{63}\log \pi_{\theta_{\mathrm{BC}}}(a^\star_{s,t} \mid o_{s,t}, m_{s,t}, h_{s,t})\right]. \tag{9}$$

Here $h_{s,0} = 0$; subsequent memory is computed through the window with the recorded episode-reset flags. All positions supply hard expert-action labels. No value, entropy or reference loss is added, and there is no weight decay. The returned model includes actor, shared and unsupervised critic parameters.

BabyAI fitting and selection. BC and the fitted reference use separately seeded episode-grouped 80/20 splits, replacement samplers and 30-epoch optimizer recipe in Appendix E. Weights are computed on the full labelled pool before splitting, including action-frequency counts; only training indices are eligible for optimizer draws. For training-index set I_{train} of size N_{train} and positive row weights w_i, each draw uses $P(i) = w_i / \sum_{j\in I_{\mathrm{train}}} w_j$ for $i \in I_{\mathrm{train}}$. One epoch is exactly N_{train} independent replacement draws, grouped into batches of 256 with the last partial batch retained. It need not visit every training row. Validation is unweighted and receives no gradient updates; its labels nevertheless contribute to the pre-split frequency counts. Each model is initialized and fitted separately. BC retains the epoch with highest unweighted validation action accuracy; the reference retains the highest validation score $\mathrm{accuracy} + 0.5\,\mathrm{non\text{-}goto\ accuracy}$. Only strict improvement replaces either checkpoint, so ties retain the earlier epoch. All collected families in the main study are non-`goto`, making the reference score proportional to overall accuracy there. Both complete the 30-epoch budget, rather than stopping on held-out policy success. The reference minimizes Eq. 1; its instruction gated recurrent unit (GRU; Cho et al., 2014) encodes the current instruction, not an environment trajectory. Neither model is selected by PPO outcomes. The online-Qwen study changes the labelled pool and sampler as specified in Appendix D.1, but uses these fitting and selection rules.

CrafText full-position fitting and selection. The main corpus has 256 environment streams, partitioned by stream index into the first 192 training, next 32 validation and final 32 test streams. Each contributes 64 saved 64-step windows. Collection samples actions from the frozen expert's softmax distribution, rather than taking its argmax. Expert memory continues across collection windows and resets at episode boundaries; fitting instead starts each window at zero memory. Observations and expert logits are stored in float16 and converted to float32 for supervised fitting. The collector uses the benchmark's instruction-reset procedure, not the expert trainer's failure-weighted reset sampler. BC uses the sampled expert actions in Eq. 9. The separately initialized reference fits the expert probabilities $p^E_{s,t}$ obtained by softmax of its stored logits:

$$L_{\mathrm{ref}}^{\mathrm{CrafText}}(\psi) = -\mathbb{E}_s\left[\frac{1}{64}\sum_{t=0}^{63}\sum_{a\in\mathcal{A}} p^E_{s,t}(a)\log \mathrm{softmax}(r_\psi(y_{s,t}))_a\right]. \tag{10}$$

Reference memory starts at zero per window and follows recorded resets. Both fits shuffle training windows without replacement each epoch, use batches of 64 windows, and run Adam (Kingma & Ba, 2015) at 3×10^{-4} with $(\beta_1, \beta_2) = (.9, .999)$, $\epsilon = 10^{-5}$, $\epsilon_{\text{root}} = 0$ and gradient-norm cap 1. Here β_1, β_2 are Adam's first/second-moment decay factors; ϵ stabilizes the denominator outside its square root and ϵ_{root} is the additive constant inside that root (zero here). There is no BabyAI-style logit penalty. They retain the lowest validation cross-entropy checkpoint, requiring improvement greater than 10^{-6}; fitting ends after ten epochs or three consecutive epochs without such improvement. Test loss and closed-loop success check the selected model; they do not choose its epoch. Follow-up fitting masks and selection rules are explicitly different where indicated in Table 10.

Transfer to PPO and continuing supervision. Each main-study PPO condition loads its source's complete BC model: observation and instruction encoders, recurrent parameters, actor and critic branches, together with the corresponding vocabulary/preprocessor. A new PPO optimizer is created; BC optimizer moments and demonstration-time memory are not transferred. BabyAI also copies batch-normalization running statistics with the model. Its trainable policy uses training mode during rollout collection and PPO, including anchor forwards; the fitted reference uses evaluation mode. The held-out evaluator loads the policy in evaluation mode, using its saved normalization statistics without updating them. Policy and recurrent-reference memories start at zero in the new environments, advance independently during interaction, and reset at episode boundaries. The main studies copy the critic-specific parameters without additional value pretraining. The CrafText mixture, Frozen-BC and BC-budget comparisons instead zero the final scalar value-output weights and bias before PPO, leaving action logits unchanged, as specified in their appendices. The direct-label, top-action and sampling-priority studies instead copy the BC critic unchanged. Thus critic initialization is shared by compared arms within each study but differs across some studies; their outcomes do not isolate the effect of this reset.

PPO updates actor, critic and shared parameters with Eq. 3; it does not update the reference or the data-generating teacher. At each PPO minibatch update (eight per rollout: four epochs of two minibatches), BabyAI samples 256 static anchor rows with replacement from the full labelled pool, using zero memory and Eq. 4. CrafText first selects a fixed pool of 1,024 training windows without replacement, seeded by the PPO seed plus 1,000. Each PPO epoch draws 128 distinct pool windows and partitions them into eight 16-window anchor minibatches. The anchor minimizes Eq. 9 with the current policy parameters, zero initial window memory and recorded resets. It is supervised at all 64 positions, including when a reference was fitted only at sparse positions. Reference KL follows its decaying schedule; the anchor coefficient remains fixed until PPO ends.

Outputs and evaluation. BC fitting returns the policy initializer; reference fitting returns the frozen guidance model. Frozen-BC conditions instead use a separate frozen copy of the initializer as reference. PPO runs to its specified interaction budget and saves the prespecified intermediate and final checkpoints. Main learning-curve results use all scheduled evaluations, not the best test checkpoint; endpoint results use the final policy. Each evaluation starts with zero memory and uses greedy policy actions without the reference, anchor dataset, expert or LLM. Retained BC is the same initializer evaluated without PPO on that study's evaluation blocks. It is not the PPO-only condition, which still updates the BC-initialized actor and critic from environment rewards.

C.2 Policy and fitted-reference architectures

Software and accounting scope. BabyAI uses PyTorch; the released publication-environment record lists PyTorch 2.5.1, Python 3.11.4 and Gym 0.25.1. Its fitting constructors use AdamW's (Loshchilov & Hutter, 2019) default moments $(.9, .999)$ and $\epsilon = 10^{-8}$, not PPO's 10^{-5}. CrafText's recorded stack uses Python 3.11.5, JAX 0.4.28, Flax 0.8.5 and Optax 0.2.1, with Craftax 1.4.3. The Adam moments above follow that Optax implementation. Version records identify the corresponding training environments, not independent attestations of every upstream run. The supplement provides these records and the study-specific source snapshots. A complete comparable allocation total for all main-study collection, expert training, fitting and PPO stages is not available in this release; reported stage-specific costs are not end-to-end totals.

The BabyAI PPO and BC policy use two 128-channel convolutional blocks over the $7 \times 7 \times 3$ symbolic view, a 128-dimensional gated recurrent unit (GRU) mission encoder (Cho et al., 2014),

Table 10: Fitting changes relative to the main recipes above. Each cited appendix gives the data, sampling, complete target loss, budget and evaluation protocol.

Study	Changed fitting operation	Checkpoint / procedure home
Direct Qwen labels	Eight hard-labelled positions per 64-step window; matched expert and Qwen fits	Lowest own-validation loss within ten epochs; Appendix H
Expert–Qwen mixture	Four positions; expert probabilities or an offline probability mixture	Final tenth epoch; Appendix I
BC budget	One uninterrupted BC fit, all positions	Fixed epochs 1 and 10, not validation-selected; Appendix K
Qwen sampling priorities	Expert probability targets retained; sequence sampling changes	Unweighted validation selection; Appendix N
Online Qwen	Mixed planner-labelled pool; Qwen scores affect sampling and online targets, not labels	BabyAI rules above; Appendix D.1
Frozen-BC reference	No additional reference fitting	Frozen initializer, Appendix J

two feature-wise linear modulation (FiLM) controllers (Perez et al., 2018), a 128-dimensional long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997), and separate 64-unit actor and critic heads. The convolutions have kernels 2×2 then 3×3, stride one and padding one; each is followed by batch normalization (Ioffe & Szegedy, 2015), rectified linear units (ReLU), and 2×2 max-pooling with stride two. Each FiLM controller applies two 3×3, stride-one, padding-one convolutions with 128 channels. The first is followed by normalization and ReLU; the second is scaled and shifted channel-wise by separate linear maps of the instruction embedding, then normalized and passed through ReLU. A final 2×2 max-pool produces the 128-dimensional LSTM input. The actor and critic hidden layers use tanh. Instruction tokens are lowercase alphabetic words, zero-padded within a batch; the policy uses the GRU output at the last non-padding token. Symbolic image codes are cast to floating point, not divided by 255 as RGB intensities would be. The fitted reference uses the same two-block convolutional geometry, a 128-dimensional image latent, a 128-dimensional GRU mission encoder, two 256-unit hidden layers, and six output logits. It has no FiLM controllers or temporal LSTM: final max-pooling and a 128-unit ReLU image layer precede concatenation with the mission embedding; the two hidden layers use ReLU. Its GRU takes the final hidden state of the padded token sequence, unlike the policy's last-non-padding output. CrafText instead uses the recurrent 17-action model detailed below; its BC and reference are fitted separately within each pipeline. GRU and LSTM units maintain learned summaries of preceding inputs. The actor predicts actions; the critic predicts return and supports PPO updates, not action selection at deployment.

CrafText BC, reference and PPO model. The recurrent policy/reference model takes a $7 \times 9 \times 21$ symbolic map and 22 scalar observations. Two valid 2×2 convolutions with 32 channels each feed a 128-unit map layer; scalars feed a 64-unit layer. Instructions are lowercased, split into alphanumeric tokens, padded/truncated to 32 tokens, and embedded in 32 dimensions. A masked mean over non-padding tokens feeds a 64-unit text layer. Concatenated map, scalar and text features pass through a 256-unit layer and a 256-unit GRU, reset at episode boundaries. Separate 256-unit actor and critic layers produce 17 logits and one value. Hidden dense and convolutional layers use rectified linear units (ReLU). The actor's output uses orthogonal initialization with scale .01 and the value output scale 1; both biases start at zero. This specifies the student/reference model, not the separately trained data-generating expert. Fitting masks and checkpoint selection differ by study as stated in Appendices H, I and K.

LLM serving terms. Qwen inference uses the vLLM model-serving system (Kwon et al., 2023). Bfloat16 (BF16) is the numerical format used for model computation; two-device tensor parallelism splits model operations across two GPUs. These are execution settings, not additional guidance sources. The study-specific prompt, decoding and annotation procedures are specified in the corresponding appendices.

C.3 IMPLEMENTATION-EXACT RECURRENT PPO AND GENERALIZED-ADVANTAGE ESTIMATION

Let θ_{old} denote the rollout policy, let $\mathbb{E}_t$ denote the empirical mean over sampled rollout positions, and let $\hat{A}_t$ and $\hat{G}_t$ denote the generalized-advantage-estimation (GAE; Schulman et al., 2016) advantage and return targets defined below. $V_\theta(x_t)$ predicts expected discounted return, and $\text{clip}(v, l, h) = \min(\max(v, l), h)$ bounds a value to $[l, h]$. The PPO clipping width ϵ is distinct from optimizer stabilization constants; $c_V, c_H \geq 0$ weight value error and entropy, respectively. The importance ratio compares the updated and rollout policies' probabilities of the recorded action:

$$w_t(\theta) = \exp[\log \pi_\theta(a_t \mid x_t) - \log \pi_{\theta_{\text{old}}}(a_t \mid x_t)]\,. \tag{11}$$

The implementation minimizes the clipped policy loss

$$L_{\text{clip}}(\theta) = -\mathbb{E}_t\Big[\min\Big(w_t(\theta)\hat{A}_t, \text{clip}(w_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\Big)\Big]\,. \tag{12}$$

The recorded a_t was sampled from the rollout policy. Its log probability, value estimate and the targets below are held fixed through PPO epochs. Its clipped value estimate and value loss are

$$V_t^{\text{clip}} = V_{\theta_{\text{old}}}(x_t) + \text{clip}(V_\theta(x_t) - V_{\theta_{\text{old}}}(x_t), -\epsilon, \epsilon)\,, \tag{13}$$

$$L_V(\theta) = \mathbb{E}_t\Big[\max\Big((V_\theta(x_t) - \hat{G}_t)^2, (V_t^{\text{clip}} - \hat{G}_t)^2\Big)\Big]\,. \tag{14}$$

Thus the base loss used in Eq. 3 is

$$L_{\text{PPO}} = L_{\text{clip}} + c_V L_V - c_H \mathbb{E}_t[H(\pi_\theta(\cdot \mid x_t))]\,. \tag{15}$$

Let d_{t+1} be zero when the environment reports episode completion and one otherwise. BabyAI's completion flag includes its environment step limit; neither principal trainer adds a separate time-limit bootstrap correction. The mask therefore suppresses bootstrapping from the reset observation. For the scaled environment reward $\bar{r}_t$, generalized-advantage estimation (GAE) (Schulman et al., 2016) uses discount $\gamma \in [0, 1]$ and trace decay $\lambda_{\text{GAE}} \in [0, 1]$. The latter controls how far future temporal-difference errors δ_t contribute; it is not the reference-loss coefficient λ_u:

$$\delta_t = \bar{r}_t + \gamma d_{t+1} V_{\theta_{\text{old}}}(x_{t+1}) - V_{\theta_{\text{old}}}(x_t), \tag{16}$$

$$\hat{A}_t = \delta_t + \gamma \lambda_{\text{GAE}} d_{t+1} \hat{A}_{t+1}, \tag{17}$$

$$\hat{G}_t = \hat{A}_t + V_{\theta_{\text{old}}}(x_t). \tag{18}$$

The backward recursion starts with zero advantage carry beyond the final collected step. The final temporal-difference residual uses the rollout-end value estimate when $d_{t+1} = 1$; the same mask suppresses this bootstrap for completed episodes. For BabyAI, advantages are used without batch normalization. Rollouts carry recurrent state, and PPO minibatches preserve contiguous 20-step unrolls.

CrafText uses the same clipped policy loss but standardizes advantages within each minibatch, $(\hat{A}_t - \bar{A})/(\sqrt{N^{-1}\sum_{i=1}^{N}(\hat{A}_i - \bar{A})^2} + 10^{-8})$, where N counts its rollout positions and $\bar{A} = N^{-1}\sum_i \hat{A}_i$. Thus the denominator uses population, not sample, standard deviation; a constant minibatch maps to zero. This changes the policy-surrogate advantages, not the return targets $\hat{G}_t$. CrafText uses $\frac{1}{2}L_V$ as its value loss. Thus its base objective is $L_{\text{clip}} + \frac{1}{2}c_V L_V - c_H \mathbb{E}_t[H]$. It uses unscaled rewards, $\gamma = .99$, $\lambda_{\text{GAE}} = .8$, $\epsilon = .2$, $c_V = .5$, $c_H = .01$, and gradient-norm cap 1. Each 64-step rollout from 1,024 environments is optimized for four epochs, with eight minibatches of 128 whole trajectories; recurrent context begins at the stored rollout-start memory. Adam uses $\epsilon = 10^{-5}$ and a learning rate decreasing linearly from 2×10^{-4} to zero across 2,048 gradient updates (64 rollouts). Its reference coefficient decreases linearly from .05 to zero over the first 32 rollouts, with unit target sharpness; the anchor coefficient is .01. CrafText study-specific donor and calibration changes are specified with their experiments. These settings are not BabyAI's cosine schedules or unnormalized-advantage implementation.

Schedules and numerical reductions. Let f_u be frames completed before collecting BabyAI rollout u, and $c(v) = \min(1, \max(0, v))$. The coefficient and sharpness, held fixed through that rollout's PPO epochs, are

$$\lambda_u = \frac{\lambda_0}{2}[1 + \cos(\pi c(f_u/750000))], \qquad \rho_u = \frac{\rho_{\max}}{2}[1 - \cos(\pi c(f_u/375000))].$$

Table 11: Implementation-exact BabyAI PPO and auxiliary-loss settings. Both reference schedules use cosine interpolation.

Setting	Value	Setting	Value
Discount γ	0.99	GAE λ_{GAE}	0.99
Clip ϵ	0.2	Value coefficient c_V	0.5
Entropy coefficient c_H	0.01	Reward multiplier	20
PPO epochs / rollout	4	Parallel environments	64
Frames / environment	40	Rollout rows	2,560
Recurrent unroll	20	Minibatch rows	1,280
Learning rate	3×10^{-5}	Adam (β_1, β_2)	$(0.9, 0.999)$
Adam ϵ	10^{-5}	Max. gradient norm	0.5
Anchor batch / coefficient	256 / 0.2	Main budget	1.5M frames
Reference λ_u	$0.05 \to 0$ by 0.75M	Reference scale ρ_u	$0 \to 2$ by 0.375M

Here $\lambda_0 = .05$ and $\rho_{\max} = 2$ in the main recipe, while the specified sensitivity and calibrated arms change λ_0. Thus guidance starts with uniform targets, sharpness reaches two at .375M frames, and the coefficient is zero from .75M frames; rollout discretization determines the first update at or beyond each boundary. Frozen-BC uses native probabilities instead of this sharpness schedule. For CrafText, with zero-based rollout index $u = 0, \ldots, 63$, $\lambda_u = .05 \max(1 - u/32, 0)$ and $\rho_u = 1$. At zero-based optimizer step $v = 0, \ldots, 2047$, its learning rate is $2 \times 10^{-4} \max(1 - v/2048, 0)$, reaching zero after the final step. Online calibration multiplies the scheduled coefficient as specified in Appendix M; it does not change target sharpness.

Categorical KL sums over actions, then averages rollout positions. BabyAI evaluates the policy logarithm as $\log(\pi_\theta(a) + 10^{-8})$ and target log probabilities by log-softmax. CrafText uses log-softmax for both distributions. Targets are detached from differentiation; these numerical conventions do not add a reference-training stage. Gradient-norm caps apply to the joint gradient over optimized parameters, before the optimizer step.

For the ideal categorical KL, before the numerical logarithm stabilization above, a six-action uniform target U gives

$$L_{\mathrm{PPO}} + \lambda_u \mathrm{KL}(\pi_\theta \| U) = L_{\mathrm{clip}} + c_V L_V - (c_H + \lambda_u)\mathbb{E}_t[H(\pi_\theta(\cdot \mid x_t))] + \lambda_u \log 6. \tag{19}$$

The final term is constant in θ, so the Uniform-target control has effective entropy coefficient $c_H + \lambda_u$. Table 11 reports all decision-relevant publication values. Adam (Kingma & Ba, 2015) adapts parameter step sizes using moving gradient moments; the fitting optimizer AdamW additionally decouples weight decay (Loshchilov & Hutter, 2019).

C.4 Expert used to construct the CrafText datasets

The data-generating expert is distinct from the student/reference model above. Its instruction encoder averages masked 48-dimensional token embeddings, then applies a 128-unit ReLU layer and layer normalization (Ba et al., 2016), which normalizes each example's feature coordinates. Two 32-channel 2×2 convolutions and a 128-unit ReLU layer encode the spatial map; a 64-unit ReLU layer encodes the remaining scalar observations. Instruction features produce feature-wise linear modulation (FiLM) scales and offsets (Perez et al., 2018) for both branches. Their outputs and the instruction features are concatenated, projected to 256 units and passed through a 256-unit gated recurrent unit (GRU). Separate 256-unit actor and critic heads produce 17 action logits and one value. Episode resets clear recurrent memory.

Expert PPO uses 1,024 environments, 64-step rollouts, four optimization epochs, eight minibatches, discount .99, generalized-advantage decay .8, clipping .2, entropy weight .01, value weight .5 and gradient-norm cap one. It starts from a new random actor–critic initialization and optimizes the CrafText base PPO loss in Appendix C.3, without a reference or anchor. Adam uses $\epsilon = 10^{-5}$. The budget is 5×763 rollout updates (250,019,840 environment steps), each with 4×8 optimizer steps. At zero-based optimizer step v, the learning rate is $2 \times 10^{-4} \max(1 - v/(5 \cdot 763 \cdot 32), 0)$; optimizer state and this counter continue across the five segments, without restarting the decay. The collector loads the final-budget checkpoint, not the best evaluated segment, and freezes its parameters. Collection

starts with zero recurrent memory and resets it at episode boundaries. Sources 4–8 and source 9 use the capped failure-weighted task sampling specified in Appendix I; it changes training reset probabilities, not evaluation's equal goal weighting. The supplement includes the expert architecture, training program, sampling implementation and the three evaluations used to form those probabilities.

C.5 CRAFTEXT QWEN INTERFACE AND EXACT PROMPT

Qwen supplies offline information for reference construction on fixed expert-visited states. The reference-fitting studies do not query Qwen during PPO or evaluation. Table 12 separates these uses from the autonomous-collection pilots and from BabyAI's online interface.

Input construction and information access. The current-observation interface deterministically converts the 1,345 input values into a 7×9 grid of tile and creature names, 12 inventory counts, four survival variables, facing direction, light and sleep state. The player is at row 4, column 5 (one-based). Inventory and survival values are multiplied by ten and rounded to integers; light is formatted to four decimal places. The tile immediately ahead is derived from the same grid and facing fields. The system message supplies public rules and all 17 action codes; the user message supplies the canonical instruction and this description. No expert action, hidden simulator state, reward, future observation or observation history is supplied. There is no BabyAI-style left/right corruption.

Complete system message. The following text is the unchanged system message of the released prompt builder, including its rules and action mapping; line wrapping is for display.

```
You choose the next action in Craftax Classic to follow the instruction. Only the local
observation is available. The player is at the center of the grid. Rows run from up to down
and columns from left to right. Movement also determines facing; do interacts with the tile or
creature in front of the player. Select exactly one action code, without explanation. A=noop
B=left C=right D=up E=down F=do G=sleep H=place_stone I=place_table J=place_furnace
K=place_plant L=make_wood_pickaxe M=make_stone_pickaxe N=make_iron_pickaxe O=make_wood_sword
P=make_stone_sword Q=make_iron_sword Game rules: Moving toward a blocked tile changes facing
but does not harvest it. Use do when facing an adjacent target. Trees yield wood; water
provides a drink; grass can yield a sapling; ripe plants provide food; do attacks creatures.
Mining stone or coal requires a wood_pickaxe; mining iron requires a stone_pickaxe; mining
diamond requires an iron_pickaxe. Placing a table costs 2 wood. Placing a furnace or stone
costs 1 stone. A table or furnace needs a non-solid tile in front. Stone may also be placed on
water. Planting requires 1 sapling and grass in front. Crafting any pickaxe or sword requires
a table in one of the eight adjacent tiles. A wood tool costs 1 wood; a stone tool costs 1
wood and 1 stone; an iron tool costs 1 each of wood, stone, iron and coal, and also requires
an adjacent furnace. Obtain missing prerequisites before attempting an unavailable crafting or
placement action. Do not assume that naming the final goal action makes its prerequisites
available.
```

Recorded user message and output. The following is the first queried position in the first training stream and first saved chunk of CrafText source 4, selected by index rather than outcome. Reconstruction from its recorded observation and instruction reproduces its saved chat-template prompt hash. Its recorded response is code `F` (`do`), with returned selected-token log probability -0.076216. This example illustrates the executed interface, not correctness of the action.

```
Instruction: Don't create an Iron Sword. You need to, however, craft a Stone Sword and gather
↪  a Sapling.
Local grid (7 rows, 9 columns; player at row 4, column 5):
sand sand sand grass grass grass sand sand grass
grass grass grass grass tree grass grass grass tree
grass tree grass grass tree grass grass grass grass
tree grass grass grass grass grass grass tree tree
stone grass grass grass grass grass grass tree grass
stone coal grass grass grass grass grass grass grass
stone iron stone grass grass grass grass tree grass
Inventory: wood=0, stone=0, coal=0, iron=0, diamond=0, sapling=0, wood_pickaxe=0,
↪  stone_pickaxe=0, iron_pickaxe=0, wood_sword=0, stone_sword=0, iron_sword=0
Survival: health=9, food=9, drink=9, energy=9
Facing: up
Tile immediately in front: tree
Light: 0.7969; sleeping: False
Next action code:
```

Decoding, output use and failure handling. The tokenizer's chat template appends the generation marker with thinking disabled. The hard-label interface restricts output to 17 distinct single-token codes, uses temperature zero and generates one token. An overlength prompt (including its output

token) fails the 4,096-token context check rather than being silently truncated. Invalid or incomplete responses and nonfinite scores cause technical failure, not an expert-action fallback. The chosen action is the hard label; its selected-token log probability is diagnostic, not a full action distribution or a sampling weight. The probability-mixture interface instead retains all 17 processed action log probabilities and normalizes their exponentials over the action codes. These are constrained action probabilities, not the model's unrestricted vocabulary distribution. Its numerical checks and execution settings are in Appendix I.

