WHEN QUEUE POLICY BECOMES LEARNING POLICY: AUDITING BATCH PATHS IN AGENTIC RL

Anonymous authors
Paper under double-blind review

ABSTRACT

An asynchronous rollout queue can change learning even when every trajectory is eventually consumed: it determines which complete prompt groups share an optimizer step and in what order. We audit this batch-path effect using immutable recorded experience and persistent-Adam GRPO. In a prospective six-seed Qwen3-0.6B experiment, reversing two task-family blocks changes final product-minus-shared success by -23.6 percentage points [-30.5, -16.3], with experience, tokens, initialization, and update count matched. Each order favors its first family relative to balanced batches. An independent six-seed intervention clears only Adam's first moment at the family switch, attenuating the contrast by +21.88 points [+16.54, +27.21]; all 168 checkpoint-and-optimizer control pairs are exact. The reset mainly removes the first-family advantage and reduces aggregate accuracy.

We make the batch-composition choice explicit through a per-update contract for required coverage and histogram total variation, with earliest-feasible complete-group admission. Simple per-family queues exactly match the general allocator in the tested two-family case. Measured multi-turn traces expose composition violations and an explicit waiting cost. A separate 3B learning result remains unconfirmed, and a public-code pilot fails validity and its numerical screens. The evidence supports a controlled queue-path diagnosis and a composition guarantee, without establishing broad learning generality or time-to-quality improvement.

1 INTRODUCTION

Agent post-training is a distributed learning pipeline. Policies generate multi-step rollouts, environments execute tools, verifiers assign rewards, and learners publish new checkpoints. Generation length, tool latency, retries, and failures make complete prompt groups eligible at very different times. Modern asynchronous systems improve utilization by consuming what is ready (Noukhovitch et al., 2024; Fu et al., 2025; Lu et al., 2025; Hu et al., 2026). Recent schedulers also expose workload mixtures, complete-group readiness, task diversity, and fast-sample skew (Hong et al., 2026; Chen et al., 2026b; Ren et al., 2026; Xu et al., 2026). We study the learning consequence of that choice after final experience and exposure are fixed: which complete prompt groups share an optimizer step, and in what sequence?

Asynchrony alone is not the mechanism. If one fixed same-policy batch is consumed exactly once by a permutation-invariant update, completion order is only a systems event. A queue becomes part of the learning algorithm in two distinct ways. *Selection/exposure distortion* changes the eventual multiset of units or an explicit weight assigned before batching. *Schedule/partition distortion* preserves the units and their recorded fields but changes the sequence of minibatches. Batch-induced normalization, clipping, and optimizer weighting are then mediators of the schedule, not a separate selection category. Behavior-policy staleness is complementary: it cannot restore excluded experience, and family-conditioned learner age can itself be induced by the batch path even when marginal lag matches.

This distinction matters for agentic RL because the atomic object is often a complete prompt group, and the update is nonlinear in the collection. For GRPO, reward normalization couples completions of one prompt, whose membership we preserve; persistent Adam couples consecutive steps (Shao et al., 2024). "Consume everything eventually" therefore does not specify the learning computation.

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.

A queue can quietly become an unrequested curriculum—beneficial or harmful—without changing final sample count. Prior work intentionally changes order, composition, or exposure (Zhang et al., 2026a; Zhu et al., 2026b; Gao et al., 2026); we instead intervene on the runtime-induced path of already-recorded units.

We organize the evidence around three questions:

- **Does the batch path change learning at equal experience?** A prospective six-seed reversal changes the task-family tradeoff; exact fixed-batch controls rule out timing alone.
- **What explains the first-family advantage?** An independent six-seed intervention identifies a material first-moment dependence, with a capability cost that limits its use as a remedy.
- **What can an admission contract guarantee?** Complete-group auditing and per-update composition constraints enforce a declared target at a measured waiting cost. Practical baselines and unsuccessful extensions delimit the guarantee.

The batch-path identity and allocation dynamic program support these experiments; neither is claimed as a standalone theoretical contribution.