Table 12: CrafText Qwen interfaces. The first four rows share the complete current-observation prompt above; query positions and output use differ. All reference-fitting rows keep expert-derived BC and demonstration anchoring.

Study	Query and output use	Procedure home
Sampling priorities	Eight positions per 64-step window; disagreement with expert argmax sets sequence weights in $[1, 2]$; expert soft targets retained	Appendix N
Direct labels	Same eight positions; selected action becomes the hard fitting label	Appendix H
Probability mixtures	Four positions (0, 16, 32, 48); normalized 17-action probabilities mixed with expert probabilities	Appendix I
BC-budget comparison	Reuses hard-label references; the added source uses the eight-position annotation recipe	Appendix K
Autonomous pilots	Execute Qwen actions; separate variants add eight-step history, observable prerequisites or a two-request reasoning step	Appendix O.2

The interface builder and the example's full chat-template text, input identity, action and matching hash accompany the supplement. The example check does not constitute new model inference or an independent reproduction of every annotation.

Autonomous-pilot prompt extensions. The current-observation pilot uses the messages above unchanged. The history variant prefixes the user message with at most eight previous observations and chosen actions from that episode, oldest first. Each summary retains only the `Inventory:`, `Survival:`, `Facing:` and `Tile immediately in front:` lines of the observation description, joined by semicolons, followed by `chosen action=` and the action name. The history starts empty and is not a plan or expert feedback. The separate prerequisite/reasoning factorial has no history prefix. Its prerequisite variant lists crafting/placement actions by the observable requirements below and adds four movement-destination descriptions; it does not remove actions from Qwen's output vocabulary.

For these checks, "adjacent" means the eight neighboring cells. The excluded placement tiles are water, stone, tree, coal, iron, diamond, crafting table, furnace, plant, ripe plant, invalid and out-of-bounds cells. Destinations must also have no creature. Stone placement additionally permits water; planting requires grass. Inventory requirements and adjacent table/furnace requirements are exactly the public rules in the system message above. The check treats crafting/placement as available while the observation reports sleeping, because its failure masks exclude sleeping states. Movement is marked blocked for the excluded placement tiles, lava or creatures. Its two literal status strings are `open` and `blocked; facing changes but position does not`. These are observable prerequisite diagnostics, not guarantees of successful execution or progress.

The following templates give the exact added wording; braces identify fields, not text sent to Qwen. Lists follow action-code order, and empty lists print `none`. Tile and action names use the same mapping as the base prompt.

```
History prefix (only when prior observations exist):
Past observations and chosen actions, oldest first (not a plan or privileged feedback):
{1: observation summary; chosen action=action_name}
{... up to eight numbered summaries ...}
{unchanged current-observation user message}

Prerequisite block (inserted before "Next action code:"):
Observable prerequisite check (computed only from the shown observation and rules):
Currently possible crafting/placement: {code=action list, or none}.
```

```
Missing prerequisite or occupied destination: {code=action list, or none}.
Movement destinations: {action: tile, status; one entry per direction}.
This is not a plan or a guarantee of progress. Other actions remain available.

Reasoning request: replace the system sentence
Select exactly one action code, without explanation.
with
First reason briefly about the next action. A separate request will ask for its action code.
Replace the user suffix "Next action code:" with
Briefly identify one executable next action toward the instruction. Consider current
↪  prerequisites, nearby obstacles and survival.

Final request: retain the reasoning-request messages, append the generated
reasoning as an assistant message, then append this user message:
Now select the next action. Return exactly one action code and no explanation.
```

Direct decoding greedily emits one permitted action token. Deliberate decoding first greedily generates at most 64 unrestricted reasoning tokens, then uses the final request above to emit one permitted action token. Both requests disable the model's chat-template thinking mode. A 4,096-token context check reserves the reasoning budget plus one action token; overflow or an invalid action fails rather than truncating the prompt or substituting an expert action. These extensions belong only to their designated autonomous pilots, not the common offline fitting interface.

D Online Guidance and Demonstration Anchoring

D.1 Online Qwen Guidance and Demonstration Anchoring in BabyAI

The main BabyAI dataset/BC/reference replications use planner labels and no Qwen queries. The separate online-Qwen experiments used Qwen-derived example priorities, learner-visited coverage and live Qwen action-score guidance; these interventions are not the same as replacing planner labels with Qwen labels. The online Qwen–reference logit combination adds scaled Qwen scores to fitted-reference logits before softmax. It is not the offline expert–Qwen probability mixture of Appendix I. The complete prompt specification below gives the textual interface, discrete action mapping, information access and score reconstruction, including the distinction between clean and corrupted prompts. CrafText studies distinguish sequence priorities, hard action annotations, probability targets and autonomous collection explicitly.

Relation to the online-LLM baseline. The unanchored online-Qwen condition adapts InstructPPO's additive policy-to-LLM-prior KL regularization (Hu & Sadigh, 2023). It shares that training principle, not the original Hanabi implementation: we use BabyAI, Qwen action-token scores, BC initialization and a KL coefficient that decays to zero; Hu and Sadigh initialize from a pretrained self-play policy and retain a nonzero KL coefficient. Our anchor adds demonstration supervision throughout PPO. Fitted-reference conditions replace the online target by a frozen fitted model, and the combination uses both score vectors. Table 1 therefore compares adaptations within our recipe, not published InstructPPO scores. Its corrupted-description cohort and the separate clean-prompt follow-up do not establish a matched ranking of guidance sources.

Data and initialization for the online-Qwen comparison. The labelled pool combines two sources; every label is a planner action.

1. Planner rollouts contribute 13,932 states from 1,200 episodes. Qwen3-32B scores their descriptions after independent left/right swaps with probability .9 per string. Qwen is the predictor used to identify disagreement with the planner in this source.
2. Mixed planner/learner roll-in contributes 23,754 states. Each executed action comes from the planner with probability .5, otherwise from an earlier learner. Here that learner, not Qwen, supplies the predictor scores used to identify disagreement with the planner.

The final pool contains two copies of source 1 and one of source 2. Repeated copies of an episode stay in the same episode/source-grouped partition, preventing duplicate leakage across splits. The final BC fit uses 30 epochs, batch 256, AdamW at 3×10^{-4} with weight decay 10^{-4} and grouped 80/20 splits. Its replacement sampler uses non-goto weight 2, source-specific predictor/planner disagreement weight 2, and rare-action balancing capped at 5, without loss-side reweighting. The final reference

uses the same labelled pool and the objective in Eq. 1, not Qwen labels or Qwen features. The source learner was first initialized by BC on planner rollouts, then trained with reference-guided PPO. Pure learner-rollout collection missed door-opening prerequisites, motivating the mixed planner/learner collection used here. This development choice predates these paired PPO outcomes. The full supplement preserves the final labelled corpus and publication-stage fits. Replay of this auxiliary comparison starts from that corpus: the complete checkpoint chain for the earlier collection learner is not certified by the released records. This boundary does not concern the separately collected five-pipeline BabyAI main study.

Reference-fitting dataset. The dataset contains one copy of the 23,754 mixed planner–learner roll-in examples and two copies of the 13,932 planner-rollout examples. All targets are planner actions. Before the training/validation split, this corpus contains 37,686 distinct source rows and

$$23{,}754 + 2 \times 13{,}932 = 51{,}618$$

training rows. The planner-action counts are 9,124 `turn_left`, 12,809 `turn_right`, 21,746 `go_forward`, 4,022 `pick_up`, 2,706 `drop`, and 1,211 `toggle`. The task-family counts are 23,944 open, 16,336 putnext, 3,904 pickup, and 7,434 sequence. The fitted-reference training split is grouped by episode/source to avoid duplicate leakage: 1,440 training groups and 360 validation groups. The reference uses a GRU instruction encoder and the action-prediction objective in Eq. 1, without Qwen features or inputs.

Fitted reference and online Qwen combination. All rows in Table 1 initialize from the BC policy fitted on the mixed demonstration dataset above. The no-reference, no-anchor cell is BC-initialized PPO alone, not retained BC. Its five saved final evaluations give mean .08144 and sample standard deviation .06881. Qwen-only guidance uses $q_t^Q = \mathrm{softmax}(\ell_t^{\mathrm{LLM}})$ in Eq. 3, with no fitted-reference component or sharpness warm-up; its KL coefficient follows the same .05-to-zero cosine schedule. The fitted-reference rows use the frozen reference fitted to the training partition of that corpus and do not query Qwen during PPO. Their KL target is

$$q_t^{\psi} = \mathrm{softmax}(\rho_u r_\psi(o_t, m_t)).$$

The fitted reference + online Qwen combination adds Qwen scores during PPO:

$$\tilde{\ell}_t = \alpha \ell_t^{\mathrm{LLM}} + \rho_u r_\psi(o_t, m_t). \tag{20}$$

Here ℓ_t^{LLM} is the reconstructed six-action Qwen logit vector and α is its mixing coefficient. Let $q_t^{\mathrm{hyb}} = \mathrm{softmax}(\tilde{\ell}_t)$. The full training loss for the fitted reference + online Qwen combination branch is

$$\begin{aligned} L_{\mathrm{total}}(\theta) = L_{\mathrm{PPO}}(\theta) &+ \lambda_u \mathbb{E}_t \left[\mathrm{KL}\Big(\pi_\theta(\cdot \mid x_t) \,\|\, q_t^{\mathrm{hyb}} \Big) \right] \\ &+ \mu \mathbb{E}_{i \sim D_w^{\mathrm{anchor}}} \left[-\log \pi_\theta(a_i^\star \mid o_i, m_i, h{=}0) \right], \end{aligned} \tag{21}$$

The reference expectation uses the offset-wise valid-target reduction in Appendix D.2, including the all-invalid case. For the combined-reference row in Table 1, $\alpha = 0.25$, $\rho_u : 0 \to 2.0$ over the first quarter of training, and $\lambda_u : 0.05 \to 0$ by 0.75M frames. Table 1 uses $\mu = .2$ with the anchor and $\mu = 0$ without it. The anchor sampler uses a multiplier of 4.0 for every non-`goto` example, disagreement weight 2.0, and action-balance cap 5.0, without an additional loss-side sample multiplier. The last term uses the same labelled state pool. Evaluation uses only the trained policy; neither the BabyAI planner nor Qwen is queried at held-out test time.

Later completion of the combination without anchoring. This additional descriptive cell was specified after the seven existing cells were observed. Five PPO seeds (1–5) reuse the same BC initializer and frozen reference, with the combination's archived command and $p = .9$, changing the demonstration-anchor coefficient to $\mu = 0$. Runs use L40S GPUs and finish at 1,500,160 environment frames. Greedy policy-only evaluation uses 500 episodes per family on the same five-family held-out construction, with environment seed base 100. The five macro success rates are .1204, .0632, .1244, .1032, .2004, giving mean .12232 and sample standard deviation .04991. These complete Table 1; they are not new upstream replications, paired confidence intervals, or evidence that execution-setting differences have been isolated. The accompanying five-policy record check is separate from the original 35-policy table reconstruction.

Clean-prompt follow-up and selection. With structured, uncorrupted Qwen prompts, final success is $.173 \pm .049$ without anchoring and $.326 \pm .030$ with anchoring (five PPO seeds each; mean $\pm$ sample standard deviation, not a confidence interval). Table 1 retains the .9 corrupted-description condition. We chose this high swapping probability to induce frequent mismatches between Qwen's text and the policy's unchanged symbolic observation, not as an action-error rate or a test of naturally occurring errors. The clean follow-up initially used three seeds per arm and expanded an arm to five if its mean exceeded the fitted-reference mean or improved on corrupted prompts by at least three percentage points. Only the anchored arm triggered that expansion. After these outcomes were known, a separate completion plan fixed the addition of unanchored seeds 4 and 5, retaining seeds 1–3 regardless of the new outcomes. Both added policies completed; the anchored cohort is unchanged. This completes the planned seed inventory, not an independent confirmation of the earlier means.

The additional runs reuse the same BC initializer and Qwen model revision, tokenizer, structured prompt and serving settings in Appendix D.2. Only the PPO seed and output identifier change: $p = 0$, $\mu = 0$, Qwen-only targets (no fitted-reference combination), and the same .05-to-zero KL schedule apply. Each policy trains for a requested 1.5M frames and is evaluated at its final 1,500,160-frame checkpoint, not a test-selected checkpoint. Greedy policy-only evaluation uses 500 episodes per family on the five-family held-out construction with seed base 100. The five unanchored macro success rates, in seed order 1–5, are $.1716, .1860, .1344, .2492, .1252$, giving mean $.17328$ and sample standard deviation $.04938$. Each seed score equally averages the five family success rates; the reported spread uses the $n - 1$ denominator defined in Appendix A.5. The smoke test is excluded and neither new policy is dropped. No further outcome-triggered expansion follows. Clean runs use L40S GPUs (two for Qwen serving and one for PPO), whereas the original .9 runs preceded the hardware audit. The comparison does not isolate corruption from execution-setting differences. Its ten policies share upstream models, rather than representing ten independently constructed pipelines.

Action-coordinate control. This diagnostic uses the mixed-corpus BC initializer and fitted reference specified above, not the five independently rebuilt planner-only pipelines. In action order (left, right, forward, pickup, drop, toggle), let $z = (z_0, \ldots, z_5)$ be the reference logits. Every target row instead uses $\widetilde{z} = (z_5, z_3, z_4, z_0, z_1, z_2)$ before sharpness scaling and softmax. PPO seeds 1–5 otherwise use the Aligned local-reference recipe: 1.5M requested frames, the same coefficient and sharpness schedules, and demonstration anchoring. Final greedy evaluation uses the same seed-base-100 set as Aligned, with 500 episodes per family. Mean final success (sample standard deviation over policy seeds) is $.038 \pm .017$, versus $.294 \pm .038$ for Aligned. This changes what each action coordinate means, not which state supplies a vector.

Removing both auxiliary losses during training. On the same mixed-corpus initialization, ten paired PPO seeds 1–10 compare Aligned with a Uniform target, giving 20 policies. Aligned uses the usual sharpness warm-up to 2 by .375M frames; the Uniform arm uses uniform targets. Both decay the reference coefficient from .05 to zero by .75M frames and switch the anchor weight from .2 to zero at that boundary. The boundary uses accumulated interaction frames; PPO continues with its existing policy and optimizer to a requested 5M frames. Final greedy evaluation uses the seed-base-100 set of 500 episodes per family. The mean of ten paired Aligned-minus-Uniform endpoint differences is $+.020$, with 95% Student-t interval $[-.018, +.058]$ (nine degrees of freedom). This tests persistence after removing training losses, not ordinary deployment, which already uses only the policy. The comparison uses one initializer, not ten independent upstream pipelines, and does not establish a lasting advantage.

D.2 Structured Qwen prompt and action target

In the online-Qwen BabyAI comparisons, the text-conditioned action prior is Qwen3-32B served locally. It is not queried with a bare state-plus-action prompt. Table 14 summarizes the evaluated interface; the model's non-thinking mode is part of the frozen configuration. Table 13 connects these settings to the reported endpoints; offline annotation corruption is a separate data-construction setting. The paragraph following the table explains the information these blocks supply; the supplement maps the settings to immutable saved configuration names.

Table 13: Online prompt settings for Table 1 and the separate clean-prompt comparison. All Qwen queries use the structured modes in Table 14. p selects each environment-description string for left/right swapping before derived text blocks are built; it is not an action-error rate. The released row manifest and launcher specify this correspondence.

Condition	p	Anchor μ	PPO seeds
Table 1: Qwen without anchor	.9	0	5
Table 1: Qwen with anchor	.9	.2	5
Table 1: online combination without / with anchor	.9	0 / .2	5 each
Table 1: fitted reference without / with anchor	not queried	0 / .2	5 each
Table 1: PPO only / Anchor only	not queried	0 / .2	5 each
Separate clean Qwen without anchor	0	0	5
Separate clean Qwen with anchor	0	.2	5

Table 14: Online Qwen configuration.

Field	Configuration
Model	`Qwen/Qwen3-32B`, revision `9216db5781bf...c16137`
Serving	vLLM completion endpoint; bfloat16 (BF16); tensor parallelism 2; seed 0
Inference	Hugging Face (HF) chat template; `enable_thinking=false`; temperature 0; top-$p = 1$
Output request	At most 4 tokens; top 20 token log-probabilities
Prompt content	Structured task and action rules; last-seen object memory (8 items)
Derived blocks	Prerequisites, action-feasibility notes and target hints; no navigation block
Actions	`1=turn_left`, `2=turn_right`, `3=go_forward`, `4=pick_up`, `5=drop`, `6=toggle`
Caching	No persistent query cache; identical prompts deduplicated within a query batch

The chat interface contains a system message, "You are an RL agent task reasoner," and a user prompt comprising mission/instruction context, fixed action semantics and rules, prerequisite notes, recent-state memory, last-step deltas, programmatic target and feasibility hints, a decision summary, the current textual observation, and the numbered action map. The Hugging Face chat template wraps these messages with thinking disabled.

Table 15 makes the information interface explicit. The PPO policy and fitted reference consume the partial symbolic image and mission; the PPO policy also maintains its own recurrent hidden state. Qwen instead receives a textualized and programmatically augmented interface. The Qwen rows are therefore secondary structured-reference comparisons, not information-matched tests of model architecture.

The completion client selects the first generated position whose stripped token is a digit 1–6; if none exists, it selects the first top-logprob position containing any such digit. Observed digit log-probabilities form the six scores, and each missing digit receives $\min_j \ell_j - 5$. If no digit has a finite score, the row is masked out of the Qwen KL after two retries beyond the initial request. The categorical normalization yields a reconstructed and imputed six-action target, not Qwen's complete native output distribution. Valid targets are stored in the rollout and reused across PPO epochs; queries stop when $\lambda_u = 0$.

Invalid targets and recurrent reduction. For each PPO minibatch, let V_ℓ contain the parallel sequence indices with valid targets at recurrent offset $\ell = 0, \ldots, 19$. The Qwen-containing reference contribution is

$$\frac{\lambda_u}{20} \sum_{\ell=0}^{19} g_\ell, \qquad g_\ell = \begin{cases} |V_\ell|^{-1} \sum_{i \in V_\ell} D_\epsilon(\pi_{\ell i}, q_{\ell i}), & |V_\ell| > 0, \\ 0, & |V_\ell| = 0, \end{cases}$$

where $D_\epsilon(\pi, q) = \sum_a \pi(a)[\log(\pi(a) + 10^{-8}) - \log q(a)]$ uses Appendix C.3's numerical convention. Here $\pi_{\ell i}$ is the policy distribution and $q_{\ell i}$ is the fixed Qwen-only or combined target. The outer denominator remains 20 even for an empty valid set; this is not a pooled mean over all valid rows. An invalid Qwen target also omits the entire combined-reference KL at that position, without fitted-reference-only fallback. PPO still uses those rollout positions; the separately sampled demonstration anchor is unchanged and added after recurrent averaging.

Table 15: Prompt-field provenance. "Local" covers the PPO policy and fitted reference; their exact temporal architectures differ.

Field	Source	Local	Qwen
Mission	Environment observation	yes	yes
Partial symbolic image	Environment observation	yes	not directly
Text descriptions	BabyAI-Text environment info via text wrapper	no	yes
Global orientation	Observation direction via context wrapper	no	yes
Carried-object state	Observation/info or `env.unwrapped.carrying` fallback	no	yes
Failure history	Instruction-context wrapper state	no	yes
Temporal memory	Separate policy/Qwen internal state	recurrent h_t	recent text, actions, deltas
Rules and derived hints	Fixed prompt rules and text/context functions	no	yes

For left/right-corruption probability p, each environment-description string is selected independently and every left/right occurrence in a selected string is swapped before downstream text blocks are constructed. Fixed action names, semantics, rules, and the policy-visible symbolic observation remain unchanged. Table 1's Qwen-containing conditions use $p = .9$; the separate uncorrupted-prompt comparison uses $p = 0$. The clean example below is generated from a fixed initial state of `BabyAI-MixedTrainLocal-v0` passed through the evaluated text and instruction-context wrappers, then through the released Qwen prompt builder, including its orientation and inventory fields. For the compact corrupted display, the released operator is applied with $p = 1$ to the same state and the context is rebuilt through the same wrappers; complete clean and corrupted prompts remain in the machine-readable artifact.

Clean prompt from the evaluated interface.

```
You are an RL agent task reasoner.
Instruction Context:
Mission: Open the blue door. Target colors: blue. Target objects: door. Orientation: facing
↪  south. Inventory: empty. Step 1: open the blue door
Action semantics (BabyAI/MiniGrid):
- turn_left / turn_right: rotate in place (position unchanged).
- go_forward: move 1 cell forward if the cell ahead is free; otherwise you stay in place.
- pick_up: pick up the object in the cell directly in front of you (only if inventory is
↪  empty).
- drop: drop the carried object into the cell directly in front of you (only if that cell is
↪  empty).
- toggle: interact with the cell directly in front of you (e.g., open/close doors; locked
↪  doors may require the matching key).

Universal decision rules:
1) Active step = the first Step not yet satisfied (follow steps in order).
2) Hard feasibility constraints (when these fields are present, obey them):
   - Never choose go_forward if forward_blocked_now: yes.
   - Never choose pick_up if can_pick_up: no.
   - Never choose drop if can_drop: no.
   - Never choose toggle unless front_cell is a door AND can_toggle: yes.
   Prefer choosing only from allowed_actions_now.
3) Priority actions (when present and feasible):
   - If priority_action_now is shown -> choose it.
   - If drop_ready: yes -> choose drop.
   - If drop_ready: no -> do NOT choose drop (wait until drop_ready: yes).
   - If can_toggle: yes AND front_cell is a door -> choose toggle.
   - If can_pick_up: yes AND the required object is 1 step forward -> choose pick_up.
4) Recovery:
   - If carrying_wrong_for_pickup: yes -> free inventory: if can_drop: yes then drop; else
   ↪  navigate to an empty front cell and then drop.
   - If blocked_forward: yes -> turn once to change the front cell before attempting
   ↪  go_forward again.
5) Navigation:
   - If suggested_nav is given, follow it (but still obey hard feasibility constraints above).
   - If recent_turns_in_row >= 2 AND go_forward is allowed -> choose go_forward (avoid spin
   ↪  loops).

Prerequisites (universal):
- Locked doors: toggle only works when a door is directly in front; locked doors require
↪  carrying the matching key.
- Put-next-to: interpret "put X next to Y" as:
  - MOVE object = X (the one you must carry),
  - REFERENCE object = Y (do NOT pick it up).
  You must pick up MOVE before you can place it; drop places the carried object into the front
  ↪  cell if empty.
```

```
Target hints (derived):
- mode: seek_key
- key_target: blue key (not visible)
- door_target: blue door (not visible)
- active_target: blue key
- carrying: empty
- note: need blue key before toggling blue door

Feasibility (derived):
- inventory: empty
- front_cell: wall
- forward_blocked_now: yes
- can_pick_up: no; can_drop: no; can_toggle: no
- blocked_forward: no; last_interaction_no_effect: no
- active_step_hint: Step 1: open the blue door
- active_pickup_target: blue key
- active_target_in_front: no
- inventory_matches_active_pickup: no
- carrying_wrong_for_pickup: no
- allowed_actions_now: turn_left, turn_right
- hard_constraint: choose only from allowed_actions_now (invalid actions waste a step)

Decision now (obey these):
- allowed_digits_now: 1, 2

Observation: You see a wall 1 step forward You see a wall 1 step right

Actions: 1=turn_left 2=turn_right 3=go_forward 4=pick_up 5=drop 6=toggle
Answer with a single digit from 1 to 6. You must not output anything else.
Answer:
```

Left/right changes for the same state ($p = 1$ illustration).

```
--- clean publication-wrapper prompt
+++ cfLR p=1 prompt from the same state
@@ -65,3 +65,3 @@

-Observation: You see a wall 1 step forward You see a wall 1 step right
+Observation: You see a wall 1 step forward You see a wall 1 step left
```

The supplement's prompt record contains the canonical template, field provenance, extraction rules, both complete examples, their differences and request configuration; the supplement inventory records its SHA-256.

D.3 Online-Qwen Comparison: PPO-Stage Resource Accounting

Table 16 reports the measured PPO-stage allocation for the five paired local-reference and structured-Qwen runs.

Table 16: Measured PPO-stage resources across five fresh L40S runs. GPU-hours are allocated devices times Slurm elapsed time. Offline model fitting and exact token counts were not comparably instrumented.

Quantity	Fitted reference	Structured Qwen + anchor
Environment frames	7,500,800	7,500,800
L40S devices per run	1	3 (2 server + 1 PPO)
Summed job wall time	2.29 h	181.56 h
Allocated L40S GPU-hours	2.29	544.68
Successful Qwen HTTP batches	0	58,600
Qwen state prompts	0	3,750,400
Deployment external calls	0	0

Each structured-Qwen run issues 11,720 successful 64-prompt batches during the KL-active first 750,080 frames. The 238-fold allocation ratio is specific to this implementation and includes model loading and server startup. The offline collection and BC/reference fitting for this online-Qwen study lack comparable wall-time and token logs, so the ratio covers PPO only. The main planner-only BabyAI study makes no Qwen calls, but its different offline dataset cannot supply those missing costs. The supplement supplies raw accounting rows and their validation alongside the resource summary.

Appendix D.1 gives the associated online-guidance cohort, clean-prompt follow-up, action-coordinate control and auxiliary-loss-removal comparison; those protocol descriptions are separate from this cost accounting.

E Main BabyAI Study: Independently Constructed Pipelines

E.1 Five-pipeline protocol

Fitting and anchor recipe. The five independently rebuilt BabyAI pipelines have record IDs L4–L8 and independently collect 300 episodes each of `pickup`, `open`, `putnext`, and `seq`, capped at 128 environment steps per episode. Collection ends on termination, truncation, this step limit or a logged planner exception. Valid rows collected before an episode ends are retained; successful completion is not an episode-selection requirement. Table 17 gives the independent collection and fitting seeds. Both BC and reference fitting use 30 epochs, batch size 256, AdamW with learning rate 3×10^{-4}, weight decay 10^{-4} and gradient-norm cap 5, $(\beta_1, \beta_2) = (.9, .999)$ and $\epsilon = 10^{-8}$, and episode-grouped 80/20 training/validation splits. They sample training rows with replacement and apply no further loss-side sample weights. BC has zero entropy coefficient; the reference predicts action logits directly with scale one and $\beta_r = 10^{-3}$ on mean squared logits, as in Eq. 1. This fitting objective does not correct another model's logits. These fitting choices are distinct from online PPO's optimizer settings.

Table 17: BabyAI source seeds. Fitting seeds also determine the respective episode-grouped split; PPO seeds are 1–4 for each main-study condition.

Source	Collection	Environment base	BC	Reference
L4	401	5,000,000	1,401	2,401
L5	501	6,000,000	1,501	2,501
L6	601	7,000,000	1,601	2,601
L7	701	8,000,000	1,701	2,701
L8	801	9,000,000	1,801	2,801

The fitting samplers use non-`goto` multiplier 2, no predictor–planner disagreement reweighting (multiplier 1), and rare-action balancing capped at 5. The online anchor samples 256 rows with replacement from the full labelled pool (D_w^{anchor}), uses multipliers 4 and 1, the same action-balance cap, no loss-side sample weights, and coefficient $\mu = .2$. Specifically, with N labelled rows and $n_a > 0$ rows labelled action a, the action-balance factor is $\min(5, N/(6n_a))$; the family multiplier is applied to every non-`goto` row before normalization. All four collected families are non-`goto`, so the family multiplier is constant and cancels. Rare-action balancing remains operative; neither 2 nor 4 preferentially upweights `open`/`putnext` in this cohort. The five BC and five reference fitting configurations and the PPO anchor settings are included in the supplement. Online-Qwen-study disagreement weights summarized in Appendix D.1 do not apply here.

Protocol and evaluation. Before inspecting aggregate outcomes, we specified and hashed five independently rebuilt planner-rollout pipelines, four policy seeds, four main-study conditions, blocks 800–1100, and the decision rules in the released protocol. Each pipeline independently recollects 1,200 planner-controlled episodes and retrains grouped-split BC and fitted-reference models; no Qwen call, stored Qwen logit, learner rollout, or intermediate model from another study enters these pipelines. Normalized trapezoidal AUC uses the actual saved frame grid $(0, 250880, 501760, 750080, 1000960, 1500160)$ in Eq. 7, including the initial BC evaluation. The primary comparison is aligned minus Uniform target; Aligned minus Reassigned is the second prespecified assignment comparison. Each requires all artifacts, policies, and evaluations valid, five positive pipeline means, and a positive lower endpoint of the five-pipeline 95% t interval.

All five pipeline-validity checks, 80 main-study policies, and 1,620 expected evaluation files pass. The supplement retains complete row-level evaluations, policy logs and hashes identifying omitted training payloads. One evaluation process failed to construct a requested sequence family within its fixed 50-attempt limit and wrote no result; an identical rerun of that model, block, seed, evaluator, and

limit produced the missing file. Both attempts were logged; the failed process produced no evaluation observation and the successful rerun supplies the reported result.

After those results were complete, we specified and hashed a separate global coefficient-calibration rule before training its 20 policies or running its 400 evaluations. For each completed Aligned and Reassigned run, the rule first averages λ_u times mean policy–target KL over active-schedule log rows, then averages the 20 run values equally within each condition. It multiplies the main-study $.05$ by the Aligned-to-Reassigned ratio, producing one initial coefficient, $\lambda_0 = .022916$, for every new calibrated policy while retaining the same decay. The completed Aligned and Reassigned runs informed this rule, but no new calibrated-policy outcome could change it. This robustness check was specified after the main results, outside their prespecified comparison hierarchy.

After the primary Uniform-target result was known, we also froze a two-sided local sensitivity before training its 40 policies or running its 800 evaluations. Uniform-target coefficients $.025$ and $.10$ bracket the primary $.05$ under the same annealing schedule, seeds, blocks, checkpoints, and five-pipeline rule; both complete outcomes were committed for retention. This sensitivity brackets one uniform-target recipe rather than all entropy regularization. Table 18 reports the observed entropy statistics; they are not an entropy-matching criterion.

Table 18: **Post-hoc main-study training diagnostics.** Policy entropy is the training-time minibatch statistic, averaged equally over four seeds and five pipelines. The active-window column averages all 293 updates through the 0.75M KL cutoff; it is descriptive rather than an entropy-matching criterion.

Condition	Active-window mean	0.25M	0.50M	0.75M	1.50M
Aligned	.651472	.716170	.633038	.664545	.667132
Reassigned	.751866	.816522	.790164	.769647	.747358
Uniform target	.827818	.858401	.904062	.858681	.778625
Anchor only	.700116	.681262	.787631	.769042	.748141

The same released logs verify target reassignment in all 20 main-study Reassigned policies. Their mean changed-logit-vector rate, before applying ρ_u, is $.820719$ (policy range $.765248$–$.865305$), the maximum logged singleton-row rate is $.002734$, the maximum entropy-invariance error is 1.192×10^{-7}, and the logged action-marginal error is zero. Donor episode identity and lag were not logged in these deterministic runs. These numerical checks use raw logits and their temperature-one softmax; preservation at other shared values of ρ_u follows from the unchanged multiset of logits.

Table 19: Main-study five-pipeline BabyAI effects. AUC is normalized over 0–1.5M frames; endpoint uses the final checkpoint. Every row is aligned minus the named control.

Metric	Control	L4	L5	L6	L7	L8	Mean	95% pipeline interval
AUC	Reassigned	+.0376	+.0272	+.0681	+.0555	+.0427	+.0462	$[+.0265, +.0660]$
AUC	Reassigned (global calibration)	+.0266	+.0334	+.0725	+.0495	+.0514	+.0467	$[+.0245, +.0689]$
AUC	Uniform target ($\lambda_0 = .025$)	+.0256	+.0449	+.0808	+.0528	+.0558	+.0520	$[+.0272, +.0767]$
AUC	Uniform target ($\lambda_0 = .05$)	+.0328	+.0374	+.0663	+.0283	+.0314	+.0392	$[+.0200, +.0585]$
AUC	Uniform target ($\lambda_0 = .10$)	+.0177	+.0183	+.0316	+.0368	+.0194	+.0248	$[+.0138, +.0357]$
AUC	Anchor only	+.0504	+.0592	+.1039	+.0475	+.0654	+.0653	$[+.0371, +.0935]$
Endpoint	Reassigned	+.0346	+.0180	+.0701	+.0468	+.0676	+.0474	$[+.0200, +.0749]$
Endpoint	Reassigned (global calibration)	+.0210	+.0394	+.1058	+.0431	+.0665	+.0552	$[+.0147, +.0956]$
Endpoint	Uniform target ($\lambda_0 = .025$)	+.0263	+.0262	+.0542	+.0390	+.0824	+.0456	$[+.0164, +.0749]$
Endpoint	Uniform target ($\lambda_0 = .05$)	+.0323	+.0539	+.0309	+.0322	+.0390	+.0377	$[+.0257, +.0496]$
Endpoint	Uniform target ($\lambda_0 = .10$)	+.0211	+.0223	+.0240	+.0511	+.0280	+.0293	$[+.0138, +.0448]$
Endpoint	Anchor only	+.0549	+.0291	+.0826	+.0082	+.0940	+.0538	$[+.0093, +.0982]$

For AUC, positive paired-seed counts are 19/20, 19/20, 20/20, 20/20, 17/20, and 20/20 against Reassigned, Reassigned (global calibration), Uniform target at $.025/.05/.10$, and Anchor only; all block means are positive. The Uniform-target sensitivity crossed-bootstrap intervals are $[+.0337, +.0700]$ and $[+.0115, +.0399]$. Endpoint positive pair counts for those two new arms are 16/20 and 17/20. The calibrated arm reduces the fixed reassigned-to-aligned mean coefficient-weighted KL ratio from 2.18 to 1.13 without closing the AUC gap; global coefficient calibration still does not match per-state KL or gradients. Uniform target minus base-coefficient Reassigned AUC is only $+.0070$ (4/5 pipeline means; pipeline interval $[-.0100, +.0241]$), and anchor only minus reassignment is $-.0191$

with interval $[-.0408, +.0027]$; these controls are not an ordered severity scale. For completeness, the original 95% crossed-bootstrap AUC sensitivity intervals are $[+.0259, +.0555]$ against base Uniform target, $[+.0311, +.0615]$ against Reassigned, $[+.0455, +.0873]$ against Anchor only, and $[+.0293, +.0655]$ against globally calibrated Reassigned. These summarize sensitivity; the pipeline intervals in Table 19 govern the stated decisions.

F CrafText Assignment and Reference-Control Studies

F.1 CrafText replication

CrafText extends Craftax-Classic with goal-conditioned tasks expressed as instructions (Volovikova et al., 2025; Matthews et al., 2024). We use `easy_strict_train_no_paraphrases`: one tokenized canonical instruction for each of 57 semantic goals, with paraphrases disabled, and a 17-action ontology. The expert, behavior initialization, fitted reference, and PPO policy consume the same vector state and canonical instruction tokens; the corresponding 57-valued mission identifier is used only to form exact donor groups. We query no LLM and use no LLM action reference during training or evaluation; the instruction strings are inherited from the benchmark release.

Predeclared decisions and model-validity checks. The expert configuration was chosen through preliminary comparisons, before the assignment studies. A 100M-step screen compared failure-weighted mission sampling, instruction-modulated features, and their combination; no variant passed all fixed checks on both development seeds. A separate 250M-step screen on two new development seeds compared weighted sampling with and without instruction-modulated features and selected their combination. The selection ordered passing variants by minimum goal coverage, minimum mean success, mean goal coverage, then lower complexity on exact ties. These development seeds and outcomes are excluded from the assignment studies. The resulting expert recipe, full-position fitting and PPO schedule are specified here and in Appendices C.4 and C.1.

The first CrafText study trained three pipelines and produced positive Aligned-minus-online-calibrated means in all three, but its three-pipeline interval crossed zero. Its logger also retained segment means rather than all per-update diagnostics. That study remains a non-confirmatory sensitivity and none of its models or outcomes enters the five-pipeline decision below.

For the subsequent five-pipeline assignment study, we fixed the expert–dataset–BC– reference construction before generating its outcomes. Each expert used a 250M-step feature-wise linear modulation (FiLM) configuration and had to pass success, random-improvement, mission-coverage, action-support, donor- eligibility, and allocated-resource validity checks. Each eligible expert then labelled 1,048,576 transitions with disjoint train, validation, and test environments. BC and reference artifacts were separately retrained and had to pass held-out accuracy, closed-loop success, action-support, completeness, and provenance checks. All five pipelines passed; failed artifacts could not be replaced. For sources 4–8, let s be 105061, 106063, 107071, 108079 and 109087, respectively. Expert training uses s; collection, BC fitting, reference fitting and closed-loop fitting evaluation use $s + 100000$, $s + 200000$, $s + 300000$ and $s + 400000$. The three PPO seeds are $s + 500000 + k$, $k = 0, 1, 2$, with evaluation base $s + 700000 + k$ and block offset $100003b$, $b = 0, 1, 2, 3$. The reference-control study below reuses these seed assignments but trains new policies.

The fixed expert checks require mean goal success at least .30, improvement over paired random play at least .25, nonzero success on at least 34 goals, at least 12 supported actions, donor eligibility at least .99, and at most two allocated L40S hours. Corpus *donor eligibility* is the fraction of rows whose exact instruction occurs at least twice in the same 64-step collection chunk, pooling all environment streams in that chunk. The expert-training check applies the same count to each 64-step rollout across its 1,024 environments and requires the minimum rollout eligibility to meet the threshold; corpus eligibility pools counts across its 256-stream chunks. It measures availability of another same-instruction row, not action agreement or useful guidance. An action is supported when its empirical frequency is at least 10^{-4}. Corpus checks require finite complete arrays, all 57 goals, at least 12 supported actions, donor eligibility at least .99, exact expert/collection provenance and disjoint stream splits. BC and reference checks require test action accuracy at least .40 and .55, respectively; each must attain closed-loop success at least .25, zero unfinished evaluation episodes and at least 12 supported actions. Validation, test and closed-loop metrics must be finite, and checkpoint/dataset/expert hashes must

Table 20: Five-pipeline CrafText assignment effects. Values are aligned minus the named control and are normalized learning AUC unless marked endpoint. C4–C8 denote CrafText pipelines 4–8 (stored as L4–L8); they are unrelated to BabyAI pipelines.

Comparison	C4	C5	C6	C7	C8	Mean
Reassigned (online coefficient calibration) AUC	+.0171	+.0035	+.0227	+.0163	+.0281	+.0175
Reassigned (base coefficient) AUC	+.0127	+.0118	+.0184	+.0264	+.0344	+.0207
Reassigned (online coefficient calibration) endpoint	+.0175	−.0051	+.0146	+.0154	−.0066	+.0072

match. These are eligibility checks on the already selected checkpoints, not rules for selecting a better epoch or replacing an unsuccessful pipeline.

Assignment intervention and training. Each pipeline contributed three paired PPO seeds to three conditions: *Aligned*, *Reassigned*, and *Reassigned (online coefficient calibration)*. The reference weights stay frozen, but its recurrent hidden state advances along each environment trajectory. Reassignment moves the resulting output-logit vectors, not hidden states, by seeded random single cycles (the distribution generated by Sattolo, 1986) only among rows with the exact same canonical instruction. At each PPO update, the calibrated condition scores the aligned and reassigned targets against the same rollout-collection policy, divides aligned mean KL by reassigned mean KL, treats that ratio as a constant during backpropagation, clips it to $[.01, 10]$, and multiplies the Reassigned KL coefficient by it. The frozen validity range for the resulting coefficient-weighted-KL ratio was $[.9, 1.1]$. When denominator flooring and clipping are inactive, the operation equates one update-mean coefficient-weighted KL contribution at the collection policy, not necessarily after PPO minibatch updates. It does not equate raw or statewise KL, gradient norm or direction, or induced visitation; Appendix A gives the step-by-step definition and a numerical example.

Each policy used 1,024 parallel environments, 64 steps per update, 64 updates, four PPO epochs, and 4,194,304 total frames. The reference coefficient started at .05 and decayed to zero halfway through training; the anchor coefficient was .01. Nine checkpoints were evaluated on four paired blocks, with two episodes for each of 57 instructions per block, each capped at 1,000 steps.

Frozen decision and diagnostics. The primary estimand was aligned-minus-calibrated normalized learning AUC. A replication required all 45 rows and intervention checks to pass, all five pipeline means to be positive, and the lower endpoint of the 95% t interval over those five means to exceed zero. The crossed-bootstrap sensitivity analysis uses 20,000 draws and NumPy's default random generator with seed 98131. Starting from each paired block-level AUC (or endpoint) difference, each draw samples five pipeline identifiers with replacement and four block identifiers with replacement. The same sampled blocks are used across that draw. For each sampled pipeline occurrence, independently sample three of its paired PPO seeds with replacement; average all $5 \times 3 \times 4 = 60$ selected differences. Sorted pipeline, seed and block identifiers determine lookup order. The 2.5th and 97.5th percentiles use NumPy's linear quantile interpolation. The random generator is restarted at seed 98131 for each comparator and metric. This sensitivity interval does not replace the prespecified pipeline-level rule.

All 45 rows passed. Donor mission mismatch and multiplicity errors were zero, nonidentity and numerically changed-target rates were effectively one, argmax marginals were preserved exactly, and the maximum entropy-invariance error was 2.384×10^{-7}. In every calibrated run, the coefficient-weighted Reassigned-to-Aligned KL ratio met the predefined range $[.9, 1.1]$.

Table 20 gives the per-pipeline effects. For the primary comparator, 14/15 paired seeds and 4/4 block means were positive. The crossed-bootstrap sensitivity interval was $[+.0063, +.0281]$ and the prespecified five-pipeline interval was $[+.0062, +.0289]$. Endpoint effects were positive in only 8/15 pairs and their pipeline interval was $[-.0077, +.0220]$. The fixed- coefficient AUC comparison was secondary (mean $+.0207$; pipeline interval $[+.0088, +.0327]$). Its slightly larger effect than the calibrated comparison ($+.0175$) is directionally consistent with KL magnitude contributing to the contrast, but it does not identify that mechanism. A prespecified cumulative sensitivity over all eight CrafText pipelines is positive (mean $+.0207$; interval $[+.0118, +.0296]$), but it retains the first study's logging deviation and cannot replace the five-pipeline decision.

Table 21: CrafText reference-control effects with shared data and initialization. Values are Aligned minus the named control over five offline pipelines and three paired PPO seeds per pipeline. C4–C8 identify the same CrafText pipelines as Table 20.

Comparison	C4	C5	C6	C7	C8	Mean
Anchor-only AUC	+.0175	+.0053	+.0342	+.0188	+.0121	+.0176
Uniform-target AUC	+.0181	+.0074	+.0314	+.0435	+.0523	+.0306
Anchor-only endpoint	−.0029	+.0270	+.0139	+.0146	+.0022	+.0110
Uniform-target endpoint	+.0132	+.0110	+.0205	+.0417	+.0146	+.0202

Table 22: Absolute CrafText reference-control scores, averaged over five pipelines and three policy seeds per pipeline. Both metrics are fractions. Retained BC is evaluated without PPO and therefore has a constant learning curve.

Policy	Time-averaged success	Final success
Aligned	.2956	.3241
Anchor only	.2780	.3132
Uniform target	.2650	.3039
Retained BC	.3118	.3118

Reference controls sharing the assignment-study data and BC. After the positive assignment result, we fixed and hashed the follow-up design before training new policies, using the same five accepted offline pipelines. For each pipeline and three paired PPO seeds, we retrained Aligned, Anchor-only, and Uniform-target policies from the same behavior-cloned initialization. The primary estimand was aligned minus Anchor-only AUC; Aligned minus Uniform-target AUC was secondary. No inspected assignment-study policies were reused, and all 45 newly trained policies passed the prespecified model, checkpoint, and evaluation validity checks. Because this extension reuses the five accepted assignment-study pipelines, its inference is specific to this selected canonical-goal configuration rather than an outcome-independent CrafText sample.

Table 21 reports the complete follow-up. Aligned-minus-anchor AUC is $+.0176$ (13/15 policy pairs; 5/5 pipeline means; pipeline interval $[+.0043, +.0309]$), and Aligned-minus-Uniform-target AUC is $+.0306$ (14/15; 5/5; $[+.0079, +.0532]$). The aligned-minus-anchor endpoint interval crosses zero ($+.0110$; $[-.0036, +.0255]$), whereas the Aligned-minus-Uniform-target endpoint interval is positive ($+.0202$; $[+.0046, +.0357]$). Thus, under this prespecified configuration, Aligned CrafText guidance improves the finite-budget trajectory over both removing the reference KL and using a uniform target under the same KL schedule; only the latter endpoint contrast is established.

Table 22 gives the absolute scores for this reference-control study. These are not the separately trained policies in the assignment comparison above.

G Assignment With the Top Recommended Action Preserved

Question. Reassigning a reference output can change its most likely action as well as the rest of its probability vector. This study asks whether assigning the complete vector to its own state matters when the recipient's top recommended action is unchanged. For example, the distributions $(.6, .3, .1)$ and $(.5, .49, .01)$ recommend the same action in a three-action illustration, but exert different KL pressure. The intervention changes complete vectors, not only their non-top coordinates; it does not isolate confidence, local gradients or instruction semantics.

Which rows exchange targets? Each row records one environment step. The donor supplies its reference vector; the recipient receives it. Both controls below act on the same eligible rows. Let z_i denote the frozen reference logits computed for rollout row i, m_i its exact instruction and $c_i = \arg\max_a z_{i,a}$ its top recommended action, breaking ties by the lowest action index. Define

$$\mathcal{E} = \{i : \exists j \neq i,\ m_j = m_i,\ c_j = c_i\}. \tag{22}$$

Here $\exists j \neq i$ means that a different row j with both matching properties exists. Rows outside $\mathcal{E}$ retain their own vectors in every arm. Aligned retains all own- state vectors. Unrestricted Reassigned

draws a random single cycle among the eligible rows for each instruction. Reassigned (same top action) instead draws a random single cycle within each eligible instruction–top-action group. Every eligible row moves under both donor rules; complete target-vector multisets remain unchanged within instructions, and the latter rule preserves each recipient's top recommended action. Singleton groups cannot move.

The two donor rules act on the same eligible rows when applied to a given rollout. This does not imply that different trained policies continue to visit the same states or have identical eligible sets. Donor mappings are fixed within an update's PPO epochs. BabyAI uses a private random stream so donor construction does not advance the policy-training random stream. CrafText reserves the same donor-key split for all arms. The new CrafText comparison uses a fixed reference coefficient schedule without online calibration.

Reference provenance and controls. BabyAI retains the original planner-label-trained reference, BC initialization, and continuing .2 demonstration anchor. CrafText reuses the ten references from the direct-label study: actual Qwen hard labels and paired expert-argmax labels on identical expert-visited fitting states, with matched fitting budgets. These are not newly collected autonomous-LLM demonstrations. Expert-provided state coverage, BC and online anchoring remain unchanged. Qwen's annotation interface is stateless whereas the expert uses recurrent context. No LLM is queried by PPO or the deployed policy in this new study.

For each of five existing upstream sources, four new BabyAI policy seeds train the three assignment arms, giving 60 policies. In CrafText, three new policy seeds train both reference sources under all three assignment rules, giving 90 policies. Another 15 Anchor-only policies, one per CrafText source and seed, use the same BC and anchor without reference KL. These 15 baseline outcomes are shared across teacher comparisons, not counted as 30 independent baselines. Retaining the actual initial BC policy is a separate comparison.

Initial task construction. An engineering audit found that the initial BabyAI assignment study's constructor draws its task family before the training entry applies the environment seed. Matching the later seeds therefore does not guarantee identical initial task-family choices. The new study seeds before that draw in all three arms and checks their saved first-rollout instructions and reference logits. Original code and completed records remain unchanged. This limits claims of exact initial pairing in that study; it does not demonstrate systematic treatment bias or an error in the released evaluation arithmetic. All new arms are rerun; corrected controls are not paired with that study's Aligned outcomes. Because donor eligibility also changes, old–new differences do not isolate the constructor fix.

Budgets, evaluation and inference. BabyAI retains the nominal 1.5M-frame budget and evaluates at actual frame counts 0, $250{,}880$, $501{,}760$, $750{,}080$, $1{,}000{,}960$ and $1{,}500{,}160$. Each checkpoint uses four newly specified evaluation blocks, 100 episodes per block for each of the five existing held-out task kinds, with equal kind weights. CrafText uses 4,194,304 frames and nine equally spaced checkpoints, four newly specified blocks, two episodes per canonical goal per block and a 1,000-step horizon. All 57 goals are retained. These new seeds do not constitute a newly held-out language or task-family split. Evaluation queries only the greedy policy.