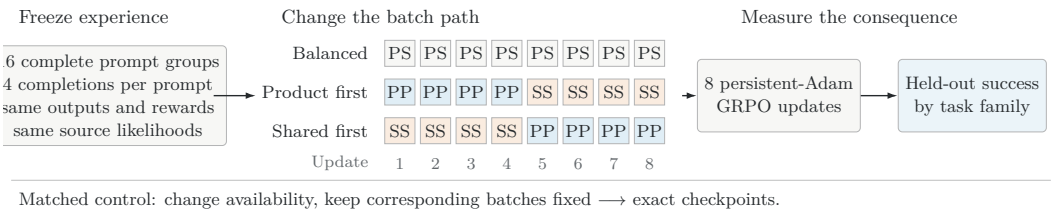


Figure 1: Controlled batch paths. Each square is one optimizer step on two complete prompt groups; P/S denote the product/shared families. All paths consume the same eight groups per family. Canonical fixed-batch controls keep corresponding batches identical while changing availability. The independent first-moment experiment resets only that state immediately before step 5.

2 Learning Semantics of a Queue

2.1 Batch Paths, Not Eventual Bags

Let B_t be the complete learner batch at update t and let $x_t = (\theta_t, o_t)$ include parameters and all optimizer state. Conditional on matched per-update randomness, write a stateful update as $x_{t+1} = x_t + u_t(x_t; B_t)$. Consider intended and realized batch paths P, Q with common x_0. Define

$$D_t = u_t(x_t^P; B_t^Q) - u_t(x_t^P; B_t^P),$$
$$R_t = u_t(x_t^Q; B_t^Q) - u_t(x_t^P; B_t^Q). \tag{1}$$

Adding and subtracting the matched-state update gives the bookkeeping identity for $\delta_t = x_t^Q - x_t^P$:

$$\delta_{t+1} = \delta_t + D_t + R_t, \qquad \delta_T = \sum_{t=0}^{T-1} (D_t + R_t). \tag{2}$$

The proof and a Lipschitz bound are in Appendix A.1. D_t is matched-state batch replacement; R_t is feedback through changed parameters and optimizer state. Even if $\biguplus_t B_t^P = \biguplus_t B_t^Q$, neither term must cancel. Conversely, identical corresponding canonicalized batches imply identical state at every checkpoint. We do not estimate D_t and R_t separately or claim they are novel causal mediators. The complete prompt groups are preserved: the treatment changes which groups share an optimizer step and when they are used, not which completions define each prompt's reward baseline.

2.2 THE IMPLEMENTED DECLARED-COMPOSITION CONTRACT

Write $B_t = (H_{t,1}, \ldots, H_{t,K})$, where H is a complete prompt group, and let declared label map $s : H \mapsto g \in \mathcal{G}$. The learning specification supplies target $p_t(g)$ and required support $\{g : p_t(g) > 0\}$. With

$$\widehat{q}_t(g) = K^{-1} \sum_{j=1}^{K} \mathbf{1}[s(H_{t,j}) = g], \tag{3}$$

the implementation releases a batch only if

$$\mathrm{supp}(p_t) \subseteq \mathrm{supp}(\widehat{q}_t), \qquad \mathrm{TV}(\widehat{q}_t, p_t) \le \epsilon. \tag{4}$$

For any declared-label diagnostic $f_h(B) = K^{-1} \sum_j h(s(H_j))$ whose range is at most one, the variational property of discrete TV gives $|f_h(B_t) - \sum_g p_t(g)h(g)| \le \epsilon$.

The contract controls declared label counts, not within-stratum fidelity. It does not certify that the target mixture is optimal or fair. Strata and p_t are versioned parts of the learning specification. Appendix A.3 formalizes this projection and its uncontrolled within-label term.

3 SYSTEM INTERFACE

Complete-group lifecycle. A task is submitted, queued, started, generated, evaluated, and made available; it is then admitted, rejected, cancelled, failed/retried, or consumed. Update start/finish and checkpoint publication close the lineage. Each append-only event records unique experiment, task, trajectory, worker, declared-stratum, source-policy, and monotonic-time fields. Content is durable before its availability receipt. Queue state never mutates the recorded policy output, tool result, reward, verifier result, or behavior log probability.

GRPO couples the completions of a prompt through reward normalization. A prompt group is therefore eligible only when all configured rollout indices are present exactly once, share one source-policy version, and have durable content and evaluation fields. Before learning, the runtime checks exact batch cardinality, unique IDs, lag bounds, cancellation/rejection exclusion, and canonical ordering. Every batch event attaches its selected IDs and strata to the optimizer update; checkpoint publication uses a temporary directory, file fingerprints, atomic rename, and a durable latest-version pointer.