The primary metric is time-normalized trapezoidal macro-success AUC, including the initial policy. There are exactly three primary contrasts: Aligned minus Reassigned (same top action) for BabyAI, Qwen-reference CrafText and expert-reference CrafText. Policy-seed and evaluation-block contrasts are averaged within each upstream source. Two-sided $98\frac{1}{3}\%$ Student-t intervals over five source means (four degrees of freedom) apply a Bonferroni correction to this three- comparison family. Unrestricted-donor, Anchor-only, retained-policy and endpoint contrasts use explicitly descriptive, unadjusted 95% intervals.

Selection and limits. This hypothesis was developed after earlier studies, including failed LLM transfer pilots. Its sample counts, outcomes, analysis and exact execution files were frozen before the new full-budget outcomes. The five upstream sources are reused. Variability in completed runs warns of low precision for small BabyAI effects. There is no addition of seeds after observing intervals, promotion of an endpoint over AUC or omission of unsuccessful goals. Technical checkpoint re-evaluations and independent arithmetic checks are not additional replications.

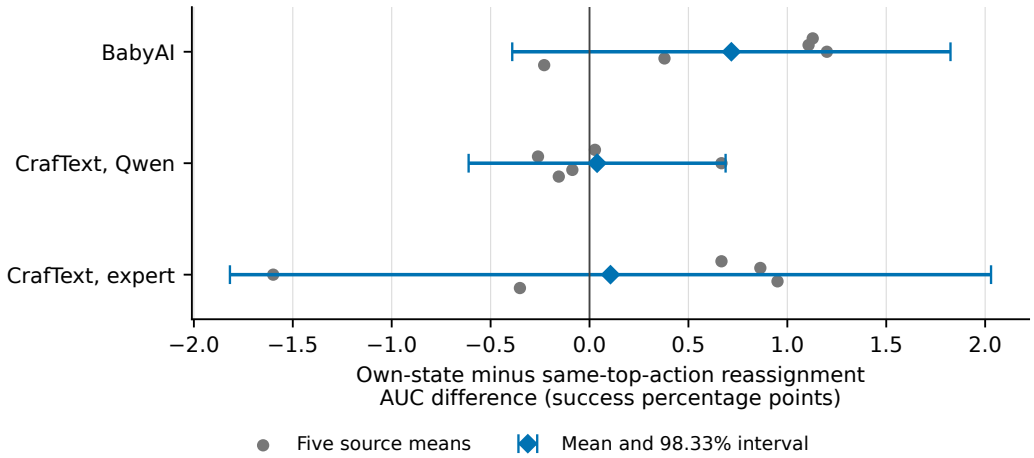


Figure 5: Primary Aligned-minus-same-top-action AUC contrasts. Points show the five source means; intervals are the governing three-comparison-adjusted source intervals, not intervals over individual policy seeds or episodes.

Primary outcomes. All 165 policies completed the frozen training and evaluation inventory. Table 4 and Figure 5 show the three primary comparisons. The BabyAI interval permits an own-state advantage of up to 1.82 success points; Section 5.2 interprets why the remaining uncertainty does not establish top-action sufficiency.

Other PPO controls and retained policies. All secondary AUC and endpoint contrasts appear in Tables 23 and 24. They use unadjusted 95% intervals. In BabyAI, Aligned minus unrestricted Reassigned is $+5.13$ AUC points $[3.21, 7.06]$; the same-top-action arm minus unrestricted Reassigned is $+4.41$ $[2.93, 5.90]$. Both contrasts are positive in all five sources and all 20 policy pairs. Thus unrestricted target reassignment changes learning substantially in this cohort. Because the operators also differ in their donor distributions and downstream trajectories, this is not an additive causal decomposition into top-action and other effects.

CrafText Aligned minus unrestricted Reassigned is $+0.65$ AUC points $[-0.18, 1.48]$ with Qwen labels and $+1.44$ $[0.32, 2.56]$ with expert labels. Qwen Aligned is below Anchor only by -1.73 points $[-3.29, -0.17]$ and retained BC by -5.11 $[-8.11, -2.11]$. Expert Aligned minus Anchor only is $+0.12$ $[-1.05, 1.29]$. Positive reference-versus-reassignment contrasts therefore do not establish improvement over the no-reference recipe. The unsmoothed curves in Figure 6 retain the early declines; later recovery does not replace the time-averaged primary outcome.

Was the constrained operator active? The process audit includes every policy, with no selection by performance. During updates with positive reference coefficients, same-top-action reassignment moves 98.69% of BabyAI rows and 99.99%/99.93% of Qwen/expert CrafText rows, averaged over policy seeds within each source and then sources. The supplied logit vector changes on 78.31%, 99.99% and 99.93% of rows, respectively, while recipient argmax changes remain zero. BabyAI counts any unequal coordinate; CrafText requires a maximum absolute difference above 10^{-6}. These are not matched cross-suite perturbation doses, and a common logit shift can leave softmax probabilities unchanged. Unrestricted reassignment changes recipient argmax on 63.48%, 55.52% and 77.18% of rows in the same three strata. These checks verify donor movement and changes in supplied logits while preserving recipient argmax. They do not by themselves quantify changes in normalized probabilities or establish a mediator, equivalent gradients or equal KL pressure.

Verification and compute. An independent implementation reconstructed all 1,150,920 evaluation episode records, all primary and secondary intervals, 180 saved BabyAI rollout/donor random-stream replays, 35,160 BabyAI update invariants and 6,720 CrafText update invariants. Re-evaluation of a saved checkpoint is a technical check, not another independent policy. The 165 training/evaluation jobs allocated 83.84 L40S GPU-hours, excluding earlier reference fitting, implementation pilots and

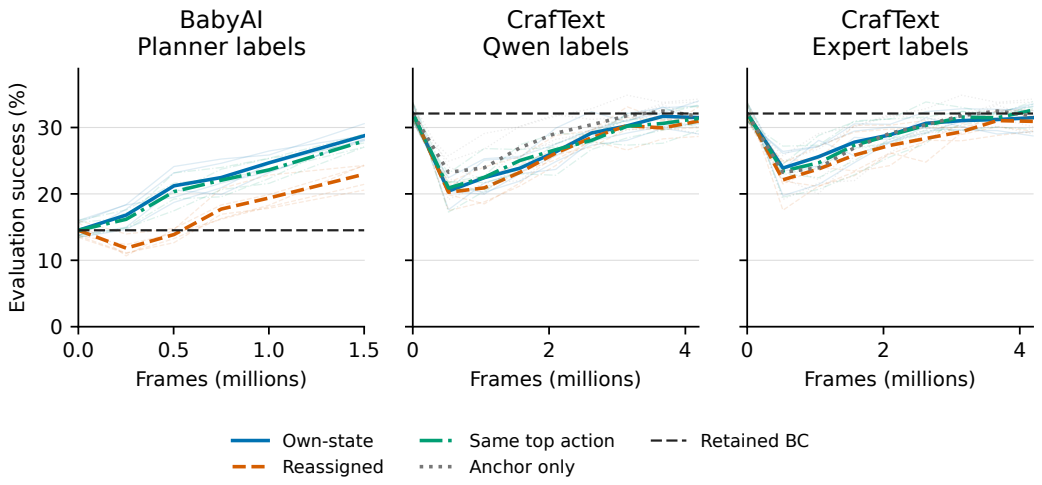


Figure 6: Complete-study learning curves without smoothing. Thick curves average the five source means, with faint source curves. CrafText panels share the same 15 Anchor-only policies; horizontal lines show the starting BC policy and vertical lines the reference-KL cutoff. Panels share a success scale, not task difficulty, instruction generalization or frame budget.

Table 23: Descriptive normalized learning AUC contrasts in success percentage points. All intervals are unadjusted 95% source intervals and do not replace the primary family. The same 15 CrafText Anchor-only policies are shared across both reference-source comparisons. Reassigned denotes the unrestricted donor control under the new common-eligibility rule.

Setting	Contrast	Mean	Interval
BabyAI	Aligned minus Reassigned	$+5.13$	$[+3.21, +7.06]$
BabyAI	Same top action minus Reassigned	$+4.41$	$[+2.93, +5.90]$
BabyAI	Aligned minus Retained BC	$+7.73$	$[+5.87, +9.58]$
BabyAI	Reassigned minus Retained BC	$+2.60$	$[+2.03, +3.16]$
BabyAI	Same top action minus Retained BC	$+7.01$	$[+5.77, +8.25]$
CrafText, Qwen	Aligned minus Reassigned	$+0.65$	$[-0.18, +1.48]$
CrafText, Qwen	Same top action minus Reassigned	$+0.61$	$[-0.08, +1.30]$
CrafText, Qwen	Aligned minus Retained BC	-5.11	$[-8.11, -2.11]$
CrafText, Qwen	Reassigned minus Retained BC	-5.76	$[-8.81, -2.71]$
CrafText, Qwen	Same top action minus Retained BC	-5.15	$[-7.96, -2.33]$
CrafText, expert	Aligned minus Reassigned	$+1.44$	$[+0.32, +2.56]$
CrafText, expert	Same top action minus Reassigned	$+1.34$	$[+0.62, +2.05]$
CrafText, expert	Aligned minus Retained BC	-3.25	$[-5.90, -0.61]$
CrafText, expert	Reassigned minus Retained BC	-4.70	$[-7.87, -1.52]$
CrafText, expert	Same top action minus Retained BC	-3.36	$[-7.06, +0.34]$
CrafText, Qwen	Aligned minus Anchor only	-1.73	$[-3.29, -0.17]$
CrafText, Qwen	Reassigned minus Anchor only	-2.38	$[-3.40, -1.37]$
CrafText, Qwen	Same top action minus Anchor only	-1.77	$[-3.34, -0.21]$
CrafText, expert	Aligned minus Anchor only	$+0.12$	$[-1.05, +1.29]$
CrafText, expert	Reassigned minus Anchor only	-1.32	$[-2.15, -0.49]$
CrafText, expert	Same top action minus Anchor only	$+0.02$	$[-0.66, +0.69]$

CPU verification. The supplement includes the fixed protocol, per-policy outcomes, contrasts and provenance hashes; omitted model payloads are identified separately.

Table 24: Descriptive final-checkpoint success contrasts in success percentage points. All intervals are unadjusted 95% source intervals and do not replace the primary family. The same 15 CrafText Anchor-only policies are shared across both reference-source comparisons. Reassigned denotes the unrestricted donor control under the new common-eligibility rule.

Setting	Contrast	Mean	Interval
BabyAI	Aligned minus Same top action	$+0.78$	$[-0.42, +1.98]$
BabyAI	Aligned minus Reassigned	$+5.79$	$[+3.38, +8.19]$
BabyAI	Same top action minus Reassigned	$+5.01$	$[+2.11, +7.90]$
BabyAI	Aligned minus Retained BC	$+14.23$	$[+12.67, +15.79]$
BabyAI	Reassigned minus Retained BC	$+8.45$	$[+6.58, +10.32]$
BabyAI	Same top action minus Retained BC	$+13.45$	$[+11.50, +15.40]$
CrafText, Qwen	Aligned minus Same top action	$+0.12$	$[-0.47, +0.71]$
CrafText, Qwen	Aligned minus Reassigned	$+0.53$	$[-0.89, +1.95]$
CrafText, Qwen	Same top action minus Reassigned	$+0.41$	$[-1.04, +1.86]$
CrafText, Qwen	Aligned minus Retained BC	-0.61	$[-3.48, +2.25]$
CrafText, Qwen	Reassigned minus Retained BC	-1.14	$[-3.53, +1.25]$
CrafText, Qwen	Same top action minus Retained BC	-0.73	$[-3.17, +1.70]$
CrafText, expert	Aligned minus Same top action	-1.14	$[-3.58, +1.30]$
CrafText, expert	Aligned minus Reassigned	$+0.56$	$[-0.68, +1.80]$
CrafText, expert	Same top action minus Reassigned	$+1.70$	$[+0.05, +3.35]$
CrafText, expert	Aligned minus Retained BC	-0.63	$[-2.40, +1.15]$
CrafText, expert	Reassigned minus Retained BC	-1.18	$[-2.31, -0.05]$
CrafText, expert	Same top action minus Retained BC	$+0.51$	$[-1.79, +2.82]$
CrafText, Qwen	Aligned minus Anchor only	-0.23	$[-2.60, +2.13]$
CrafText, Qwen	Reassigned minus Anchor only	-0.76	$[-3.80, +2.28]$
CrafText, Qwen	Same top action minus Anchor only	-0.35	$[-2.64, +1.94]$
CrafText, expert	Aligned minus Anchor only	-0.25	$[-2.22, +1.72]$
CrafText, expert	Reassigned minus Anchor only	-0.80	$[-2.41, +0.80]$
CrafText, expert	Same top action minus Anchor only	$+0.89$	$[-0.86, +2.64]$

H Direct Qwen-Label References on Fixed CrafText States

Question and chronology. The earlier sampling-priority study (Appendix N) changes sequence-sampling priorities, while retaining expert action targets. This follow-up instead trains the reference to predict Qwen's chosen actions. It asks whether the reference-label source changes learning when fitting states, supervised positions, architecture and optimization budget are matched. The study was specified after the earlier construction results and a bounded autonomous-Qwen development pilot. The preselected history-based pilot failed its expansion criteria, so autonomous collection did not expand. The present study therefore uses fixed expert-visited states, not autonomous LLM demonstrations. Its design was fixed before these reference fits and PPO outcomes; it is an outcome-aware follow-up, not an independently preregistered domain test.

Inputs, targets and splits. We reuse the archived current-observation Qwen3-32B annotations described in Appendices C.5 and N, with no new LLM queries. Qwen receives the canonical instruction, deterministic text derived from the current symbolic observation, public rules and the 17-action code map. It supplies one hard action label with greedy decoding and thinking disabled, not a normalized action distribution. The expert target is the argmax of its stored action logits on the same row. Qwen is stateless in this interface; the expert that generated the stored logits uses recurrent context. Matching queried rows does not make these teacher information histories identical.

For each of the five existing expert/data/BC sources, the original 192 training environment streams are partitioned into 144 reference-training and 48 reference-validation streams by a fixed seeded permutation. All 64 saved chunks of a stream remain in the same partition. Each chunk contains 64 time steps; labels are supplied only at $\mathcal{P} = \{0, 8, 16, 24, 32, 40, 48, 56\}$. This gives 73,728 supervised training positions and 24,576 validation positions per source. The original corpus's validation and test streams are not used. These new splits are for reference fitting; they are not newly held-out data for the unchanged BC initializer or a test of unseen instructions. Environment streams and fitting chunks should not be mistaken for complete episodes.

Both reference models receive all 64 observations, recorded reset flags and per-step instruction tokens. Their recurrent carry starts at zero at each fitting window. Only the eight annotated positions contribute to the fitting objective. Write $p_\psi = \mathrm{softmax}(r_\psi)$ for reference action probabilities. For teacher $\tau \in \{Q, E\}$ (Qwen or expert), with B sequences in a minibatch, the matched hard-label loss is

$$L_\tau(\psi) = -\frac{1}{B|\mathcal{P}|} \sum_{j=1}^{B} \sum_{t \in \mathcal{P}} \log p_\psi(a^\tau_{j,t} \mid o_{j,0:t}, m_{j,0:t}, \mathrm{resets}_{j,0:t}). \tag{23}$$

Unannotated observations still provide recurrent context; they are not filled with expert labels in the Qwen arm. Training uniformly shuffles sequences without replacement for exactly ten epochs with batch size 64, learning rate 3×10^{-4} and gradient-norm cap 1. Paired teachers share the initialization seed and optimizer recipe. Each keeps the lowest own-validation-cross-entropy checkpoint within this same budget. There is no early stopping or additional fit selected after PPO. The ten fits each execute 1,440 updates. The soft expert targets and replacement sampling of the earlier sampling-priority study are not reused as this hard-label comparison's matched expert control.

Online conditions and evaluation. For each source, three paired PPO seeds train six conditions: Qwen Aligned, Qwen online-calibrated Reassigned, expert Aligned, expert online-calibrated Reassigned, Uniform target and Anchor only. All 90 policies are newly trained; the source's expert-derived BC initializer and demonstration anchor remain unchanged. PPO retains the original 4,194,304-frame recipe. Its reference coefficient begins at .05 and reaches zero halfway through training; the .01 anchor continues. The Uniform-target arm uses the same reference coefficient schedule; Anchor only has no reference KL. Reassignment uses the existing CrafText donor operator and detached online coefficient calibration, which does not match local gradients.

CrafText's demonstration anchor applies hard expert-label cross-entropy to all 64 positions of each sampled sequence, not just the eight reference-fitting positions. An anchor minibatch contains 16 sequences drawn from the fixed 1,024-sequence pool. Its recurrent carry is initialized to zero at the start of each window, then propagated through its observations, instructions and recorded resets. Thus Eq. 4 for BabyAI's static examples is not a claim that CrafText resets memory at every supervised position.

The policy alone is evaluated greedily at nine equally spaced checkpoints on four fixed seed blocks, each containing two episodes per canonical instruction and a 1,000-step horizon. Seeds are newly specified for this follow-up and shared by the compared policies; the 57 instructions are not new. Retained BC is the initial policy evaluated on these same blocks, one policy per source rather than three new upstream replications. Checkpoint re-evaluations on the same seeds are verification, not additional experimental observations.

Inference and scope. The two primary contrasts are Qwen Aligned minus expert Aligned and Qwen Aligned minus its calibrated Reassigned control, using time-normalized success AUC. For each contrast, three paired policy-seed differences are averaged within source; a two-sided 97.5% Student t interval is computed over the five source means, with four degrees of freedom. This is a two-comparison Bonferroni rule, not a claim of distribution-free coverage with five sources. Comparisons against Uniform, Anchor only and retained BC, endpoint differences, the expert assignment contrast and the difference of assignment contrasts remain descriptive secondary analyses. All five sources and all planned policy cells are retained regardless of their outcome. The frozen manifest records seeds, hashes and technical-retry rules.

This tests actual Qwen-label reference fitting on expert-visited states while holding expert-based initialization and anchoring fixed. It does not test an expert-free pipeline, independently manipulate instruction semantics, or make the fresh planner-labelled BabyAI study an LLM-distillation experiment.

H.1 Development feasibility and branch selection

Before fitting direct-label references, we tested whether Qwen could supply autonomous trajectories in the pinned CrafText interface. There were 57 canonical instructions, two episodes per instruction, and a 1,000-step horizon. Both Qwen variants used temperature zero, thinking disabled and one constrained action token. The current-observation variant received the observation and public rules;

Table 25: Fixed autonomous-Qwen development pilot. All variants use the same 114 initial episodes. A successful goal has at least one successful episode. The history variant was designated primary before outcomes; no winner selection or population-level LLM capability claim follows.

Agent	Successful episodes	Success	Successful goals
Uniform random	$3/114$	2.63%	$3/57$
Qwen, current observation (diagnostic)	$8/114$	7.02%	$5/57$
Qwen, history of eight (primary)	$4/114$	3.51%	$3/57$

the preselected primary variant also received eight preceding observable summaries and actions. Neither received expert actions, hidden state, rewards or goal-checker outputs. The action and episode seeds were fixed before the pilot.

Expansion required the primary variant to achieve at least 10% macro success, at least five percentage points above the paired random baseline, and success on at least six different goals, with complete trace coverage. These are development feasibility criteria, not a significance test. The history variant failed all three performance/coverage thresholds (Table 25); the diagnostic current-observation variant was not substituted as a new primary candidate. No autonomous corpus was collected for the final comparison. All outcomes, including unsuccessful episodes, remain in the record. Exact action/seed replay recovered float32 observations and matched all original prompt hashes; the original lower-precision stored traces remain separate. This reconstruction does not constitute additional teacher trials or policy training.

H.2 Direct-label learning-curve and endpoint results

Table 5 presents the two primary AUC comparisons. Qwen Aligned minus expert Aligned is $-.01264$, with 97.5% interval $[-.01754, -.00773]$; all five paired source means are negative. Qwen Aligned minus its calibrated Reassigned control is $+.00520$, with interval $[-.00082, +.01121]$. The additional benefit of aligned Qwen targets remains unresolved despite five positive source estimates. The sample AUC ordering of the label sources therefore differs from the final-checkpoint ordering, where Qwen Aligned minus expert Aligned is $+.01623$ with descriptive, unadjusted 95% interval $[+.00044, +.03202]$. Every source mean has this sample ranking reversal; the final-checkpoint interval does not replace the prespecified primary comparison.

Tables 26, 27 and 28 retain all source means and secondary contrasts. For each source ℓ, let $A_{Q,\ell}$ and $R_{Q,\ell}$ be its scores under Qwen Aligned and calibrated Reassigned, and define $A_{E,\ell}$ and $R_{E,\ell}$ analogously for expert labels. Each score uses the same metric (normalized AUC or endpoint success), averaging paired policy seeds before source-level inference. The expert assignment contrast is $d_{E,\ell} = A_{E,\ell} - R_{E,\ell}$; the difference of assignment contrasts is

$$d_{\mathrm{int},\ell} = (A_{Q,\ell} - R_{Q,\ell}) - (A_{E,\ell} - R_{E,\ell}).$$

A positive value means a larger assignment benefit for Qwen labels, not higher absolute Qwen performance. Secondary-contrast tables report 100 times the source-mean contrasts and intervals, in percentage points. The AUC interaction is $-.00122$ with unadjusted 95% interval $[-.00803, +.00560]$, leaving its direction unresolved. Its endpoint counterpart is $+.02354$ $[+.01448, +.03260]$. Different inferential conclusions at different metrics are not interchangeable evidence for one primary hypothesis.

The actual starting BC has mean success .29737 on these fresh episode blocks. Qwen Aligned has mean AUC .27056 and endpoint .32602; expert Aligned has mean AUC .28320 and endpoint .30980. Qwen's retained-BC differences are $-.02681$ AUC (descriptive 95% interval $[-.05181, -.00181]$) and $+.02865$ endpoint ($[+.01042, +.04689]$). Its time-averaged benefit over Uniform and Anchor only remains unresolved. Endpoint recovery does not identify what caused it: the reference loss has stopped, but demonstration anchoring continues.

Figure 7 shows the primary contrasts and source means; Figure 8 shows all conditions without smoothing. The complete inventory and separate block-level arithmetic checks passed before interpretation. Saved-checkpoint re-evaluation also passed for all 90 policies, covering 3,240 checkpoint/block records and 369,360 episode-checkpoint outcomes. These use the original seeds and repeated initial

Table 26: Direct-label CrafText study: all source means. Each trained condition averages three paired policy seeds. Q and E denote Qwen and expert hard labels; A and R denote Aligned and online-calibrated Reassigned. U is Uniform target and H Anchor only. Values are success fractions; retained BC is one shared initial policy per source.

Source	A_Q	R_Q	A_E	R_E	U	H	BC
Normalized learning AUC							
4	0.2729	0.2713	0.2839	0.2896	0.2917	0.2887	0.2939
5	0.2729	0.2623	0.2817	0.2679	0.2642	0.2562	0.3114
6	0.2887	0.2809	0.3042	0.2973	0.2743	0.3019	0.2873
7	0.2483	0.2455	0.2645	0.2573	0.2433	0.2654	0.3004
8	0.2700	0.2668	0.2816	0.2717	0.2618	0.2678	0.2939
Final-checkpoint success							
4	0.3209	0.3099	0.3063	0.3238	0.3319	0.3158	0.2939
5	0.3363	0.3173	0.3304	0.3275	0.3187	0.3121	0.3114
6	0.3253	0.2917	0.3231	0.3063	0.2939	0.3246	0.2873
7	0.3077	0.2931	0.2778	0.2961	0.2741	0.2734	0.3004
8	0.3399	0.3209	0.3114	0.3158	0.3019	0.2990	0.2939

Table 27: Secondary direct-label comparisons in normalized learning AUC, expressed in success percentage points. All intervals are descriptive, unadjusted 95% intervals over five paired source means; they do not replace the primary rule.

Contrast	Mean	Interval	Level
$A_E - R_E$ (expert assignment)	$+0.64$	$[-0.26, +1.55]$	95%
$(A_Q - R_Q) - (A_E - R_E)$	-0.12	$[-0.80, +0.56]$	95%
Qwen Aligned minus Uniform	$+0.35$	$[-1.25, +1.96]$	95%
Qwen Aligned minus Anchor only	-0.54	$[-2.35, +1.27]$	95%
expert Aligned minus Uniform	$+1.61$	$[-0.14, +3.37]$	95%
expert Aligned minus Anchor only	$+0.72$	$[-0.81, +2.26]$	95%
Qwen Aligned minus retained BC	-2.68	$[-5.18, -0.18]$	95%
expert Aligned minus retained BC	-1.42	$[-3.98, +1.15]$	95%

states; they are verification, not additional independent observations. Count-based interval reconstruction differs from the saved-block calculation by at most 6.5×10^{-9} in success fraction. The inferential sample remains five source means.