Earliest feasible admission. At elapsed time τ, let $N_g(\tau)$ be eligible counts, K batch size, and $\ell_g = \mathbf{1}[p_g > 0]$. Feasible integer allocations satisfy $\ell_g \le n_g \le N_g(\tau)$ and $\sum_g n_g = K$. The controller computes

$$d^*(\tau) = \min_n \frac{1}{2} \sum_g |n_g/K - p_g|, \qquad \tau^* = \inf\{\tau \le \tau_{\max} : d^*(\tau) \le \epsilon\}. \tag{5}$$

An exact dynamic program retains, for each prefix of strata and selected count, the least cumulative absolute target error and a deterministic tie-breaking tuple. It uses $O(GK^2)$ transitions for G strata. Evaluating distinct eligibility events in time order returns the earliest feasible event; the live coordinator has the same guarantee at poll granularity. Appendix A.2 proves the invariant. Within each allocated stratum, seeded stable hashing chooses among currently eligible groups. This removes arrival order only within that eligible set; it cannot remove timing-dependent within-stratum selection.

If no allocation satisfies coverage and TV at the deadline, the interface reports missing support and best attainable TV instead of renormalizing the target. Tighter budgets can increase wait or cancellation. The point is to expose that tradeoff and fail visibly, not to promise a free quality or throughput improvement.

4 EXPERIMENTAL DESIGN

Controlled learning paths. The recorded learner uses revision-pinned Qwen3-0.6B with rank-8 LoRA (Hu et al., 2022) and two verifiable arithmetic tool families. Product-chain tasks require

expressions of the form $abc - d$; shared-rate tasks require $ad + bd - c$. The model translates a word problem into a calculator call, and executed exact success is the endpoint. Every trajectory, reward, token log probability, and behavior-policy field is immutable before controlled availability is assigned. GRPO uses persistent Adam, a fixed reference policy, completion masking, and no reward-based prompt filtering. Held-out greedy tool execution measures exact success.

An earlier balanced-versus-product-first study motivated a prospective three-schedule control (Figure 1). Six unused seeds compare balanced, product→shared, and shared→product paths. Each seed fixes 16 prompt groups, four completions per group, and eight continuous GRPO updates. Two complete groups enter every update; all paths consume every group once. A canonicalized fixed-batch regime supplies the negative control under each availability condition.

The primary endpoint is the update-8 reverse-minus-forward change in product-minus-shared success, evaluated on four common 64-task panels (256 held-out tasks total). Either sign could support the registered order-effect hypothesis. We average the six seed rows and four panel columns; 10,000 crossed empirical bootstrap draws resample paired seed rows and common panel columns. These intervals describe stability of the frozen matrix, without treating its 24 cells as independent replicates or establishing population generality. Intermediate checkpoints have unequal cumulative family exposure and serve only as path diagnostics. Appendix B.2 gives the task definitions, settings, protocol clarification, and full results.

All trajectories come from the same behavior/reference policy. Marginal source lag is identical (two groups at each step 0–7), but blocking assigns mean family-conditioned lags 1.5/5.5 or 5.5/1.5 rather than 3.5/3.5. The estimand is therefore the total minibatch partition-and-order path, including Adam history, clipping, and family-conditioned learner age. The order-reversal study alone does not isolate an optimizer submechanism; Section 5.2 adds a separate intervention.

Controller and natural execution. The controller benchmark uses 8,000 seeded 64-candidate pools crossing two task families and two content subtypes. Each selects 16 under two timing conditions and 11 policies: 176,000 paired replay decisions. Baselines are arrival-first-16, task-marginal FIFO, stable-hash reservoirs, oracle propensity weighting, and wait-all/select-K.

For bounded natural relevance, three persistent Qwen3-8B actors execute the same frozen 32-task τ^2-bench panel in three repeated runs, with a Qwen2.5-3B user simulator on a fourth GPU (Yang et al., 2025; 2024; Barres et al., 2025). The runs yield three measured service-time traces over the same task IDs, not independent workload samples. Before analysis, 4,096 common queue orders and 1,024 duration permutations per trace were frozen. Coverage loss and mean declared-histogram TV are joint primary endpoints. No learner update is performed on this 8B path.

Independent workload checks separate reward support, timing association, learner validity, and held-out learning. These gates answer different questions: a workload can support GRPO while failing to exhibit the timing contrast needed for a natural-queue claim. The 3B, SQL, code, and public-mixture outcomes are summarized in Section 5.4 and detailed in Appendix D.

5 Results

5.1 Reversing order changes the task-family tradeoff

Figure 2(a,b) shows a consistent first-family advantage relative to balanced batches. Product-first increases product success by 2.9 points and reduces shared success by -5.1 points; reversing the blocks changes these effects to -10.5 and 5.1 points. The registered reverse-minus-forward family interaction is -23.6 [-30.5, -16.3] points, negative in all 6 training seeds. It reverses the deviation from balanced; the absolute product-minus-shared ranking remains positive in all three paths. Appendix Table 3 reports the full family and aggregate intervals. The four blocked-minus-balanced family intervals are pointwise, unadjusted secondary summaries; the registered interaction is primary.

Every continuous path consumes the same 16 prompt groups and 64 trajectories within seed. The two blocked paths reverse only the family order; balanced updates mix one group per family. All 96 corresponding fixed-batch condition-pair checkpoints are byte-identical, while continuous paths diverge. This excludes content generation, reward, final membership, total token exposure, step count, initialization, and a fixed-batch timing permutation. It does not exclude task-conditioned

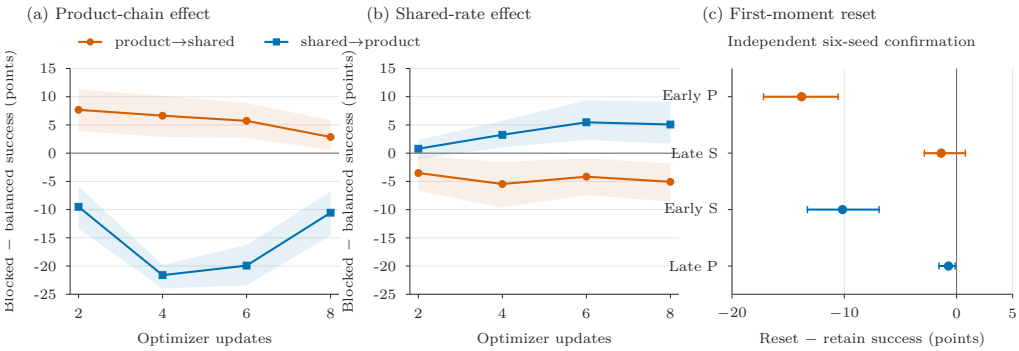


Figure 2: Queue-order control and independent mechanism confirmation. (a,b) Blocked-minus-balanced family effects in the original six-seed cohort; earlier checkpoints have unequal cumulative family exposure. (c) First-moment reset-minus-retain success in six new seeds: the early family loses its advantage without a late-family benefit. P/S denote product/shared; colors retain the schedule legend. Both cohorts consume the full matched corpus at update 8. All intervals are pointwise, unadjusted crossed seed–panel intervals; the two registered interactions are reported in the text.

learner age or Adam history: those are precisely parts of the queue-induced batch path. No schedule produces a supported aggregate quality gain: product→shared changes aggregate success by -1.1 [-3.4, 1.1] points, whereas shared→product lowers it by -2.7 [-4.7, -0.5].

5.2 First-moment memory contributes to the first-family advantage

Saved-Adam diagnostics motivate an intervention on optimizer memory. After a two-seed development screen, we freeze a six-unused-seed confirmation using new corpora and four new 128-task panels (512 tasks total). Each of the three schedules is trained with retained Adam or with only its first moment zeroed immediately before update five. Parameters, second moments, and the optimizer clock are preserved at the intervention; the balanced path receives the same reset as a matched control. The matrix has 54 branches and 432 updates, including unmodified fixed-batch controls. All 168 declared tensor and optimizer-state control pairs are exact.

Let $D_a = (P - S)_{\text{reverse,a}} - (P - S)_{\text{forward,a}}$ for arm a. The registered attenuation $I = D_{\text{reset}} - D_{\text{retain}}$ is +21.88 [+16.54, +27.21] points, positive in 6/6 seeds. It passes the frozen five-point minimum, positive crossed-lower-bound, seed-direction, and parent-replication rules. Retained D is -35.09 points and reset D remains -13.22: first-moment memory is a material contributor, not the entire explanation.