Allocated computation. On L40S GPUs, the ten reference fits used 0.24 allocated GPU-hours and the 90 PPO policies used 21.40; saved-checkpoint verification used another 14.98. Development and GPU software checks used 3.19, giving 39.80 GPU-hours for this bounded follow-up study. This measures allocation, not GPU utilization. It excludes earlier expert/data/BC construction and the preceding annotation/ sampling study, whose computation is reported separately in Table 43. Existing Qwen annotations are reused; 39.80 GPU-hours is not a complete end-to-end cost for producing them.

Table 28: Secondary direct-label comparisons in final-checkpoint success, expressed in success percentage points. All intervals are descriptive, unadjusted 95% intervals over five paired source means; they do not replace the primary rule.

Contrast	Mean	Interval	Level
Qwen Aligned minus expert Aligned	$+1.62$	$[+0.04, +3.20]$	95%
Qwen Aligned minus Qwen calibrated Reassigned	$+1.94$	$[+0.88, +3.01]$	95%
$A_E - R_E$ (expert assignment)	-0.41	$[-2.24, +1.42]$	95%
$(A_Q - R_Q) - (A_E - R_E)$	$+2.35$	$[+1.45, +3.26]$	95%
Qwen Aligned minus Uniform	$+2.19$	$[-0.28, +4.67]$	95%
Qwen Aligned minus Anchor only	$+2.11$	$[-0.09, +4.30]$	95%
expert Aligned minus Uniform	$+0.57$	$[-1.90, +3.04]$	95%
expert Aligned minus Anchor only	$+0.48$	$[-0.88, +1.85]$	95%
Qwen Aligned minus retained BC	$+2.87$	$[+1.04, +4.69]$	95%
expert Aligned minus retained BC	$+1.24$	$[-1.43, +3.91]$	95%

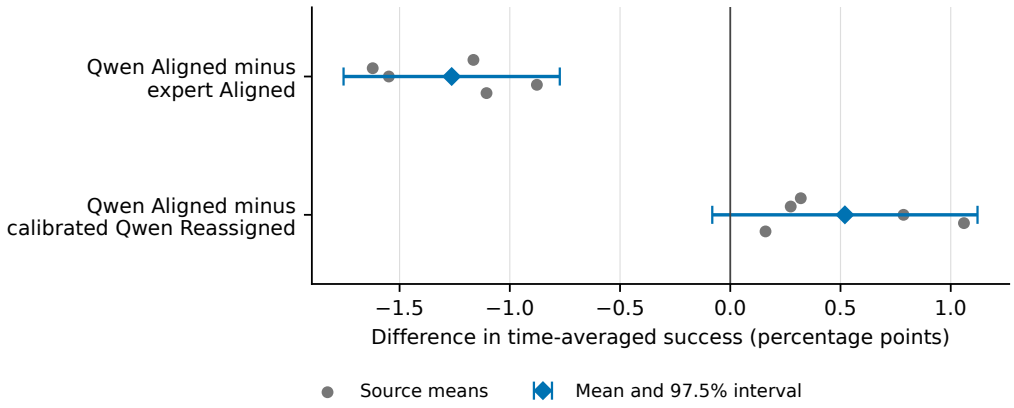


Figure 7: Direct-label CrafText primary comparisons. Grey points are five source means, each averaging three paired policy seeds. Diamonds and bars show the mean and governing 97.5% source-level interval. Values are success percentage points averaged over the learning curve.

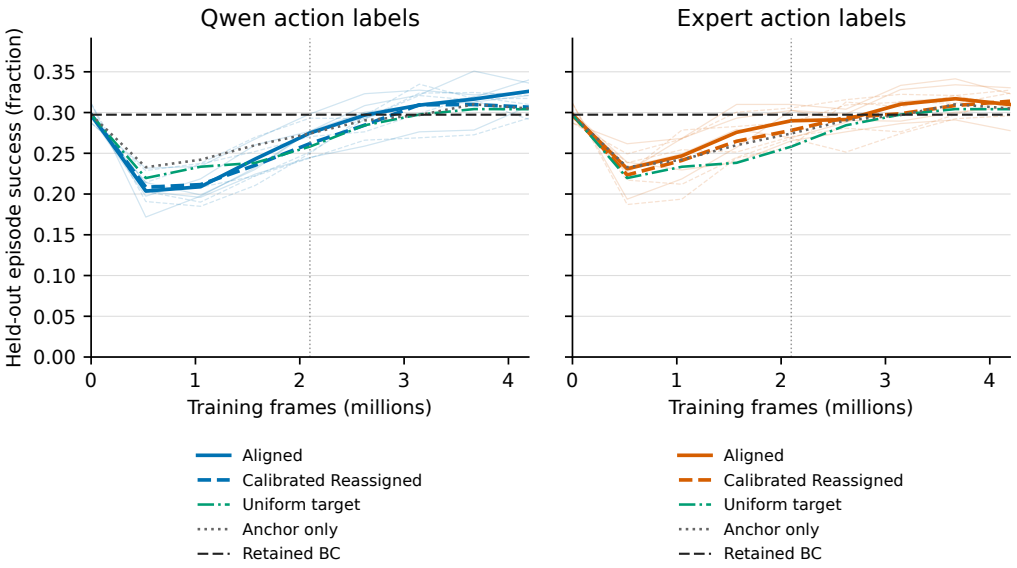


Figure 8: Matched direct-label learning curves. Thick lines average five source means; faint reference-condition lines show source means. Both panels include the same Uniform, Anchor-only and retained-BC controls. The vertical line marks reference-KL cutoff; the demonstration anchor continues. Success is a fraction; differences in the main text are percentage points. No curve is smoothed.

I EXPERT–QWEN PROBABILITY-MIXTURE STUDY

This protocol was fixed after the complete mixture-development result, before constructing the additional expert source or observing new annotation, reference fitting, or PPO outcomes. It tests whether Qwen contributes useful, appropriately assigned information when combined with an expert. It does not test an expert-free pipeline, novel instruction families, or a unique language-grounding mechanism. The development source (number 4) is excluded. Sources 5–8 reuse existing expert/data/BC artifacts; source 9 is newly constructed. New reference fits and policy seeds do not make the four reused datasets fresh.

Source construction and failure handling. Source 9 uses the recurrent FiLM expert specified in Appendix C.4. Training tasks are sampled using per-goal success from the final evaluations of three earlier 250M-frame experts (record IDs 1–3; seeds 61001–61003), all available before this source was constructed. These use the baseline recurrent architecture in Appendix C.2, not the FiLM expert, with the PPO budget and optimizer of Appendix C.4 and the benchmark instruction-reset sampler rather than failure-weighted sampling. Their evaluations set weights; their policies do not initialize source 9. For goal $g \in \{0, \dots, 56\}$, average those three successes:

$$b_g = \tfrac{1}{3}\sum_{j=1}^{3} s_{jg}, \qquad w_g = .2 + 1 - b_g, \qquad \widehat{w}_g = \min\{w_g, 4\min_h w_h\}, \qquad p_g = \widehat{w}_g / \sum_h \widehat{w}_h.$$

At each training reset, draw the instruction from p; explicit evaluation instructions override this draw. Thus less successful tasks are sampled more often, with a maximum weight ratio of four. The supplement records all 57 goal indices, input successes and normalized probabilities, together with the three input evaluations and sampler implementation. The expert trains five segments of 763 updates with seed 110093. Its quality rule requires success at least .30, an improvement over random of at least .25, at least 34 goals with nonzero success, at least 12 supported actions, donor eligibility at least .99, and actual training wall time at most two hours. No replacement source is permitted. Collection uses seed 210093, 256 environments, and 64 chunks of 64 steps. BC uses seed 310093, at most ten epochs, 64 sequences per batch, learning rate 3×10^{-4}, gradient clipping at one, and the original validation-based selection with patience three. Its evaluation seed is 510093. No new BC-success threshold is imposed. A failed source gate or invalid/incomplete BC leaves the five-source confirmation incomplete, rather than reducing its primary sample.

Qwen inputs and queried positions. Each source retains 192 original training-environment streams, each with 64 chunks. Of these streams, 144 supply reference-fitting sequences and 48 supply validation sequences; original validation/test streams are excluded. Sources 5–8 retain their existing partition, while source 9 uses the same split rule with seed 9140009. Qwen is queried at steps 0, 16, 32 and 48 of every retained 64-step sequence: 49,152 queries per source and 245,760 overall. No filtering by confidence, expert agreement, action, reward or successful goal is allowed.

The common prompt in Appendix C.5 contains the current symbolic observation, correct canonical instruction, public environment rules and all 17 action codes. It contains neither expert actions nor privileged hidden state, history, rewards, future observations or success feedback. Qwen3-32B uses bfloat16, two-device tensor parallelism, a 4,096-token context, batches of 128 and a one-token output restricted to the action codes. Temperature is zero. Prefix caching and V1 asynchronous multiprocessing are disabled, as in the audited probability pilot; the inference seed is 9190101. All 17 processed action log probabilities are retained and normalized. Every prompt hash, input position, action token and probability vector is independently checked before reference fitting.

Four matched reference targets. For each queried row i, let Δ^{16} denote the probability simplex over 17 actions and let $p_i^E \in \Delta^{16}$ be the native-temperature softmax of the recurrent expert logits at that exact position, and $p_i^Q \in \Delta^{16}$ the Qwen action probabilities, as in Eq. 6. Let $U(a) = 1/17$. Four target sets are fit with identical inputs and budgets:

$$p_i^E, \qquad h_i = \tfrac{1}{2}p_i^E + \tfrac{1}{2}p_i^Q, \qquad b_i = \tfrac{1}{2}p_i^E + \tfrac{1}{2}U, \qquad \widetilde{h}_i = \tfrac{1}{2}p_i^E + \tfrac{1}{2}p^Q_{\sigma^Q(i)}. \tag{24}$$

Here σ^Q is one fixed random single cycle within each exact-instruction and fitting-split group, with NumPy generator seed $9204000 + s$ for source s. Singleton groups retain self and are reported; other

rows have no fixed points. The last target keeps every expert component own-state while reassigning only Qwen probabilities. It preserves the group's Qwen-vector multiset, not its mixture-vector multiset, target entropy, gradients or fitted-reference outputs. Expert and Qwen probabilities are stored and normalized in float32. An independent reconstruction must pass before any reference fit begins.

All four references per source use the same recurrent architecture, initial parameters, minibatch order, 64-step context and four-position supervision mask. For any target $v_{j,t}$ from Eq. 24, the loss is

$$L_{\mathrm{soft}}(\psi) = -\frac{1}{B|\mathcal{P}|}\sum_{j=1}^{B}\sum_{t\in\mathcal{P}}\sum_{a\in\mathcal{A}} v_{j,t}(a)\log p_\psi(a \mid o_{j,0:t}, m_{j,0:t}, \mathrm{resets}_{j,0:t}), \quad \mathcal{P}=\{0,16,32,48\}.$$

Here B counts windows and $p_\psi = \mathrm{softmax}(r_\psi)$; every window has four labels. Other positions contribute recurrent context but zero loss weight. Replacing v by a one-hot action label gives Eq. 23 with that study's eight-position mask. Soft-label cross-entropy is minimized for exactly ten epochs (1,440 updates), using 64 sequences per batch, learning rate 3×10^{-4}, clipping at one, and seed $9201000 + s$. The final tenth-epoch checkpoint is used; validation is descriptive, not a selection criterion. Twenty reference fits are required. Qwen remains stateless while the original expert is recurrent: matching rows and fitting budgets does not equate teacher information or memory.

Seven online conditions. Every source trains three paired policy seeds for each condition: expert–Qwen mixture Aligned; expert–Qwen mixture Reassigned; expert Aligned; expert-Uniform Aligned; expert-reassigned-Qwen Aligned; Uniform target; and Anchor only. The first two use the reference fit to h_i, whereas the next three use p_i^E, b_i, and $\widetilde{h}_i$. Uniform uses an action-uniform target and Anchor only has zero reference-KL coefficient. The loaded mixture reference in those two controls supplies no state-dependent target information. Expert BC and demonstration anchoring remain fixed across all seven arms within a source.

Online mixture Reassigned applies the trainer's unrestricted seeded random single cycle within exact-instruction rollout groups. This differs from the fixed offline Qwen-component donor map σ^Q. The trainer draws random tie breakers for each update. Neither reassignment operation is a uniform draw from all possible derangements. No online KL coefficient calibration is used in this confirmation.

PPO and evaluation. Only the final scalar value head of each original full BC initializer is zeroed; actor parameters and shared features are unchanged. Original and standardized actors must produce identical evaluation episodes before PPO. Each of the 105 policies uses 1,024 environments, 64 steps, eight segments of eight updates, four PPO epochs, eight minibatches, learning rate 2×10^{-4}, $\gamma = .99$, generalized-advantage parameter .8, clipping .2, entropy coefficient .01, value coefficient .5 and gradient clipping at one. The reference coefficient starts at .05 and reaches zero halfway through 4,194,304 frames. The .01 demonstration anchor continues, with batches of 16 sequences from a 1,024-sequence pool. Policy seeds are $9201100 + 100s + k$ for $k \in \{1, 2, 3\}$. The framework and loss are those in Eq. 3.

Nine fixed checkpoints, including initialization, are evaluated on all 57 canonical goals. Each uses two blocks, two episodes per goal per block and a 1,000-step horizon. Block seeds are $9202000 + s$ and $9302003 + s$, shared across arms and policy seeds within source. Retained BC is evaluated once per source on those same blocks; its reuse does not create three independent BC policies. The complete record contains 215,460 policy-checkpoint episode observations and 1,140 retained-BC episodes. Repeated checkpoint evaluation is not additional independent training evidence.

Fixed inference and interpretation. The primary outcome is trapezoidal time-normalized macro-success AUC over all nine checkpoints, equally weighting all goals and blocks. The six primary contrasts subtract expert Aligned, expert-Uniform Aligned, expert-reassigned-Qwen Aligned, mixture Reassigned, Anchor only and retained BC from mixture Aligned. Paired differences are first averaged over three policy seeds within each source. Two-sided Student-t intervals use the five source means, four degrees of freedom, and Bonferroni confidence $1 - .05/6 = 99.1667\%$. The fixed joint rule requires all six lower bounds above zero, mean advantages at least .01 over expert, mixture Reassigned and Anchor only, and at least .005 over expert-Uniform. These magnitude benchmarks and the sample size were fixed before outcomes; they do not guarantee adequate power at five sources.

Uniform comparisons, endpoints, per-goal results and any nominal intervals are secondary and never substitute for a failed primary. Every source, arm, seed and checkpoint is required; missing cells cannot yield a passed joint decision. Earlier unsuccessful studies remain reported. No extra seeds, new mixture weights, alternative prompts, task selection or endpoint promotion follows from these confirmation outcomes.

I.1 Complete confirmation results

All five sources, 20 reference fits and 105 policies completed under the fixed protocol; no policy retry, extra seed, source replacement or target change was needed. Source 9 passed the original expert gate with .54474 success, 44 goals with nonzero success, .99995 donor eligibility and 1.456 training hours. All actor-preservation checks and agreement checks against the existing evaluator passed before PPO.

Tables 29 and 30 report every condition and all six primary contrasts. Figure 9 shows complete learning curves. The mixture's mean AUC is .30183, compared with .30742 for expert-only and .31020 for expert–Uniform. The corresponding differences are negative in all five source means, but their governing intervals include zero: neither population inferiority nor equivalence is established. Their upper bounds, .00506 and .00423, are nevertheless below the predefined .01 and .005 magnitude benchmarks. These intervals exclude those target advantages under this study's inferential rule; the limitation is not simply an unexamined opportunity to add seeds.

The Qwen-component comparison is $-.00397$ with interval $[-.01237, +.00444]$. Thus appropriately assigning Qwen before fitting is not established as better than reassigning Qwen while retaining own-state expert targets. The online whole-mixture assignment difference is $+.00325$ $[-.02413, +.03063]$. Against Anchor only the difference is $-.00247$ $[-.02593, +.02100]$; against retained BC it is $+.01060$ $[-.03063, +.05183]$. All six positive-lower-bound checks and all four mean magnitude checks fail. The predefined joint scientific decision is therefore negative, despite successful execution and record verification.

Table 29: Complete expert–Qwen mixture confirmation: five sources and three paired policy seeds per source. All entries are success fractions; AUC is time-normalized. Four upstream sources are reused and one is new; the development source is excluded. Endpoint success is secondary.

Condition	Mean AUC	Mean endpoint	AUC minus BC
Expert–Qwen mixture Aligned	0.30183	0.31930	+0.01060
Expert–Qwen mixture Reassigned	0.29857	0.32515	+0.00735
Expert Aligned	0.30742	0.32749	+0.01619
Expert-Uniform Aligned	0.31020	0.32924	+0.01897
Expert-Reassigned-Qwen Aligned	0.30579	0.32135	+0.01457
Uniform target	0.28403	0.32573	-0.00720
Anchor only	0.30429	0.32836	+0.01307
Retained BC	0.29123	0.29123	0.00000

Table 30: Six fixed primary contrasts, expert–Qwen mixture Aligned minus each comparator. Intervals are 99.1667% Student-t intervals over five source means (four degrees of freedom); Bonferroni adjustment applies to this study's six contrasts, not the entire adaptive sequence of studies.

Comparator	Mean difference	Governing interval	Positive sources
Expert Aligned	-0.00559	[-0.01624, +0.00506]	0/5
Expert-Uniform Aligned	-0.00837	[-0.02097, +0.00423]	0/5
Expert-Reassigned-Qwen Aligned	-0.00397	[-0.01237, +0.00444]	1/5
Expert–Qwen mixture Reassigned	+0.00325	[-0.02413, +0.03063]	3/5
Anchor only	-0.00247	[-0.02593, +0.02100]	2/5
Retained BC	+0.01060	[-0.03063, +0.05183]	4/5

The earlier source-4 pilot remains development evidence; the five-source confirmation did not reproduce its specified target-sized gains. Endpoints, individual goals and the Uniform-target comparison do not replace the failed primary outcomes. The experiment does not establish universal LLM failure, instruction irrelevance or a unique explanation for the pilot-to-confirmation difference. In particular,

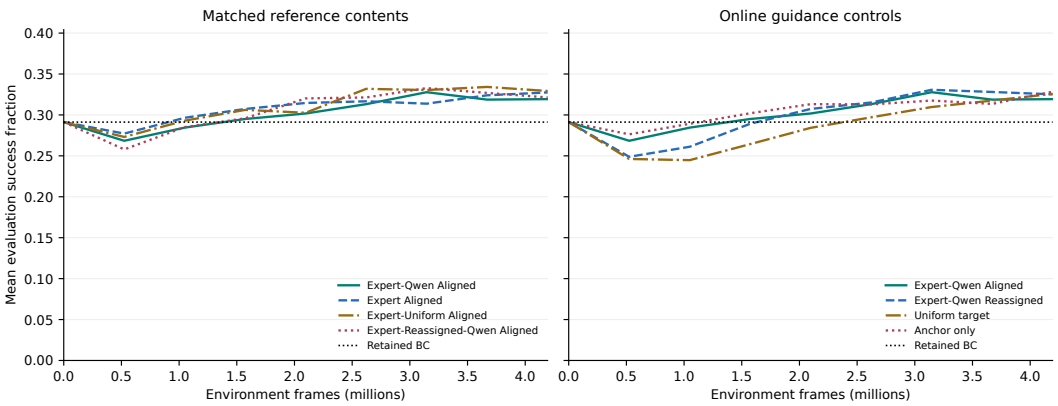


Figure 9: Complete confirmation trajectories, equally averaging five source means after three policy seeds per source. All 57 goals and both evaluation blocks are retained. "Expert-Qwen" in the legends denotes the expert–Qwen probability mixture; its Aligned curve appears in both panels. Governing uncertainty concerns paired AUC differences in Table 30, not these descriptive curves.

expert state coverage, initialization and anchoring remain present, and Qwen has less temporal context than the expert. The matched expert-only reference is fitted on the same sparse positions and budget, not the original expert-reference recipe using every available position.

Independent reconstruction and preservation. The target audit reconstructed all 983,040 probability vectors across the four target modes and five sources, including the exact float32 arithmetic and Qwen-component donor groups. Every prompt and queried position was checked. After training, separate arithmetic reconstructed all 105 learning curves and six source-level intervals from 215,460 saved policy episode observations and 1,140 retained-BC observations. The supplement provides these records and identifies omitted checkpoints, corpora and external dependencies by hash. This is saved-record reconstruction, not independent retraining.

J Frozen-BC Reference in CraftText

Question and evidence status. Does separately fitting a reference improve on using the actual starting BC policy as the reference? This differs from retaining BC without online training. The following baseline addition was specified after inspecting the completed mixture and BC-budget studies. It reuses all 15 expert-Aligned, all 15 expert–Qwen-mixture-Aligned and all 15 Anchor-only policies from the mixture study, rather than treating those known outcomes as new independent evidence. All five training-data sources and their three original PPO seeds are included; the development source is excluded. The earlier mixture study's seven arms and failed confirmation rule remain unchanged.

Exactly copied policy reference. For each source, the reference parameters are a byte-identical copy of the actual standardized starting BC model (its scalar value head is zeroed without changing actor parameters). The policy and reference use separate recurrent states, with the same episode-reset convention. The reference is frozen at its native temperature of one. No new labels, BC fitting, reference fitting, temperature search or coefficient calibration is used. The 15 new policies retain the original observation and instruction interface, expert anchor data, optimizer, policy/evaluation seeds, 4.194304M-frame budget, nine checkpoints, 57 canonical goals, two evaluation blocks and two episodes per goal per block. The reference coefficient starts at .05 and reaches zero halfway through training; the .01 demonstration anchor continues.

This is a comparison of available training recipes, not an isolated test of architecture or equal reference-fitting budgets. The separately fitted expert and mixture references use the original sparse-position fitting procedure; the starting BC was fit on the full original BC data. Consequently, any difference must be attributed to the tested recipes rather than uniquely to LLM knowledge or reference architecture.

Table 31: All four CrafText baseline recipes, five sources and three paired seeds per recipe. Only the 15 PPO policies using the Frozen-BC reference are newly trained; the other 45 outcomes are reused from the mixture study. The reference is a frozen copy of each policy's starting BC model. AUC and endpoint use success fractions. Retained BC is the starting policy deployed without PPO; its shared mean success is 0.29123.

Recipe	AUC	Endpoint	AUC − retained BC
Expert Aligned	0.30742	0.32749	0.01619
Expert–Qwen mixture	0.30183	0.31930	0.01060
Anchor only	0.30429	0.32836	0.01307
Frozen-BC reference	0.32432	0.34415	0.03310

Table 32: All three fixed primary AUC comparisons. Governing two-sided $98\frac{1}{3}\%$ source-level t intervals use five source means (df 4), adjusted for three comparisons. Outcomes of the reused comparators were known before this baseline addition.

Contrast	Mean	Governing interval	Positive sources
Expert − Frozen-BC	-0.01690	$[-0.02898, -0.00483]$	0/5
Mixture − Frozen-BC	-0.02250	$[-0.03398, -0.01102]$	0/5
Frozen-BC − Anchor only	$+0.02003$	$[+0.00245, +0.03761]$	5/5

Fixed analysis. Three primary contrasts use normalized success AUC: expert Aligned minus Frozen-BC reference, expert–Qwen mixture Aligned minus Frozen-BC reference, and Frozen-BC reference minus Anchor only. For each contrast, the three paired seed differences are averaged within each source. Inference uses the five source means, two-sided $98\frac{1}{3}\%$ Student-t intervals with four degrees of freedom, and a Bonferroni correction for these three comparisons. Superiority of the first-named recipe requires a positive governing lower bound; an entirely negative interval favors the second recipe. Endpoints, per-goal results and retained-policy comparisons are descriptive. No extra sources/seeds, selected instructions or alternate endpoints are substituted after outcomes.

Technical checks. CPU checks verify exact reference/initializer parameter identity and identical synthetic recurrent forward outputs for all five sources. A separately logged, short-horizon GPU smoke test checks execution and record integrity only; its scores do not select scientific settings. The scientific protocol is frozen before that test, and execution/evaluation/analysis code is frozen before the 15 full-budget policies. The full audit requires every new policy and all 45 predetermined comparator records, including complete raw episode arrays, initial-state agreement, checkpoint hashes and loss schedules. A second arithmetic implementation independently reconstructs all 60 curves, three source-level intervals and every per-goal AUC.

J.1 Complete baseline results

All 15 new policies completed without retries. The frozen starting BC reference outperforms each designated training comparator: expert Aligned minus Frozen-BC reference is $-.01690$ AUC with governing interval $[-.02898, -.00483]$; mixture Aligned minus Frozen-BC reference is $-.02250$ $[-.03398, -.01102]$; and Frozen-BC reference minus Anchor only is $+.02003$ $[+.00245, +.03761]$. Every source mean favors Frozen-BC reference in each comparison. Table 32 retains the three originally specified directions, rather than changing the analysis family after observing which recipe performed best.

The Frozen-BC-reference recipe reaches mean AUC .32432, compared with .30742 for the separately fitted expert reference, .30183 for the expert–Qwen mixture and .30429 for Anchor only. Its endpoint is .34415; all endpoints remain secondary. The actual starting BC success is .29123, so the observed mean Frozen-BC AUC difference from retaining that policy is $+.03310$; this is a descriptive comparison, not an additional primary test. Table 31 reports all four recipes and Table 33 reports every source.

Table 33: Every source mean AUC in the Frozen-BC baseline addition. Each cell averages the three original policy seeds. Development source 4 is excluded.

Source	Expert	Mixture	Anchor only	Frozen-BC
5	0.32392	0.31762	0.31862	0.34348
6	0.31652	0.30309	0.32209	0.32447
7	0.31296	0.30793	0.31296	0.33891
8	0.30601	0.30336	0.30090	0.32410
9	0.27769	0.27714	0.26690	0.29066

This positive result supports using the already available BC policy as a reference in this recipe. Offline fitting coverage differs between BC and the separately fitted references, and no Frozen-BC reassignment arm is run; the comparison evaluates available recipes rather than isolating coverage or assignment as the explanation.

Verification and preserved records. The complete audit checks 60 policy records (15 new and 45 predetermined reused), 123,120 policy-checkpoint episode observations, 1,140 retained-policy episodes and 3,840 update records. Each new policy's initial episode array matches the paired original starting policy. An independent implementation reconstructs all 60 curves, all three primary source-level intervals and 3,420 per-goal AUC records from the raw episodes. The supplement includes the new and reused comparator episode records; model payloads, demonstration-anchor data and other omitted dependencies are identified by hash.

K BC Fitting-Budget Confirmation in CraftText

This follow-up tests whether the BC fitting budget moderates the benefit of the same Qwen reference. Its development pilot appears in Table 45; that source is excluded here. The confirmation was initially deferred in favor of the more promising mixture pilot. After the complete mixture confirmation failed its fixed rule, we selected this distinct, previously specified hypothesis. The scientific protocol, source set, sample size, evaluation and three primary contrasts were fixed before new fitting or policy outcomes. This is an outcome-informed continuation, not a preregistered sequence of studies or independent domain replication. All five expert/data sources (5–9) are reused, including source 9 constructed for the mixture study. No source was included or excluded by mixture outcome.

BC budgets and retained policies. Within each source, a single new BC fit uses all 192 original training streams and all 64 positions in each 64-step sequence. There are 12,288 training sequences, 64 per minibatch, and 192 updates per epoch. Learning rate is 3×10^{-4}, gradient clipping is one, and the fitting seed is $9253000 + s$ for source s. The exact epoch-one and epoch-ten checkpoints (192 and 1,920 updates) are taken from the same uninterrupted optimization prefix. Validation is recorded but does not select either checkpoint. Both checkpoints have their final scalar value-head weights and bias set to zero; the actor and shared features are unchanged. Raw and standardized checkpoints are preserved, and their evaluation episodes must match before PPO.

The low/high distinction is therefore a fitting-budget intervention, not a controlled intervention on competence alone: actor distributions, shared representations and subsequent optimization can differ. Each actual starting BC is evaluated and retained as a comparator. The separately available original full BC is also evaluated on exactly the same blocks. Its presence matters when a weak initial policy improves yet remains below a policy already available before this experiment. New epoch-ten BC and original full BC are distinct fitted artifacts, not interchangeable labels.

Fixed hard-label Qwen reference. Each source uses one frozen Qwen reference shared by both BC budgets and all online conditions. Sources 5–8 retain the original hard-label references from the matched direct-label study. Source 9 receives new labels using that same eight-position recipe, not the later four-position probability-mixture recipe. Qwen3-32B sees the current symbolic observation, canonical goal, public rules and 17 action codes. It sees neither expert actions nor rewards, history, future observations or hidden simulator information. State coverage, BC actions and anchor labels still come from the expert.

Source 9 is queried at steps $0, 8, 16, 24, 32, 40, 48, 56$ of every retained training sequence: 98,304 queries. All queries are retained, without confidence, agreement, action, reward or successful-goal filtering. The original hard-label interface uses bfloat16, two-device tensor parallelism, a 4,096-token context, at most 128 sequences and 8,192 batched tokens, eager execution, .85 GPU-memory utilization, enabled prefix caching, seed 9132026, temperature zero and a one-token action-code output. Only the selected token log probability is requested; this is not the complete-probability interface used in the mixture. Input identities, mission IDs, expert modes and every prompt hash are audited.

The original reference split partitions 192 training streams into 144 fitting and 48 validation streams (seed 9140009 for source 9); original validation/test streams are excluded. Full recurrent contexts and episode resets are retained, but the loss is masked to the eight queried positions. Reference fitting uses hard-label cross-entropy, ten epochs, 1,440 updates, batch size 64, learning rate 3×10^{-4} and clipping at one. The seed is $9141000 + s$; the lowest validation-loss checkpoint is selected with the original 10^{-6} improvement rule. The unchanged references for sources 5–8 used this same recipe. Qwen is stateless while the fitted reference is recurrent; no equal-memory or expert-free inference follows.

Online conditions and evaluation. Each BC budget trains Aligned, unrestricted same-instruction random-cycle Reassigned, and Anchor only, with two paired policy seeds per source. All $5 \times 2 \times 2 \times 3 = 60$ policies are required. Reassignment preserves the within-rollout target-vector multiset; singleton groups retain self. It does not match later visitation, state-wise KL or gradients. No KL calibration is used. Anchor only has zero reference coefficient but retains supervised anchoring. The loaded reference supplies no KL information in that arm.

The existing PPO recipe is unchanged: 1,024 environments, 64 steps, eight segments of eight updates, four optimization epochs, eight minibatches, learning rate 2×10^{-4}, discount .99, generalized-advantage parameter .8, clipping .2, entropy coefficient .01, value coefficient .5 and gradient clipping at one. The .05 reference coefficient reaches zero halfway through 4,194,304 environment steps; the .01 anchor continues, using batches of 16 sequences from a 1,024-sequence pool. Policy seeds are $9254000 + 100s + k$, $k \in \{1, 2\}$, paired across both budgets and all arms.

Nine fixed checkpoints include initialization and the endpoint. Each uses all 57 canonical goals, two blocks, two episodes per goal per block and a 1,000-step horizon. Block seeds are $9255000 + s$ and $9355003 + s$. All arms and policy seeds share these blocks within a source. Each retained policy is evaluated once per source, not counted again as an independent policy for each PPO seed. There are 123,120 policy-checkpoint episode observations and 3,420 retained-policy episodes. Initial actor equivalence and the instrumented evaluator's agreement with the original evaluator must pass before any online policy outcome.

Three fixed primary contrasts. Let A^c_{ske} denote normalized all-goal success AUC for source s, policy seed k, BC epoch $e \in \{1, 10\}$ and condition c. Define

$$D_{s,e} = \frac{1}{2} \sum_{k=1}^{2} (A^{\text{Aligned}}_{ske} - A^{\text{Reassigned}}_{ske}), \tag{25}$$

$$B_s = \frac{1}{2} \sum_{k=1}^{2} (A^{\text{Aligned}}_{sk1} - A^{\text{Anchor}}_{sk1}). \tag{26}$$

The three primary source-level quantities are $D_{s,1}$, B_s, and $D_{s,1} - D_{s,10}$. AUC is trapezoidal and time-normalized over all nine checkpoints, equally weighting all 57 goals and both blocks. Student-t intervals use the five source means, four degrees of freedom and Bonferroni confidence $1 - .05/3 = 98\frac{1}{3}\%$. The fixed directional rule requires all three lower bounds above zero. This directional rule does not establish practical superiority or generalizability; effect magnitudes and retained-policy comparisons are reported separately.

Endpoints, high-budget Aligned-minus-Anchor, all per-goal outcomes and both types of retained BC are secondary. No pilot pooling, added seed, selected instruction, reduced horizon, changed budget, target or endpoint is allowed. An incomplete 60-cell inventory cannot pass the five-source rule. A BC-budget interaction would not by itself identify initial competence as its mechanism.

K.1 COMPLETE CONFIRMATION RESULTS

All five sources and all 60 policies completed. Reducing BC fitting did not establish the predicted increase in the assignment benefit (Table 35). Low-budget Aligned minus Reassigned is $+.00891$ with interval $[-.00172, +.01954]$; all five source means are positive. Low-budget Aligned minus Anchor only is $+.00162$ $[-.01159, +.01483]$, with three positive source means. The low-minus-high assignment difference is $+.00148$ $[-.00601, +.00897]$, also with three positive source means. The observed high-budget assignment gap is $+.00743$. Neither the low-budget reference advantage over both controls nor its moderation by BC budget is established under the prespecified family.

Table 34 reports all six cells; Table 36 retains all source means. Figure 10 shows all learning curves and both retained-policy comparators. The low-budget initializer has success $.02544$; Aligned reaches AUC $.04751$, compared with $.04589$ for Anchor only. The low-budget gain over retaining its actual initializer is therefore not specific to reference guidance. Both remain far below the separately available original full BC at $.30965$. At the high budget, Aligned's AUC $.28671$ is below its actual initializer's $.29474$ on average, but the interval leaves the direction of this retention contrast unresolved. Its secondary endpoint is $.32500$. Table 37 reports all twelve retention contrasts with unadjusted 95% intervals.

Table 34: Complete BC-budget confirmation: five reused sources, two paired policy seeds, two BC budgets and three arms. All entries are success fractions. AUC is time-normalized over the fixed budget; endpoint success is secondary. The last two columns compare AUC with retaining the actual starting BC and the separately available original full BC.

BC epochs	Condition	AUC	Endpoint	Minus actual BC	Minus full BC
1	Qwen Aligned	0.04751	0.05395	+0.02207	-0.26214
1	Qwen Reassigned	0.03860	0.05263	+0.01316	-0.27105
1	Anchor only	0.04589	0.05877	+0.02045	-0.26376
10	Qwen Aligned	0.28671	0.32500	-0.00803	-0.02294
10	Qwen Reassigned	0.27928	0.32719	-0.01546	-0.03037
10	Anchor only	0.30921	0.33246	+0.01447	-0.00044

Table 35: Three fixed primary contrasts. Decision intervals are $98\frac{1}{3}\%$ Student-t intervals over five source means (four degrees of freedom). Bonferroni adjustment covers this study, not all studies; source signs are descriptive.

Contrast	Difference	Governing interval	Positive sources
Low: Aligned minus Reassigned	+0.00891	[-0.00172, +0.01954]	5/5
Low: Aligned minus Anchor only	+0.00162	[-0.01159, +0.01483]	3/5
Low minus high assignment gap	+0.00148	[-0.00601, +0.00897]	3/5

Table 36: All source-level AUC means, after averaging the two policy seeds. Retained-policy success is reused within a source, not counted as a separate policy per seed. Source 4 was used for development and is excluded from confirmation.

Source	BC epochs	Aligned	Reassigned	Anchor only	Actual BC	Original full BC
5	1	0.05839	0.04975	0.04578	0.01754	0.32895
5	10	0.32018	0.31181	0.32538	0.31140	0.32895
6	1	0.04756	0.03893	0.04249	0.03070	0.33333
6	10	0.30099	0.29592	0.33018	0.29825	0.33333
7	1	0.04852	0.03838	0.04838	0.02632	0.27632
7	10	0.28358	0.27029	0.30743	0.27632	0.27632
8	1	0.05236	0.03536	0.05592	0.04386	0.30263
8	10	0.27741	0.26809	0.31346	0.30702	0.30263
9	1	0.03070	0.03056	0.03687	0.00877	0.30702
9	10	0.25137	0.25027	0.26960	0.28070	0.30702

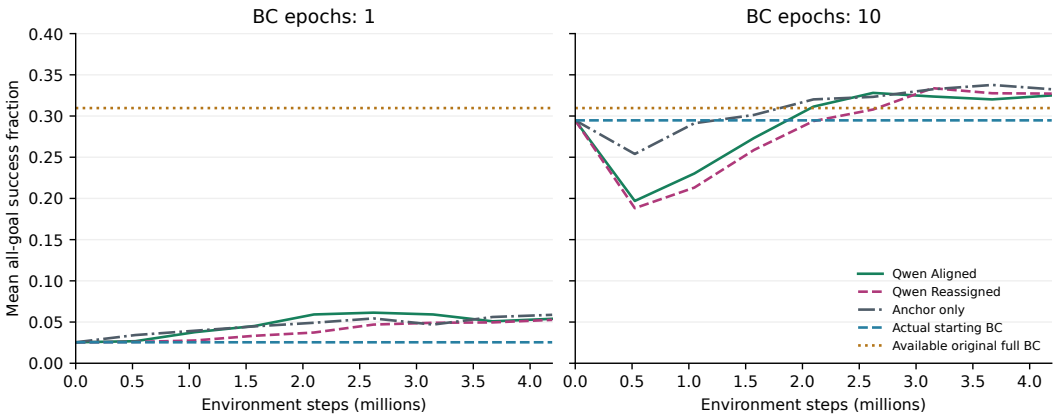


Figure 10: Complete BC-budget confirmation trajectories, equally averaging five source means after two policy seeds. All 57 goals and both evaluation blocks are retained; both panels share the success-fraction scale. Horizontal lines distinguish the actual initializer from the separately available original full BC. These are descriptive curves; governing uncertainty is in Table 35.

Table 37: Secondary AUC-minus-retained-policy contrasts, with descriptive unadjusted 95% source-level Student-t intervals. These intervals do not replace the primary family or its joint directional rule.

BC epochs	Condition	Comparator	Difference	Descriptive interval
1	Qwen Aligned	Actual BC	+0.02207	[+0.00732, +0.03681]
1	Qwen Aligned	Original full BC	-0.26214	[-0.29093, -0.23335]
1	Qwen Reassigned	Actual BC	+0.01316	[-0.00580, +0.03211]
1	Qwen Reassigned	Original full BC	-0.27105	[-0.29704, -0.24507]
1	Anchor only	Actual BC	+0.02045	[+0.01030, +0.03060]
1	Anchor only	Original full BC	-0.26376	[-0.29617, -0.23135]
10	Qwen Aligned	Actual BC	-0.00803	[-0.03249, +0.01642]
10	Qwen Aligned	Original full BC	-0.02294	[-0.05257, +0.00668]
10	Qwen Reassigned	Actual BC	-0.01546	[-0.03774, +0.00682]
10	Qwen Reassigned	Original full BC	-0.03037	[-0.05466, -0.00609]
10	Anchor only	Actual BC	+0.01447	[-0.00792, +0.03686]
10	Anchor only	Original full BC	-0.00044	[-0.03152, +0.03065]

The original source-4 pilot passed its development rule, but this larger study does not confirm that rule. It also does not establish that BC budget is irrelevant: the intervals allow effects in either direction, and changing the fitting prefix changes representations and optimization as well as initial success. The earlier calibrated Pong moderator and this uncalibrated CrafText comparison use different references and protocols. Together they do not identify a shared cross-domain explanation or a language-specific boundary. No source, seed, goal or endpoint was substituted after these outcomes.

Verification and preservation. All fifteen initial-actor checks were completed before the full-policy runs, including two resumed checks after a validation-job timeout. No actor, reference, evaluation seed or scientific setting changed. All ten PPO bundles completed without policy retries. The audit checked 3,840 updates, 123,120 policy-checkpoint episode observations and 3,420 retained-policy episodes. A separate implementation reconstructed all 60 curves, the three source-level intervals and the joint decision. The supplement includes the evaluation records, code and logs; omitted expert/data sources and model payloads are identified by hash.

L Additional BabyAI Assignment and Reference Controls

These studies address specific alternatives left open by the earlier results. Their protocols and executable hashes were fixed after the main-study results but before these follow-up runs. BabyAI

reuses the five existing offline pipelines and main-study Aligned policies. No source is selected or replaced using these outcomes. The separate Pong initialization study is reported in Appendix O.1.

L.1 BabyAI: changing donor choice while preserving targets

For an exact-instruction group $G_{u,m}$ of size $k \geq 2$, sample a uniform ordering $(j_1, \ldots, j_k)$ of its rows, then assign donor $\sigma(j_r) = j_{r+1}$, with $j_{k+1} = j_1$. A singleton keeps its target. This produces a uniform random *single cycle*, not a uniform draw over all permutations without fixed points. It preserves the complete target-vector multiset because every row donates exactly once. The deterministic control instead cycles in the stored environment-major row order.

The computation is:

1. Collect one 64-environment, 40-step rollout and compute every row's original fitted-reference logits. Group rows by their complete encoded instruction, not by task family.
2. Use a private CPU random generator, seeded by the paired policy seed plus 20,260,909, to sample one ordering per nonsingleton group. Its state advances across rollouts; it does not consume the learner's random-number stream.
3. Cycle through that ordering and copy each donor's already-computed logits to its recipient. Apply the same update-wide inverse temperature ρ_u as Aligned. Keep this mapping and these targets fixed throughout the PPO epochs for the rollout.
4. Keep the original BC initializer, anchor, PPO optimizer, coefficient schedule, training budget, and held-out evaluation fixed. Train four paired seeds for each of the five existing pipelines.
5. Log target-multiset checks and donor locations. For donors in the same vectorized environment, record signed time-index difference; separately count cross-environment donors and donors within the same contiguous episode segment.

The last check uses rollout reset masks to identify segments; it does not recover complete cross-rollout episode identities. Same-environment lags range from -39 to 39; moved rows cannot have lag zero in the same environment. Cross-environment donors have no meaningful temporal lag and are not placed in that histogram. Randomizing donors does not make states independent, nor does it preserve the later state distributions of separately trained policies. The original deterministic runs did not record donor lag; the new diagnostics must not be presented as measurements from those older runs. The logged changed-vector rate compares logits before temperature scaling; it is not a measurement of changed post-softmax probabilities. For example, at $\rho_u = 0$ all probability targets are uniform even when logits differ.

L.2 BabyAI: a frozen BC reference is not a retained BC policy

Both comparisons start from the same available behavior-cloned weights, but use them differently:

1. **Retained starting BC:** deploy the unchanged BC policy directly. It receives no PPO updates and supplies a constant performance curve.
2. **Frozen-BC reference:** make a separate frozen copy of those weights. Initialize a trainable PPO policy from the same weights, then query the frozen copy on the learner's rollout observations to construct KL targets.
3. Advance the reference's own recurrent memory along each environment's rollout, resetting it when that environment's episode ends. Do not substitute the learner's hidden state. Carry it across PPO rollout boundaries if the episode continues. Freeze model parameters, not this evolving memory.
4. Use the BC policy's native action probabilities at temperature one: $q^{\mathrm{BC}}_{u,i} = \pi_{\theta^{\mathrm{BC}}}(\cdot \mid o_{u,i}, m_{u,i}, h^{\mathrm{BC}}_{u,i})$. The reference is queried without drawing an unused action, so its queries do not alter the learner's action-sampling random stream.
5. Train with the same policy-to-reference KL coefficient ($\lambda_0 = .05$, cosine decay to zero at 0.75M frames), continuing anchor ($\mu = .2$), optimizer and 1.5M-frame budget. Only the trained PPO policy is evaluated after online training.

The fitted-reference treatment warms ρ_u from zero to two; the native BC reference does not use that fitted-reference-specific sharpness schedule. The two references also have different architectures and training objectives. Thus this is a practical reference-recipe comparison, not an isolated test of temperature, architecture, or whether any separately trained reference is necessary. No temperature is selected from these outcomes.

Evaluation and inferential units. Each new arm contains 20 policies: five reused pipelines and four paired seeds. All original evaluation blocks and checkpoint rules remain unchanged, with actual frame counts 0, 250,880, 501,760, 750,080, 1,000,960, and 1,500,160. Each checkpoint uses four blocks of 100 episodes per task family. The two extension comparisons are Aligned minus random-cycle Reassigned and Aligned minus Frozen-BC reference. Each uses a two-sided 97.5% t interval over five pipeline means, giving Bonferroni family coverage of at least 95% for this pair of comparisons. The positive-direction rule also requires all five pipeline effects to be positive. Report contrast-specific 95% intervals and crossed-bootstrap sensitivity intervals separately. The original BabyAI intervals remain 95% contrast-specific intervals; they are not retroactively relabelled as multiplicity-adjusted.

Retained-starting-BC analysis uses each pipeline's unchanged initializer on the same evaluation blocks. Reusing its score for four paired policy seeds does not create four independently trained BC policies. Endpoint differences and random-minus-deterministic differences are secondary, not further confirmations.

BabyAI results. All 40 new policies pass the fixed technical checks. A separate implementation recomputes the analysis from 2,420 evaluation files with 1,210,000 episode records, including the original controls; this is arithmetic verification, not independent training replication. Exact episode-count fractions resolve numerical ties. Table 38 reports the complete two-comparison family. Only the random-cycle AUC contrast satisfies its frozen directional rule.

Table 38: Additional BabyAI contrasts on five reused pipelines. AUC comparisons use the 97.5% intervals specified for their two-comparison family; endpoint and operator-order contrasts are secondary.

Aligned minus	AUC difference	95% interval	97.5% interval
Random-cycle Reassigned	+.0521	$[+.0185, +.0857]$	$[+.0098, +.0943]$
Frozen-BC reference	+.0253	$[+.0025, +.0481]$	$[-.0034, +.0540]$
Secondary endpoint differences; 95% intervals			
Aligned minus	Endpoint difference	95% interval	Positive pairs
Random-cycle Reassigned	+.0570	$[-.0044, +.1183]$	16/20 positive pairs
Frozen-BC reference	+.0405	$[+.0154, +.0656]$	18/20 positive pairs

The AUC pipeline effects are $(.0318, .0430, .0996, .0447, .0412)$ against random-cycle Reassigned and $(.0108, .0259, .0566, .0172, .0159)$ against Frozen-BC reference. There are respectively 20/20 and 17/20 positive paired policy effects. Random minus deterministic Reassigned has an interval spanning zero ($-.0058$, 95% interval $[-.0275, +.0159]$, 3/5 positive pipelines). Thus both donor operators yield assignment gaps, without establishing an ordering between operators or independent-state randomization. The 95% crossed-bootstrap sensitivity intervals are $[+.0327, +.0780]$ against random-cycle Reassigned and $[+.0112, +.0425]$ against Frozen-BC reference; they do not override the adjusted decisions.

Aligned's AUC is .2317, random-cycle Reassigned's .1796, and Frozen-BC reference's .2064; their endpoints are .2941, .2371, and .2536. The unchanged starting BC has mean success .1511. Aligned exceeds that static comparator by $+.0806$ AUC ($[+.0587, +.1025]$; 5/5 pipelines, 20/20 pairs). The two BC uses answer different questions: deploying the initializer is worse here, but Aligned's AUC superiority over using that initializer as a frozen reference has an adjusted interval spanning zero. A positive secondary endpoint contrast does not change that primary decision.

Figures 11–12 show the curves and intervention behavior. A same-episode segment spans only contiguous rows inside one rollout; that label is not a recovered global episode identity.

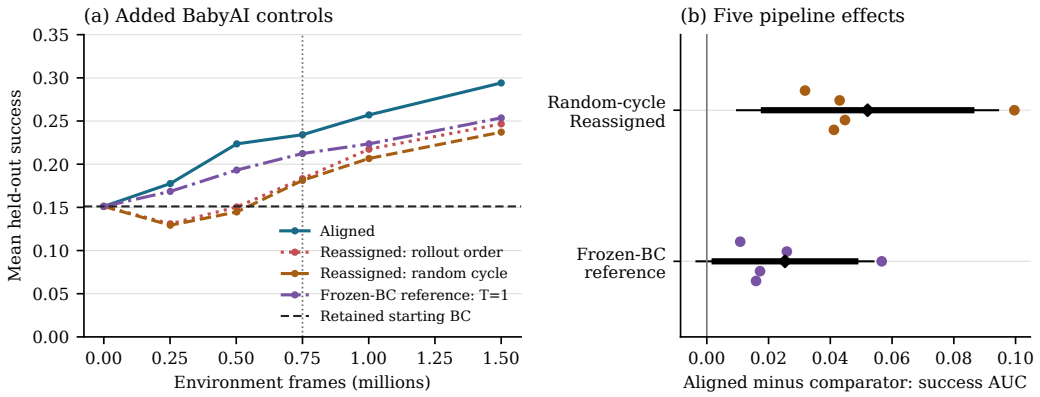


Figure 11: Additional BabyAI controls. Left: mean success and the static starting-BC comparator; the vertical line marks reference-loss cutoff. Right: colored points are the five pipeline AUC effects, diamonds their means, thick bars 95% intervals, and thin bars the governing 97.5% intervals.