Figure 2(c) shows the capability cost. Reset-minus-retain success for the first family is -13.80 points in product-first order and -10.16 points in shared-first order. The second family changes by -1.37 and -0.72 points, respectively; neither estimate indicates a second-family benefit. Aggregate success falls in all three schedules, including the balanced reset control. Thus attenuation comes mainly from losing the early family's benefit, not recovering the late family. The result establishes a first-moment dependence, not an aggregate improvement or a novel theory of optimizer memory (Sweeney, 2026). Appendix B.3 gives every seed, pointwise secondary interval, the source-density convention, and the separate development deviation.

5.3 Composition guarantees require waiting

Every controller batch satisfies its declared coverage and TV bound. In the paired replay, arrival-first-16 releases in 1.129 s with joint-histogram TV 0.326. Task-marginal FIFO restores 50/50 task balance but retains joint TV 0.344, illustrating the within-label limitation in Appendix A.3. Zero-TV admission releases at 3.18 s, 8.9% of the 35.712 s wait-all/select-K time. At $\epsilon = .25$, it releases at 1.621 s with TV 0.217 and 73.6% of the arrival-first candidate release rate. Candidate count per release time is a replay metric, not GPU throughput.

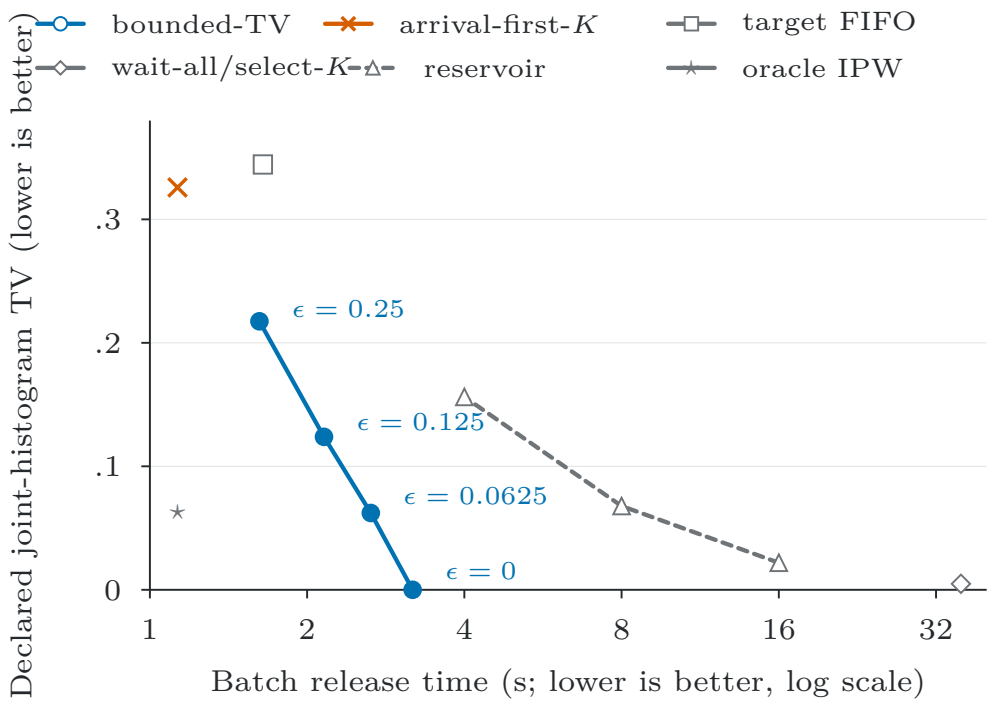


Figure 3: Declared-composition fidelity versus median batch release time in trace replay (lower left is better). Circles vary the TV budget. Oracle IPW retains raw wait but reports weighted TV and has ESS fraction 0.574.

Oracle inverse-propensity weighting cannot restore a missing required stratum and reduces ESS to 57.4%. At the two experimental coordinates $(G, K) = (4, 16)$ and $(2, 4)$, single-core controller medians are 150.1 and 46.9 μs. Durable logging costs more than the allocator at these snapshots; Appendix C.2 reports tail latency, recovery, storage, and event-rate measurements. The controller therefore makes a measured tradeoff rather than claiming negligible overhead.

Simple queues and bounded random buffering. On the six measured 3B traces and their six equal-duration controls, we also replay FIFO, capacity-4/8 random buffers (256 fixed random seeds each), and one queue per family, always consuming all 16 groups in eight updates. With two families and batch size two, the only zero-TV allocation is one group per family. Under matched tie-breaking, simple queues and the exact controller therefore produce identical selected IDs and readiness times in all 12 traces. Table 1 shows that buffering partly reduces imbalance but gives no support guarantee. In equal-duration controls, FIFO already has zero TV, whereas random buffering creates mean TV .202/.230. A general allocator adds no benefit over simple queues at this experimental coordinate.

Table 1: Consume-all replay of six constructed-service 3B traces. Missing is the fraction of batches lacking a family; age gap is the absolute difference in families' mean optimizer-step indices. Every policy's last batch is ready at the same mean 3.603 s. Readiness excludes learner compute and is not training wall time or time-to-quality.

Policy	Mean TV	Missing (%)	Mean ready (s)	Family age gap
FIFO	0.500	100.0	1.224	4.00
Random buffer 4	0.394	78.7	1.674	3.50
Random buffer 8	0.288	57.6	2.575	1.92
One-per-family queues	0.000	0.0	2.241	0.00
Exact TV controller	0.000	0.0	2.241	0.00
Wait all / balanced	0.000	0.0	3.603	0.00

5.4 INDEPENDENT WORKLOADS DELIMIT THE LEARNING CLAIM

Natural episode service spans 19.5–22.3× across repeated runs. Every direct arrival-first-8 batch has nonzero declared-histogram TV and one of three omits the airline/high stratum. The $\epsilon = .125$ controller restores declared coverage in all three orders; zero-TV also preserves coverage.

Appendix Table 7 reports all policies and traces. Exact uniform size-eight sampling already misses a stratum 27.24% of the time. Conditional on each measured trace and the frozen order ensemble, identity-bound durations yield 33.4–35.0% coverage loss, 6.5–8.1 points above duration-permutation means ($p = 0.0010$ per trace). Mean-TV excess is only 0.0205–0.0211, below the preregistered 0.025 rule, and median TV is unchanged. No trace passes the full joint decision rule. This is bounded evidence of a duration-associated coverage component, not population prevalence or an 8B learning effect.

The learning extensions provide no confirmed cross-workload replication:

- **Measured-service 3B: not confirmed.** Six new seeds cross calculator/hash service conditions with FIFO, a 1:1 composition contract, and fixed-batch controls. The update-8 fast–slow interaction is $+2.08$ $[-1.11, +5.34]$ points, below the three-point rule with an interval crossing zero. The study carries a disclosed checkpoint-metadata deviation; its literal byte gate remains failed (Appendix D.1).
- **Public code: invalid pilot.** Official MBPP tasks and Qwen2.5-Coder-1.5B-Instruct provide a four-seed, 32-update boundary test (Austin et al., 2021; Qwen Team, 2024). Strict finalization fails despite exact model/Adam controls. The descriptive interaction is +0.781 points with 2/4 matching signs, below both numerical screens. No retrospective serialization exception is accepted; confirmation is not run (Appendix D.3).
- **SQL and predictor checks: stopped before learning.** Public Spider calibration lacks sufficient mixed-reward prompt groups and fails its timing gate (Yu et al., 2018). An independent 3B public math/code mixture passes reward support but fails natural timing and the frozen gradient-premise screen (Appendix D.4). Neither supplies a test of held-out order sensitivity.

Invalid pilots, failed support screens, and inconclusive learning effects are distinct outcomes; none establishes the absence of order effects in general.

6 RELATED WORK

Asynchronous RL systems. IMPALA, asynchronous RLHF, AReaL, and Stable Asynchrony address off-policy mismatch, staleness, or weighting variance (Espeholt et al., 2018; Noukhovitch et al., 2024; Fu et al., 2025; Huang et al., 2026). Worker rates can alter objective weight (Mahran et al., 2026). RollPacker, DORA, Heddle, Libra, and ROLL Flash reduce bubbles, bound lag, or schedule long tails (Gao et al., 2025; Hu et al., 2026; Zhang et al., 2026b; Chen et al., 2026a; Lu et al., 2025). RolloutPipe pipelines complete groups with trainer FIFO and frontier dispatch; TideRL prioritizes group completion and task diversity; WAR explicitly notes fast-sample skew (Chen et al., 2026b; Ren et al., 2026; Xu et al., 2026). These establish that readiness-aware scheduling and complete-group admission are not new by themselves. Staleness–learning-rate analyses instead characterize behavior-policy mismatch and cumulative learner drift in asynchronous GRPO (Song et al., 2026). Our paired trajectories share one behavior policy and the same marginal lag multiset; family-conditioned learner age remains part of the batch-path treatment rather than a separately identified mechanism.