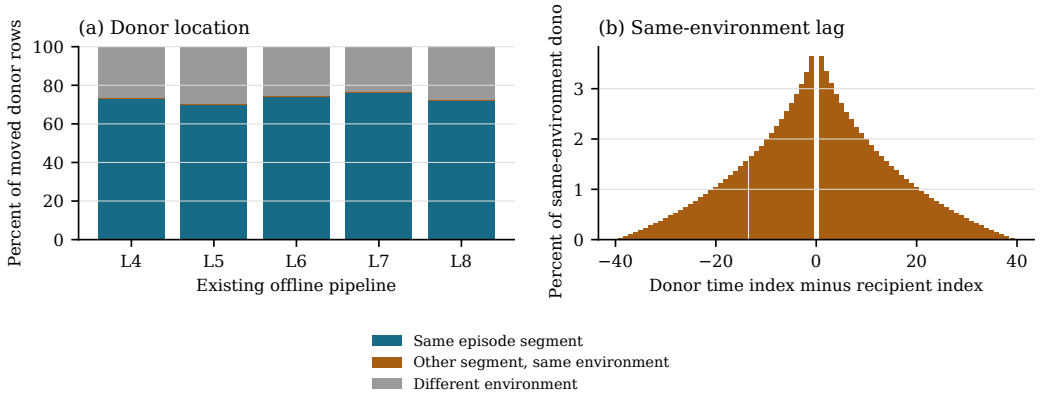


Figure 12: Random-cycle donor diagnostics over all 586 rollouts of each new policy, including rollouts after the KL cutoff. Left: donor location among moved rows. Right: signed lag conditional on the donor sharing the recipient's vectorized environment; cross-environment rows are excluded from this histogram.

M COEFFICIENT CALIBRATION

This section gives the calibration used in Section 3. Auxiliary-suite results are in Appendix O.1. Calibration controls change only Reassigned's KL coefficient. *Global calibration* sets $\lambda_0^{\text{global}} = \lambda_0 P_{\text{A}}/P_{\text{R}}$, where $P_{\text{A}}, P_{\text{R}}$ are condition means of run-averaged $\lambda_u \text{KL}$ in completed Aligned/Reassigned logs. All new calibrated policies use this fixed value and the original decay; their outcomes never update it.

Online coefficient calibration instead recomputes a multiplier for each CrafText update or auxiliary-suite minibatch $\mathcal{G}$. It scores both targets on the same rows under fixed scoring distribution $p_{d,\mathcal{G},i}$, where d indexes the suite (collection policy in CrafText, current policy elsewhere). Let $R_d(a, b)$ protect the ratio a/b: CrafText, HighwayEnv, and Pong floor the denominator at 10^{-8} and clip the result to suite-specific bounds; MiniHack instead aborts if $b \le 10^{-8}$ and otherwise uses the unclipped ratio. Let $\text{stopgrad}[v]$ keep value v but assign it zero derivative. For target family τ, define

$$\begin{aligned} \bar{D}^{\tau}_{d,\mathcal{G}} &= |\mathcal{G}|^{-1} \sum_{i \in \mathcal{G}} \text{KL}\big(p_{d,\mathcal{G},i} \,\|\, q^{\tau}_{u,i}\big)\,, \\ \kappa_{d,\mathcal{G}} &= \text{stopgrad}\Big[R_d\Big(\bar{D}^{\text{align}}_{d,\mathcal{G}}, \bar{D}^{\text{reassign}}_{d,\mathcal{G}}\Big)\Big]\,, \qquad \lambda^{\text{reassign}}_{u,\mathcal{G}} = \lambda_u \kappa_{d,\mathcal{G}}. \end{aligned} \tag{27}$$

When guards do not alter the ratio, $\lambda^{\text{reassign}}_{u,\mathcal{G}} \bar{D}^{\text{reassign}}_{d,\mathcal{G}} = \lambda_u \bar{D}^{\text{align}}_{d,\mathcal{G}}$. This matches one group-mean coefficient-weighted KL value, not raw or per-state KL, gradients, or visitation. The CrafText ratio is clipped to $[.01, 10]$ and scored under the collection policy; HighwayEnv/Pong use $[.05, 20]$ in their archived protocols. MiniHack aborts at the denominator guard. Appendix A gives both procedures step by step, and a numerical example. The scoring group $\mathcal{G}$ need not equal an instruction donor group $G_{u,m}$.

N MATCHED QWEN-ASSISTED CRAFTEXT REFERENCE CONSTRUCTION

Question and experimental unit. This extension asks whether using Qwen to prioritize expert-labelled training sequences changes the quality of a training-time action reference, and whether state assignment remains important under that construction. It uses the five existing CrafText sources in the selected configuration; it does not create five new environments or five new expert datasets. Each source supplies one fixed dataset and BC initializer. We fit two references per source and train three paired PPO seeds in each of four conditions, for ten reference fits and 60 PPO policies. The prompt and construction rule were fixed before these policy outcomes. The configuration itself was selected during the earlier CrafText study.

Qwen interface. Appendix C.5 gives the complete prompt, a recorded example and the mapping between CrafText studies. Qwen3-32B receives the canonical instruction and a deterministic textualization of the current 1,345-dimensional symbolic observation: the 7×9 local map, creatures, inventory, survival variables, facing direction, light, and sleep state. The prompt includes public game rules and identifies the tile in front using those same observation fields. It includes neither the expert action nor hidden simulator state, goal-checker outputs, or past observations. The policy and learned reference remain recurrent, so their available temporal context differs from this stateless Qwen annotation interface. A permitted-information interface does not imply that Qwen and the recurrent expert have identical information.

We decode one of 17 single-token action codes with temperature zero and thinking disabled, using the locally fingerprinted Qwen weights. Each response specifies one action, not a normalized 17-action distribution. The selected token's returned log-probability is diagnostic only and is not used for weighting. Invalid responses cause a technical failure rather than silently substituting the expert action. The released prompt, action mapping, model fingerprint and annotation hashes specify the interface. Greedy decoding fixes the decoding rule; it does not claim bitwise reproducibility across arbitrary inference software and hardware.

Fixed sequence priorities. Each source has 64 saved chunks of 64 steps and 192 training-environment streams, giving $N = 12{,}288$ training sequences. Validation and test streams are excluded from Qwen annotation and remain unweighted for checkpoint selection and reporting. For

sequence j, Qwen is queried at the eight positions $\mathcal{P} = \{0, 8, 16, 24, 32, 40, 48, 56\}$. Let $a^Q_{j,t}$ be its selected action and $a^E_{j,t}$ the argmax of the saved expert logits. The two sequence weights are

$$w^Q_j = 1 + \frac{1}{8} \sum_{t \in \mathcal{P}} \mathbf{1}\{a^Q_{j,t} \neq a^E_{j,t}\}, \qquad w^E_j = 1. \tag{28}$$

Thus disagreement increases sampling priority by a bounded factor in $[1, 2]$. Both constructions draw exactly N sequences with replacement in each epoch, with probability $w_j / \sum_k w_k$. There is no additional loss-side weighting. The annotation budget is 98,304 decisions per source, 491,520 in total. Qwen does not execute environment actions or provide the supervised reference targets.

For either construction, the reference minimizes mean soft-label cross-entropy on all 64 positions of sampled sequences:

$$L_{\text{fit}}(\psi) = -\frac{1}{64B} \sum_{j=1}^{B} \sum_{t=0}^{63} \sum_{a=1}^{17} p^E_{j,t,a} \log p^{\psi}_{j,t,a}, \qquad p^E_{j,t} = \text{softmax}(\ell^E_{j,t}). \tag{29}$$

Here $B = 64$ sampled sequences, ℓ^E denotes the expert logits, and p^{ψ} is the reference's action distribution. The recurrent state starts at zero for each sampled sequence and respects saved resets. The architecture, seed, Adam optimizer, learning rate 3×10^{-4}, gradient-norm cap 1, and validation selection are matched. Training uses at most ten epochs and stops after three epochs without validation improvement greater than 10^{-6}. The minimum-validation-loss checkpoint is retained.

The original CrafText reference fitter shuffled sequences without replacement. Consequently, its reference and downstream PPO policies are not the matched expert-only controls here: both reference constructions are refit with the same replacement sampler, and both sets of PPO policies are newly trained. BC and the online anchor are reused unchanged. This avoids attributing a change in the sampling algorithm to Qwen.

Four matched online conditions. Each fitted reference supplies either Aligned targets or online-calibrated Reassigned targets under the existing CrafText donor operator. All four cells reuse the source's BC initializer, paired PPO seeds, anchor data and evaluator. PPO uses 1,024 parallel environments, 64-step rollouts, 64 updates, four update epochs and eight minibatches, for 4,194,304 frames. The learning rate is 2×10^{-4}; the anchor coefficient is .01 throughout; and the reference KL coefficient starts at .05 and reaches zero halfway through training. The online Reassigned multiplier clips the rollout-collection-policy KL ratio to $[.01, 10]$ and is detached from differentiation. It does not match local gradients or identify the mechanism of a difference. Evaluation queries only the policy at nine fixed checkpoints, using four paired blocks, each with two episodes for each of the 57 canonical instructions and a 1,000-step horizon. The instruction set is shared with training: held-out evaluation refers to episode seeds, not new goals or paraphrases.

Prespecified contrasts and uncertainty. Let A_Q, R_Q, A_E, R_E denote normalized learning-curve AUC for Qwen-assisted Aligned/Reassigned and expert-only Aligned/Reassigned. The two primary contrasts are $A_Q - A_E$ and $A_Q - R_Q$. For each, we average the three paired policy-seed differences within source, then report the mean and two-sided 97.5% Student t interval over five source means (four degrees of freedom). These two intervals give Bonferroni familywise 95% coverage. Evaluation episodes and policy seeds do not increase the number of independent source replications. Endpoint success, retained-BC AUC and endpoint differences, the expert-only assignment contrast, and $(A_Q - R_Q) - (A_E - R_E)$ are secondary descriptive analyses; their nominal intervals do not replace the primary rule. All five sources and all completed prespecified seeds are reported, without selecting favorable goals.

Provenance and claim limits. This extension adapts the Qwen-based example weighting used in the online-Qwen BabyAI comparison (Appendix D.1), not every stage of that pipeline. It uses learned-expert rather than planner targets, clean rather than left/right-corrupted prompts, sequence rather than individual-row weights, 17 rather than six actions, and no Qwen changes to BC or online anchoring. It is not a fresh, uniformly matched two-domain LLM ablation. A favorable construction contrast would establish the effect of this Qwen-assisted reweighting recipe, not distinguish language knowledge from all other consequences of example reweighting. An assignment contrast would not establish that instruction grounding is the unique mechanism. Neither contrast alone implies improvement over retaining BC or generalization to paraphrases.

Table 39: Matched CrafText reference-construction study: five existing sources and three paired policy seeds per source. Differences are normalized learning AUC (success fractions); intervals are the governing 97.5% source-level intervals for two primary contrasts. Both reference families use replacement sampling.

Contrast	Mean	97.5% interval
Qwen-assisted minus expert-only Aligned	-0.0008	$[-0.0218, +0.0201]$
Qwen-assisted Aligned minus calibrated Reassigned	$+0.0109$	$[-0.0020, +0.0237]$

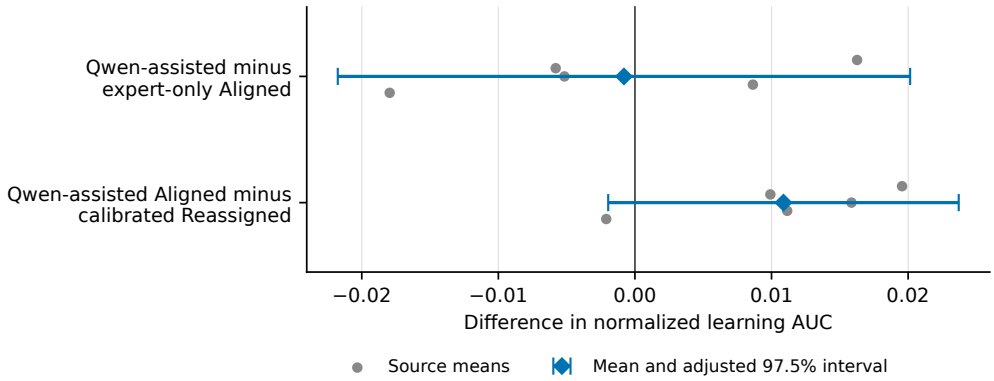


Figure 13: Both primary CrafText construction contrasts. Diamonds and horizontal bars show means and governing 97.5% source intervals; grey points show the five source means. Both primary comparisons remain inconclusive.

N.1 Results, diagnostics and execution

Table 39 reports both primary contrasts of the earlier sampling-priority study. It does not establish either an advantage from Qwen-based example weighting over expert-only fitting or an assignment benefit for the Qwen-weighted reference. Figure 13 shows those intervals with all five source means; Table 40 gives the four cells and retained BC for every source. The new expert-only reference's assignment contrast is $+.0129$ AUC with unadjusted 95% interval $[+.0054, +.0205]$. The construction-by-assignment interaction has interval $[-.0122, +.0081]$ around $-.0021$ (Table 41). Thus a positive unadjusted expert-only interval and a Qwen-assisted adjusted interval crossing zero do not establish that Qwen reduces assignment sensitivity.

Both Aligned constructions initially decline and later recover (Figure 14). Qwen-assisted Aligned minus retained BC is $-.0188$ AUC (unadjusted 95% interval $[-.0340, -.0036]$), whereas its endpoint difference is $+.0050$ with interval $[-.0057, +.0156]$. The endpoint does not establish superiority to retention. These secondary outcomes answer the retention question, not the primary construction and assignment questions.

Table 42 shows the annotation diagnostics. The weight-concentration measure $N_{\text{eff}} = (\sum_j w_j)^2 / \sum_j w_j^2$ is 98.8–99.0% of N. This describes a mild departure from uniform sampling, not how many unique sequences training actually visited or whether gradients were similar. Expert-mode agreement is 23.6–26.3%; it is not an action-correctness score. Neither these diagnostics nor the primary intervals identify why this particular construction did not demonstrate incremental benefit. In particular, the stateless annotation interface and bounded weighting remain bundled design choices.

All 491,520 annotation identities were checked against the saved expert modes and instruction IDs, and all sequence weights were independently reconstructed. A separate check reconstructed 384 fixed prompt hashes per source. All 60 policy records passed initial-parameter, evaluation-block, schedule, and donor invariance checks. An independent implementation reconstructed the AUCs and all ten contrast intervals from 2,160 saved evaluation-block records. These checks are not independent raw-episode evaluation or retraining. Table 43 reports the completed main-study allocation. No training run was repeated or selected according to its learning outcome.

Table 40: All five CrafText sources in the matched construction study. Each policy cell averages three paired seeds. A denotes Aligned and R online-calibrated Reassigned; subscripts E and Q denote expert-only and Qwen-assisted reference fitting. Values are success fractions. Retained BC is the same starting policy within each source, evaluated on the paired seed blocks.

Source	A_E	R_E	A_Q	R_Q	Retained BC
A. Normalized learning AUC					
4	0.3107	0.3025	0.2927	0.2948	0.3143
5	0.2697	0.2561	0.2783	0.2672	0.3165
6	0.3099	0.2867	0.3047	0.2888	0.3107
7	0.2934	0.2820	0.2876	0.2777	0.2997
8	0.2855	0.2770	0.3018	0.2822	0.3180
B. Final-checkpoint success					
4	0.3275	0.3268	0.3063	0.3355	0.3143
5	0.3289	0.3048	0.3326	0.3136	0.3165
6	0.3151	0.2946	0.3158	0.2990	0.3107
7	0.2968	0.3099	0.3063	0.3026	0.2997
8	0.2997	0.2946	0.3231	0.3275	0.3180

Table 41: All predefined secondary CrafText construction contrasts. Intervals are nominal 95% intervals over five source means, not multiplicity-adjusted primary decisions. Values are success fractions. A/R denote Aligned/calibrated Reassigned, and Q/E Qwen-assisted/expert-only reference fitting.

Contrast	Mean	95% interval	Positive sources
$A_E - R_E$ (AUC)	$+0.0129$	$[+0.0054, +0.0205]$	5/5
$(A_Q - R_Q) - (A_E - R_E)$ (AUC)	-0.0021	$[-0.0122, +0.0081]$	1/5
A_Q−retained BC (AUC)	-0.0188	$[-0.0340, -0.0036]$	0/5
A_E−retained BC (AUC)	-0.0180	$[-0.0435, +0.0074]$	0/5
$A_Q - A_E$ (endpoint)	$+0.0032$	$[-0.0169, +0.0233]$	4/5
$A_Q - R_Q$ (endpoint)	$+0.0012$	$[-0.0231, +0.0254]$	3/5
A_Q−retained BC (endpoint)	$+0.0050$	$[-0.0057, +0.0156]$	4/5
A_E−retained BC (endpoint)	$+0.0018$	$[-0.0144, +0.0179]$	3/5

Table 42: Qwen annotation diagnostics over 98,304 probes and 12,288 sequences per source. Agreement is with the recurrent expert's modal action, not environment-level action correctness. N_{eff} measures weight concentration, not unique data exposure.

Source	Agreement	Mean weight	Weight s.d.	N_{eff}/N
4	0.2627	1.7373	0.1874	0.9885
5	0.2570	1.7430	0.1924	0.9880
6	0.2389	1.7611	0.1774	0.9900
7	0.2517	1.7483	0.1812	0.9894
8	0.2361	1.7639	0.1784	0.9899

Table 43: Allocated GPU time for the matched CrafText study, including model startups. This is allocation rather than measured utilization, and excludes existing expert/data/BC construction, pilots, validation, paper builds and queue waits. All GPU tasks use L40S devices.

Stage	Tasks	Allocated GPU-hours
Qwen annotation (two GPUs/task)	20	14.70
Expert-only reference fits	5	0.25
Qwen-assisted reference fits	5	0.23
Expert-only reference PPO	30	7.22
Qwen-assisted reference PPO	30	7.18
Total GPU stages	90	29.57

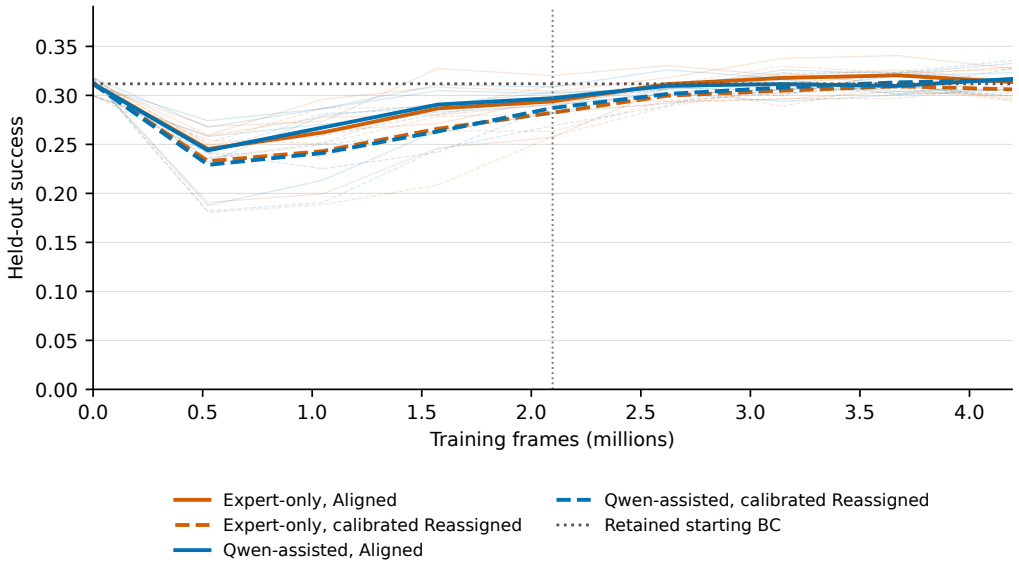


Figure 14: All four matched CrafText construction conditions. Thick lines average five source means, each over three paired policy seeds; faint lines show sources. The horizontal line is retained starting BC and the vertical line the KL cutoff. The anchor continues. Curves are unsmoothed; no outcome-based source selection is used.

O AUXILIARY SUITES AND DEVELOPMENT STUDIES

These studies evaluate training choices in non-language suites and candidate reference constructions. Their protocols and replication units differ from the main experiments. We retain favorable, mixed and failed outcomes, identify the relevant comparison locally, and keep their saved records in the supplementary material.

O.1 AUXILIARY SUITES AND CONFIGURATION DEPENDENCE

These studies use the separate recipes below, not the BabyAI or CrafText fitting defaults. They do not independently vary the presence of language. The HighwayEnv assignment study has 79 kinematic/validity inputs, nine actions, three pipelines and four seeds, 500k steps and return as outcome. MiniHack uses four binary map channels and task identity, eight compass actions, breadth-first-search labels, three pipelines and four seeds, and 500k transitions. Object-based Pong has four-frame object histories, six actions, two pipelines and four seeds, and 2M decisions. Reassignment is instruction/task-grouped for CrafText/MiniHack but rollout-wide for HighwayEnv/Pong, with online-calibrated coefficients.

HighwayEnv: teacher, collection and fitting. The environment is `highway-fast-v0` with a 40-second duration, one action per second and normalized environment rewards. Three experts use separate 128–128 tanh policy and value networks and train for 500,000 requested steps with 16 environments, 128-step rollouts, ten epochs, minibatch 1,024, learning rate 3×10^{-4}, discount .99, GAE .95, PPO clip .2, entropy coefficient .01, value coefficient .5 and gradient cap .5. The final-budget model, not a validation-selected checkpoint, labels 200,000 transitions by greedy actions. All collected rows are retained, including episodes ending in collision and the final partial episode. Episodes are shuffled into 80/10/10 train/validation/test groups. BC and reference are independent 79–128–128–9 tanh networks; both minimize hard-action cross-entropy, with shuffled rows without replacement, batches of 1,024, Adam at 3×10^{-4}, no weight decay and gradient cap 1. They retain the lowest validation cross-entropy, requiring improvement greater than 10^{-5}; fitting stops after six nonimproving epochs or 50 epochs. Neither has memory or a value branch. For source $j = 1, 2, 3$, expert, collection and split seeds are $83601 + 100j$, $83611 + 100j$ and $83621 + 100j$; BC/reference initialization adds one/two to the split seed. The expert must exceed random return by $0.1|\bar{R}_{\text{random}}| + 1$ and have collision rate at most .05 and no worse than random. Reference test accuracy must reach .8, and BC/reference closed-loop returns must exceed random. Failed sources cannot be replaced.

HighwayEnv: transfer, PPO and evaluation. Only BC actor layers and the action head are copied into a newly initialized PPO model; the separate value network and optimizer are new. The reference is frozen. PPO uses 16 environments, 125-step rollouts, four epochs, batch 1,000 and the expert's learning rate, discount, GAE, clipping and loss coefficients. Advantages are standardized per minibatch; value loss is unclipped mean squared error. The policy-to-reference KL coefficient decreases linearly from .05 to zero at 250,000 steps. Each gradient update also draws 512 anchor rows uniformly with replacement from the complete collected pool (including fitting-validation/test rows); hard-action cross-entropy has weight .05. These fitting partitions are not the disjoint closed-loop evaluation episodes. Reassignment uses a rollout-wide random single cycle. Online calibration uses a detached mean-KL ratio with denominator floor 10^{-8} and clipping to $[.05, 20]$. Four policy seeds per source are $84600 + 100j + k$, $k = 1, \ldots, 4$. Eleven checkpoints at $0, 50{,}000, \ldots, 500{,}000$ use greedy actions on two blocks of 50 episodes; episode seeds are $b + 10{,}000j + 100k + e$, $b \in \{85100, 86100\}$, $e = 0, \ldots, 49$. Trapezoidal return AUC is divided by the interaction horizon.