Mixture contracts and semantic boundaries. MISA-T lets the trainer declare a target workload mixture, derives a long-run backlog condition, and measures target–completion TV (Hong et al., 2026). Our narrower contract requires support and histogram TV independently for each optimizer update, returns the exact earliest feasible complete-group batch, and fails closed when that update target is infeasible. Periodic Asynchrony and RoutePack provide the complementary null boundary: reordering all units inside a sealed, permutation-invariant update preserves semantics (Lu, 2025; Shen et al., 2026). Our treatment repartitions fixed units across persistent optimizer steps; its canonicalized fixed-batch control lies inside that harmless equivalence class.

Order, curriculum, and sampling. Classical curriculum and shuffling work establishes that intentional example order can alter optimization (Bengio et al., 2009; Safran & Shamir, 2020; Nguyen et al., 2021). Sweeney (2026) directly analyzes equal-multiset order effects under optimizer memory using deterministic replay. Liu & Li (2026) studies signed finite-horizon AdamW responses through joint parameter and moment dynamics. Optimizer resets also have precedent in deep RL and LLM training (Asadi et al., 2023; Huang et al., 2025). Stateful order sensitivity, replay controls, and momentum resets therefore are not general novelty claims here. Recent LLM-RL work intentionally organizes prompts, rollout allocation, or problems per batch. IsoCompute varies rollouts and problems per batch and reports interference and stability effects (Cheng et al., 2026); SortedRL and ReOrder-OPD construct deliberate curricula, while SDO, Shuffle-R1, DARTS, and graph-difficulty scheduling change composition, exposure, allocation, or weights (Zhang et al., 2026a; Zhu et al., 2026b; Gao et al., 2026; Zhu et al., 2026a; Wang et al., 2026; Liu et al., 2026). ReOrder-OPD is closest in manipulating prompt order, but constructs a reliability-aware curriculum for on-policy distillation. We audit an unrequested runtime path of immutable recorded units, prospectively reverse its family order, and enforce declared composition per optimizer update. ASTOR explicitly adapts code-task training budgets and per-task KL penalties using utility estimates (Chen et al., 2026c). Our intervention instead fixes realized outputs, rewards, and exposure before varying the batch path; it does not propose adaptive code curricula. Single-rollout methods show that complete groups are algorithm-specific (Hou et al., 2026); classical importance and stratified sampling instead target weighting and variance (Zhao & Zhang, 2015; Carpentier & Munos, 2011).

7 Discussion and Limitations

The scientific boundary is sharper than "asynchrony is biased." A completion permutation before one fixed update is harmless under a permutation-invariant functional. Selection changes eventual exposure. Schedule/partition changes the sequence of nonlinear stateful updates. The reverse-order experiment tests that total effect and identifies an order-sensitive component of the combined partition-and-order path. It cannot separate task-conditioned learner age, clipping, homogeneous-batch autocorrelation, and Adam feedback. The independent first-moment intervention identifies a material dependence on that state at the family boundary, while its residual contrast leaves other contributors open.

The implemented contract is intentionally narrower than the theory. It bounds declared histograms, not full batch laws, conditional quality, reward-component visibility, or fairness. Coarse 50/50 task strata can hide content selection. Deployments should version p_t, audit submitted/eligible/consumed IDs, monitor within-stratum diagnostics, and treat infeasible support as an alarm. Weighting cannot recreate absent experience and may lower effective sample size.

Learning generality is the largest limitation: one 0.6B model, two arithmetic templates, eight updates, and one optimizer. Neither blocked order improves aggregate success. The 8B traces contain only seven successes in 96 episodes and no learner. The measured-service 3B effect remains unconfirmed with a metadata deviation; MBPP is invalid, and the independent gradient premise fails. Thus no cross-workload learning benefit or accuracy predictor is established. Earlier inconclusive integrations and stopped SQL studies remain separate (Appendix D), with no changes to frozen outcomes.

Systems evidence is single-machine, with tens of strata and one ext4 filesystem. Multi-GPU scaling uses a logical membership barrier; multi-node admission is untested. Release time excludes learner compute and establishes neither throughput nor time-to-quality improvement.