MiniHack: planner corpus and supervised models. The three tasks are Room-Random-15x15, MazeWalk-Mapped-9x9 and MazeWalk-Mapped-15x15. The model consumes four 17×17 maps for reachable cells, blocked cells, player and goal, plus task identity; the richer native environment interface is not its model input. A breadth-first planner supplies action labels. For each task and each of 2,048 training levels, collect a clean trajectory and a recovery trajectory that executes a uniformly random cardinal action with probability .15 but retains the replanned action as its label. Validation uses 256 separate clean levels per task. Episodes are capped at 500 steps; any failure invalidates the corpus, rather than being discarded or replaced. Source j uses training, validation

and closed-loop level bases $220000 + 10000j$, $320000 + 10000j$ and $420000 + 10000j$. BC and reference are independently initialized 96-channel, ten-residual-block convolutional models with a 16-dimensional task embedding, 128-unit shared layer, eight-action head and scalar value head. The input convolution and both convolutions in every block use 3×3 kernels, stride one and padding one. Eight-group normalization (Wu & He, 2018) follows each convolution; it normalizes channels within each group independently of the batch. The activation is the sigmoid-weighted linear unit $f(x) = x/(1 + e^{-x})$ (Elfwing et al., 2018). It follows the input convolution, the first normalized convolution in a block, and the sum of the block input and its second normalized output. Features at the player cell, selected by the player-indicator map, are concatenated with the task embedding before the shared layer and activation. Convolutional and linear weights are orthogonal with gain $\sqrt{2}$, except action/value output gains .01/1; biases are zero. They have no recurrence. Both minimize cross-entropy against $\widetilde{y}(a) = .98\,\mathbf{1}[a = a^\star] + .02/8$: label smoothing mixes the planner-action label with a uniform distribution over all eight actions (Szegedy et al., 2016). Only shared and action parameters are supervised. Training shuffles rows without replacement, using batch 512, AdamW at 3×10^{-4}, weight decay 10^{-5} and gradient cap 1. The maximum is 40 epochs, with patience five and selection by validation top-1 accuracy improving by more than 10^{-6}. BC/reference seeds are $520101 + 10000j$ and $520103 + 10000j$. Eligibility requires validation accuracy at least .90 overall/.85 per task and greedy closed-loop success at least .80 overall/.70 per task on 128 new episodes per task. There is no learned teacher-training stage.

MiniHack: transfer, PPO and evaluation. The complete BC model, including its unsupervised value head, initializes PPO; a fresh Adam optimizer uses learning rate 3×10^{-4} and $\epsilon = 10^{-5}$. The separate reference is frozen. PPO uses 32 environments, 64-step rollouts, four epochs, batch 1,024, discount .99, GAE .95, clip .2, entropy weight .01, value weight .5 and gradient cap .5. Advantages use population-standard-deviation normalization within a minibatch. Value loss is one half the maximum of unclipped and .2-clipped squared errors; environment rewards are not rescaled. Uniform replacement sampling supplies 512 training-corpus anchor rows per gradient update, with hard-action loss weight .05. Reference weight is $.05 \max(1 - t/250000, 0)$, using the completed rollout's transition count t. Exact-task random cycles are fixed across PPO epochs. Calibration uses the detached minibatch KL ratio without clipping; mean KL for the reassigned target at or below 10^{-8} aborts the run. Policy seeds are $720100 + 10000j + k$, $k = 1, \ldots, 4$. Evaluation uses greedy actions, 500-step episodes and eleven nominal checkpoints from zero to 500,000 transitions. An intermediate checkpoint occurs at the first vector step reaching its target (at most 31 transitions later); AUC uses the nominal grid. Four disjoint blocks have eight episodes per task, with seeds $b + 10000j + 1000k + 100g + e$, block bases $b \in \{810000, 910000, 1010000, 1110000\}$, task index $g = 0, 1, 2$ and episode index $e = 0, \ldots, 7$. The selected critic-zeroing study below is a separate comparison, not a change to this original assignment protocol.

Pong: original teachers and BC/reference models. The object interface uses RAM-derived object positions over four frames, `ALE/Pong-v5`, frame skip four and sticky-action probability .25. The first expert starts with seed 2001 for 2M requested decisions, then continues the same policy and optimizer for another 8M with environment/RNG seed 2002 after the first budget fails its quality criterion. The second trains from scratch for 10M with seed 2011. Both use separate 128–128 tanh actor/value networks and the Stable-Baselines3 implementation of PPO (Raffin et al., 2021): eight environments, rollout 256, batch 512, four epochs, learning rate 2.5×10^{-4}, discount .99, GAE .95, clip .1, entropy .01, value .5 and gradient cap .5. Final-budget checkpoints must obtain mean return at least $+5$ with at least 15/20 positive greedy evaluation episodes and no 20,000-decision cap hit. Each expert greedily collects 24 training, four validation and four test episodes, with disjoint seeds 3001–3032 (first source) or 3201–3232 (second). All completed episodes are retained; a cap hit aborts collection. Records contain 36 object features, the chosen action and all six expert log-probabilities. Independent 36–128–128–6 tanh BC/reference networks use initialization seeds 4001/4002 or 4101/4102. BC minimizes hard-action cross-entropy; the reference minimizes $\mathrm{KL}(p^E \| p^{\mathrm{ref}})$ on the same rows. Both shuffle rows without replacement, use Adam at 3×10^{-4}, batch 1,024, no weight decay or gradient clipping, and select the final twentieth epoch without early stopping. The test gate requires BC/reference agreement at least .85/.90, mean teacher-to-reference KL at most .10 nats, and greedy reference return at least $+3$ with at least 8/10 positive episodes and no cap hit.

Pong: original assignment PPO and evaluation. BC actor weights initialize PPO's actor; its value network and optimizer are new. These feedforward models have no recurrent memory to transfer. PPO uses the expert hyperparameters above, with policy-to-reference KL weight .05 decreasing linearly to zero at 1M decisions and hard-action anchor weight .05. Anchor minibatches uniformly sample 512 training rows with replacement. Advantages are standardized per minibatch; value loss is unclipped mean squared error and game rewards are not rescaled. Rollout-wide random-cycle targets remain fixed across epochs; the calibration ratio is clipped to $[.05, 20]$ with denominator floor 10^{-8}. Paired policy seeds are 5001–5004 and 5101–5104. Six checkpoints at $0, .25, .5, 1, 1.5, 2$ million requested decisions are evaluated greedily in two five-episode blocks, capped at 20,000 decisions: 6001–6005/6101–6105 for the first source, 6201–6205/6301–6305 for the second. No test checkpoint is selected. The two-source result averages return-AUC contrasts, not the later factorial's three-source normalized effects. For HighwayEnv and Pong, a minibatch of $N > 1$ advantages is standardized as

$$\frac{\hat{A}_i - \bar{A}}{\sqrt{(N-1)^{-1}\sum_{j=1}^{N}(\hat{A}_j - \bar{A})^2 + 10^{-8}}}, \qquad \bar{A} = N^{-1}\sum_{j=1}^{N}\hat{A}_j.$$

Thus these trainers use sample standard deviation; a single-element batch is left unnormalized. MiniHack uses the population-standard-deviation convention stated above. These transformations affect the PPO surrogate, not return targets. All auxiliary fitting optimizers use $(\beta_1, \beta_2) = (.9, .999)$ and $\epsilon = 10^{-8}$; the PPO optimizers use $\epsilon = 10^{-5}$.

The Aligned-minus-calibrated-Reassigned means in these assignment studies are $-.0051$ HighwayEnv return (3/12 positive seed pairs), $+.0025$ MiniHack success (5/12), and $-.325$ Pong return (4/8). These mixed or small contrasts coexist with the positive Pong low-update comparisons below. The additional HighwayEnv expert failed its quality criterion before reference-guided PPO; that assignment comparison was not run.

MiniHack: selected initialization and critic-reset comparison. This outcome-informed follow-up uses the three-epoch BC snapshot, not the original full-BC initializer. Development crosses three initializers (three epochs, ten epochs and the validation-selected full BC) with PPO learning rates 3×10^{-4} and 3×10^{-5} on two reused sources, two paired seeds and four conditions: Aligned, Uniform target, online-calibrated Reassigned and Anchor only (96 policies). A configuration is eligible if Aligned's mean AUC exceeds all three controls and its assignment contrast is positive in at least three of four pairs. Selection first favors configurations exceeding their own retained initializer in both mean AUC and endpoint, then maximizes the smallest mean AUC advantage over the three controls. Ties favor ten-epoch, three-epoch, then full BC, and then the larger learning rate. This selects three-epoch BC and learning rate 3×10^{-4}. All 96 development cells are required before selection.

Three fresh sources then collect planner corpora and fit full BC/reference models with the architecture, collection, fitting and quality criteria above. Separately initialized BC fitting uses the same training corpus, row shuffling, batch size, smoothed loss and optimizer, saving epochs three and ten without validation selection; the selected third-epoch snapshot initializes PPO. For source $j = 1, 2, 3$, training/validation/quality-evaluation level bases are $71000000 + 1000000j$, $81000000 + 1000000j$ and $91000000 + 1000000j$. Full BC, reference and fixed-epoch BC seeds are $8100000 + 1000j + \{1, 3, 5\}$. All three full-model source gates must pass; failed sources cannot be replaced. Confirmation trains the four conditions with three paired seeds per source (36 policies). Policy seeds are $9100000 + 1000j + p$, $p = 1, 2, 3$. PPO settings, anchor sampling and target calibration are those above. Training-level seeds are $1500000000 + 10000000j + 1000000(p-1) + 20000v + e$, where $v = 0, \ldots, 31$ is the environment slot and e its episode count. Each policy trains for 500,000 transitions. Eleven greedy evaluations use four blocks of eight 500-step episodes per task, with seeds $1450000000 + 1000000j + 10000p + 1000b + 100g + e$ for block $b = 0, \ldots, 3$, task $g = 0, 1, 2$ and episode $e = 0, \ldots, 7$. Unlike the original assignment study, AUC integrates the actual checkpoint steps $32\lceil 50000k/32 \rceil$, $k = 0, \ldots, 10$.

The reset candidate was specified after development diagnostics. The twelve probes replay the first rollout's first minibatch, before any optimizer step, for three BC initializers, two development sources and two paired seeds at learning rate 3×10^{-4}. The lower-learning-rate cells have the same pre-update quantities and are not additional probes. For each probe, concatenate gradients over shared parameters, excluding the action and value output heads, and divide the Euclidean norm of

the .5-weighted value-loss gradient by the policy-surrogate gradient norm. The ratio must exceed one in at least eight probes and average above one over the six probes of each source, with a float64 arithmetic check. These are gradients before clipping or Adam scaling; the probe takes no optimizer step. Activation also required all 36 confirmation policies to pass validation and Aligned-minus-retained-initializer AUC to fail its support rule: average three paired seeds per source, then require all three source differences positive and a positive lower 95% Student-t bound (two degrees of freedom). This motivates a critic-initialization test, not a claim that gradient norms identify PPO parameter updates or a causal mechanism. The reset comparison uses the first two paired seeds of these three sources in the same four conditions: 24 newly trained zero-head policies versus their 24 matched confirmation policies. It sets only the initial scalar value-head weights and bias to zero; the action head and shared features are initially identical, and the value head subsequently learns through PPO. The data, frozen reference, optimizer recipe, budget and evaluation seeds remain fixed. The comparison averages paired reset-minus-original AUC differences over four conditions and two seeds within each source, then forms a 95% Student-t interval over the three source means (two degrees of freedom). Zeroing improves this average by $+.0224$ $[+.0124, +.0324]$. Within the zero-head cohort, the assignment contrast is $+.0026$ $[-.0167, +.0219]$ and Aligned minus retaining its three-epoch initializer is $-.0493$ $[-.0796, -.0190]$. These are not 24 independent upstream replications or an improvement established separately in every condition. The primary reset rule requires all three source effects positive and a positive interval lower bound. Subsequent assignment, Uniform/Anchor and retained-initializer tests are gated in that order; all effects remain reported if a gate fails. Values and shared gradients change together.

Pong: sources inherited by the initialization study. The later three-source study does not reuse the preceding two experts. For source $j = 1, 2, 3$, a new expert trains for 10,000,384 decisions with the same PPO recipe and seed $21000000 + 10000j$. Five checkpoints at 2M, 4M, 6M, 8M and the final budget are scored on 20 shared selection episodes, starting at seed $1550000000 + 1000000j$. The highest mean return selects the checkpoint, with earliest-budget tie-breaking and capped episodes ineligible. The selected models must independently pass the same expert-quality criterion on 20 new episodes starting at $1560000000 + 1000000j$; a failure cannot be replaced by the next-best checkpoint. The selected budgets are 8M, 10,000,384 and 6M. Each then supplies 32 complete episodes with seed base $1570000000 + 1000000j$: offsets 1–24 train, 25–28 validate and 29–32 test. Full BC and reference use the twenty-epoch fitting recipe above with seeds $22000000 + 10000j + 1$ and $22000000 + 10000j + 3$; the same offline agreement/KL and ten-episode reference-quality gates apply. Closed-loop fitting checks start at $1580000000 + 1000000j$. These datasets, full-BC comparators and references are reused unchanged in the factorial; its initializers are newly fitted.

The Pong initialization study crosses eligible BC data fraction (20%/100%) with optimizer-update budget on three reused pipelines, three new seeds and two conditions per cell: 72 policies. High-budget fitting continues low-budget weights and optimizer state. Let $d_j(f, b)$ be source j's mean paired Aligned-minus-calibrated-Reassigned AUC difference over its three policy seeds, with eligible data fraction $f \in \{.2, 1\}$ and update budget $b \in \{L, H\}$ (low/high). AUC uses actual checkpoint steps, is divided by the actual training budget and, for these contrasts, by the 42-point return range. The source-level primary contrasts are

$$\Delta_j^{\text{data}} = \tfrac{1}{2}[d_j(1, L) + d_j(1, H) - d_j(.2, L) - d_j(.2, H)],$$
$$\Delta_j^{\text{updates}} = \tfrac{1}{2}[d_j(.2, H) + d_j(1, H) - d_j(.2, L) - d_j(1, L)].$$

Thus positive values mean that more eligible data or more BC updates, respectively, increase the assignment contrast, not absolute return. Each mean and two-sided 97.5% Student-t interval uses the three source contrasts (two degrees of freedom), giving Bonferroni 95% family coverage for these two tests. Directional support additionally requires all three source contrasts to share the interval's sign. These are not success points. More BC updates reduce the calibrated assignment gap: the high-minus-low update contrast is $-.2645$ $[-.4848, -.0442]$; the data-fraction contrast is $+.0436$ $[-.0599, +.1471]$. Low-update treatment gaps are $+.2455$ and $+.2852$, with positive secondary contrasts, yet all 36 Aligned AUC observations are below retaining the available original full-BC policy. That original policy is not the factorial's high-budget initializer. Four runs failing initial-critic equality were corrected and rerun before contrast analysis; excluded originals and failure records are preserved. Table 44 gives all four treatment gaps. Each initializer uses a 36-input, two-layer 128-unit tanh actor, six actions, uniform replacement batches of 512, Adam at 3×10^{-4} and gradient clipping at one. Within each source, both data fractions start from identical weights (seed $61000000 + 10000j + 1$)

and draw batches with a private generator (seed $62000000 + 10000j$). The 20% subset is the recorded earlier short-BC subset; it is not reselected. For source j, its construction initializes NumPy's default random generator with $22000000+10000j+5$, permutes all training-row indices once for a diagnostic sample (first 8,192 indices), then permutes all indices again and retains the first $\max(1, \lfloor .2N \rfloor)$, where N is the training-row count. The second permutation's retained indices define the reused subset. Low/high update counts are 168/2,780, 201/3,340 and 195/3,220 for the three pipelines; the final low/high updates select the checkpoints, without validation-based selection. PPO uses eight environments, 256 steps, four epochs, batch 512, learning rate 2.5×10^{-4}, clipping .1 and anchor .05; reference weight .05 falls linearly to zero at 1,000,448 decisions. The eleven checkpoint steps are $2048\lceil(2000000k/10)/2048\rceil$, $k = 0, \ldots, 10$, ending at 2,000,896 decisions. Each uses four blocks of ten greedy episodes, capped at 40,000 decisions. Policy seeds are $63000000 + 10000j + 100k$, $k = 1, 2, 3$, and evaluation seeds are $1800000000 + 1000000j + 10000k + 1000b + e$, $b = 0, \ldots, 3$, $e = 0, \ldots, 9$. The actor-only transfer, fresh critic/optimizer, full-data anchor, frozen reference, normalization and calibration rules remain as above. This is moderation of a calibrated recipe contrast, not identification of competence or language as its mediator.

Table 44: Pong initialization study: all four Aligned-minus-online-calibrated-Reassigned gaps in return AUC divided by 42. Three pipeline means; secondary 95% t intervals. Full-data BC used as a retained comparator is separate from these initializers.

Eligible data	BC updates	Difference	Interval
20%	Low	+.2455	$[+.0177, +.4732]$
100%	Low	+.2852	$[+.1107, +.4597]$
20%	High	$-.0229$	$[-.1144, +.0687]$
100%	High	+.0246	$[-.0663, +.1155]$

O.2 Outcome-informed development and failed expansion rules

The studies below select or assess recipes rather than provide independent confirmation. Their complete cell records and implementation accompany the supplement; outcomes used to choose follow-ups are identified below. No failed primary rule was replaced by a favorable endpoint.

Protocol map for the development probes. The autonomous CrafText interface factorial uses the base and extension prompts in Appendix C.5, environment seed 9147101, inference seed 9142026, two episodes per instruction and a 1,000-step horizon. Its random control samples uniformly from the seven non-crafting/placement actions and the currently prerequisite-feasible crafting/placement actions. After all five cells finish, a Qwen variant is eligible for collection if macro success is at least .10, at least six goals have a success, and its advantage over this random control is at least .05. Highest macro success selects the interface; ties within one successful episode prefer lower GPU time per decision. Missing cells prevent selection. The selected autonomous corpus retains all 114 episodes. Split seed 9152200 assigns one episode per goal to training and one to validation, with no test partition. Paired references fit actual Qwen actions or source-4 expert argmax labels on these same observations. They use the hard-label recurrent objective and optimizer of Appendix H, reference seed 9152301, 64-window batches and 480 epochs (1,440 updates), retaining minimum validation cross-entropy without early stopping. PPO retains source-4 BC and the demonstration anchor, copies the complete BC model with a fresh optimizer, and uses policy seed 9153003 and the common 4,194,304-frame recipe. Nine checkpoints use two blocks of two episodes per goal, horizon 1,000, evaluation seed 9153001 and block stride 100003. No new expert or BC fit is made.

The four BabyAI label-probe settings below are, in order, grid/direct, coordinates/direct, grid/reasoning and coordinates/reasoning. They share 1,024 source-4 reference-fitting rows, sampled uniformly without replacement using seed 9149100, and inference seed 9149101. The first cell's labels are reused. Both representations supply the same 7×7 visible object/color/door-state grid, carried object, instruction and public action rules, without history or planner advice. Grid rows run from six cells ahead to the agent's row, with lateral offsets $-3, \ldots, 3$. The coordinate variant lists all unknown and nonempty cells by lateral offset and forward distance; unlisted cells are explicitly known empty. Direct decoding emits one of six action codes A–F; reasoning adds a separate request of at most 256 tokens before that action. All decoding is greedy, thinking mode is disabled, and the

4,096-token context limit rejects overflow instead of truncating inputs. Invalid actions fail. No model is fitted in this probe. Expansion requires agreement at least .50 and at least .05 above the sampled majority-action baseline; all four cells must finish before selection. This is separate from the clean online-PPO study.

The CrafText frozen-model context probe samples 256 training windows without replacement from each source 4–8 using seed $9186900 + s$ for source s. It reuses the BC, expert-reference and Qwen-reference models of the direct-label study, without optimization. Each window starts at zero memory and honors recorded resets. The cyclic condition replaces canonical instruction index g by $(g + 1) \bmod 57$; other conditions zero all instruction tokens or reset memory every step. Top-action changes are measured against each model's unaltered predictions on valid positions, not against counterfactual labels. The separate fixed-state Qwen interface probe samples 512 valid labelled training positions per source without replacement, seed $9188000 + s$. It compares the base current/direct prompt with the prerequisite/reasoning extension, using 256 reasoning tokens (not the autonomous factorial's 64), inference seed 9188101 and batches of 128. The action constraint, context and failure rules are otherwise those of Appendix C.5. Neither frozen-model prediction changes nor fixed-state label agreement is closed-loop policy performance.

Teacher interface and autonomous collection. The basic interface supplies the current observation and public rules; the prerequisite interface adds observable action requirements without masking actions. Direct decoding emits one action token; deliberate decoding first generates up to 64 reasoning tokens, then requests one action token. Across 57 goals and 114 episodes per cell, basic/direct, prerequisite/direct, basic/deliberate and prerequisite/deliberate interfaces succeed in 7, 7, 12 and 23 episodes, respectively; a random prerequisite baseline succeeds in 6. The selected final interface contributes all 18,459 decisions, including failures. On one source and one seed, Qwen Aligned, Qwen Reassigned, paired expert Aligned and Anchor only have AUC .28125, .28975, .28262 and .27714, respectively (retained BC .34649). Qwen's two required contrasts are $-.00850$ and $+.00411$. The rule for proceeding to a larger study required both AUC differences to be positive, so no larger study followed. The favorable assignment endpoint is not promoted.

BabyAI input and CrafText context probes. Four clean-input BabyAI Qwen settings achieve planner-label agreement of 32.91%, 41.50%, 41.02% and 49.22% on 1,024 fixed rows, below the 51.27% majority-action baseline; no large annotation/PPO expansion follows. Privileged planner and policy-visible Qwen information differ. On CrafText training rows, cyclic instruction replacement changes the top action on 50.38% of Qwen-reference, 13.75% of expert-reference and 20.28% of BC predictions. This probe measures instruction sensitivity; without counterfactual action labels, it does not determine whether the changed recommendations are correct or useful. A five-source fixed-state interface probe decreases expert agreement by 4.41 points and prerequisite failures by 6.09 points; its rule required an expert-agreement gain of at least five points on average and a positive gain in at least four sources, with nonincreasing prerequisite failures on average and in at least four sources. The agreement requirement fails. Both favorable and unfavorable proxies remain recorded; neither identifies a transfer mechanism.

BC-budget development. Table 45 retains every pilot arm. Positive low-budget contrasts motivated the separate 60-policy confirmation in Appendix K, which failed its own joint rule. Near-floor initialization describes the observed regime; it does not establish why confirmation failed.

Table 45: BC-budget development: one source, one seed, all six arms. Success fractions, not population confidence intervals. Retained starting BC is .03070 after one epoch and .26754 after ten; separately available original full BC is .26316 on these episodes.

BC epochs	Condition	AUC	Endpoint
1	Aligned	.04304	.04386
1	Reassigned	.03427	.04386
1	Anchor only	.02906	.03070
10	Aligned	.30757	.35526
10	Reassigned	.30839	.35965
10	Anchor only	.33745	.35088

Probability and mixture development. The probability pilot uses one source and one policy seed. Three references fit the same 49,152 expert-visited positions (steps 0, 16, 32 and 48 of each 64-step sequence), with the same 1,440 fitting updates. The soft target is Qwen's 17-action probability vector; the hard target selects its maximizing action. The entropy-smoothed target keeps that maximizing action but gives each other action equal probability. Specifically, its top probability is $1 - \varepsilon$ and each other probability is $\varepsilon/16$, choosing $0 \leq \varepsilon \leq 16/17$ to match the Qwen target's categorical entropy. This matches fitting-target entropy, not subsequent reference or policy entropy. All six PPO arms retain expert BC and anchoring, train for 4,194,304 frames, and use two evaluation blocks with all 57 goals. The six probability-study AUCs are .28755 (soft Aligned), .30811 (soft Reassigned), .29715 (hard Aligned), .31634 (entropy-smoothed), .30154 (Uniform) and .31552 (Anchor only), with retained BC .29386. The rule for a larger study required soft Aligned to exceed Reassigned and Anchor only by .01 AUC, hard and entropy-smoothed Aligned by .005, and Uniform and retained BC by a positive amount. All six requirements fail. An initial probability-extraction check changes four of 128 actions; a follow-up disabling prefix caching and engine multiprocessing has complete repeated agreement. This bundled correction does not identify a unique cause. All trained runs finish before a metadata-serialization failure. An evaluation-only recovery matches every saved aggregate block and adds episode records; no training outcome is replaced.

The one-source mixture pilot reuses those queries and fits expert-only, equal expert–Qwen and equal expert–Uniform probability targets at the same positions and fitting budget. It trains four new PPO policies (including mixture Reassigned), retaining the preceding pilot's BC, anchor, policy seed, training budget and evaluation blocks. Qwen, Uniform, Anchor-only and retained-BC outcomes are reused. Mixture Aligned obtains .33279 AUC, versus .30482 for mixture Reassigned, .30757 for expert Aligned and .31935 for expert–Uniform. The rule required advantages of .01 over expert Aligned, mixture Reassigned and Anchor only, .005 over expert–Uniform, and positive advantages over Qwen Aligned, Uniform and retained BC. It passes these seven development criteria; the 105-policy confirmation in Appendix I does not. Probability-study controls are reused, not additional observations.