8 Conclusion

At equal final experience, a rollout queue can change the learner's update sequence and task-family outcomes. Independent first-moment intervention identifies a contributor at a capability cost. A per-update contract enforces declared composition at a waiting cost; broader learning benefit remains unconfirmed. Learning specifications should state both consumed experience and its assignment to updates.

REPRODUCIBILITY STATEMENT

The anonymous supplement contains frozen configurations and protocols, exact model and adapter revisions, complete seed registries, machine-readable raw results, artifact hashes, environment manifests, and fail-closed generators for every table and figure. It includes exact commands for the controlled learner, trace-replay, lifecycle, and audit paths. Appendix E summarizes the artifact map and audited environment. The public-code study additionally includes raw generated programs, original correctness outcomes, once-only verifier replays, and the qualified runtime. Historical source snapshots preserve the implementation identities of earlier experiments.

ETHICS STATEMENT

The experiments use generated arithmetic tasks, open model checkpoints, and the public τ^2, Spider, MBPP, GSM8K, and HumanEval benchmarks; no human subjects or private records are involved. The work identifies a way runtime policy can silently alter training and supplies an auditable mitigation. We report stopped experiments and nulls to avoid unsupported claims about production prevalence or quality.

AI USE STATEMENT

OpenAI Codex, a generative-AI coding and research assistant, was used substantially to propose and challenge hypotheses; design and preregister experiments; implement methods, tests, causal invariants, artifact packagers, and analysis code; orchestrate local runs; perform adversarial artifact and lineage audits; interpret results; search and summarize literature; create figures and tables; and draft and edit the manuscript. AI suggestions did not serve as experimental evidence: numerical claims are regenerated by deterministic scripts from hash-verified artifacts, and thresholds and seeds were frozen before their registered outcomes. The authors must complete final human review before submission and retain responsibility for the paper, code, citations, claims, and all AI-assisted content.

REFERENCES

Kavosh Asadi, Rasool Fakoor, and Shoham Sabach. Resetting the optimizer in deep rl: An empirical study. *arXiv preprint arXiv:2306.17833*, 2023. URL https://arxiv.org/abs/2306.17833.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. URL https://arxiv.org/abs/2108.07732.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*, 2025.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 41–48, 2009. doi: 10.1145/1553374.1553380. URL https://icml.cc/2009/papers/119.pdf.

Alexandra Carpentier and Rémi Munos. Finite time analysis of stratified sampling for monte carlo. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

Kaiwen Chen, Xin Tan, Jingzong Li, and Hong Xu. Libra: Efficient resource management for agentic rl post-training. *arXiv preprint arXiv:2606.03077*, 2026a.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Rongjian Chen, Jianmin Hu, Kejiang Ye, and Minxian Xu. Rolloutpipe: Overlapping pipelined rollout and training in disaggregated on-policy llm reinforcement learning. In *Proceedings of the 2026 International Conference on Cognitive Computing (ICCC 2026)*, 2026b. URL https://arxiv.org/abs/2606.26997.

Yujia Chen, Yang Ye, Xiao Chu, Yuchi Ma, and Cuiyun Gao. Schedule-and-calibrate: Utility-guided multi-task reinforcement learning for code LLMs. *arXiv preprint arXiv:2605.06111*, 2026c. URL https://arxiv.org/abs/2605.06111.

Zhoujun Cheng, Yutao Xie, Yuxiao Qu, Amrith Setlur, Shibo Hao, Varad Pimpalkhute, Tongtong Liang, Feng Yao, Zhengzhong Liu, Eric Xing, Virginia Smith, Ruslan Salakhutdinov, Zhiting Hu, Taylor Killian, and Aviral Kumar. Isocompute playbook: Optimally scaling sampling compute for LLM RL. *arXiv preprint arXiv:2603.12151*, 2026. URL https://arxiv.org/abs/2603.12151.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymyr Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 1407–1416. PMLR, 2018.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. Areal: A large-scale asynchronous reinforcement learning system for language reasoning. *arXiv preprint arXiv:2505.24298*, 2025.

Jinliang Gao, Ning Yang, Hai Wang, Baili Xiao, and Pin Lyu. Sdo: Structure-aware data organization for efficient llm post-training. *arXiv preprint arXiv:2607.27273*, 2026. URL https://arxiv.org/abs/2607.27273.

Wei Gao, Yuheng Zhao, Dakai An, Tianyuan Wu, Lunxi Cao, Shaopan Xiong, Ju Huang, Weixun Wang, Siran Yang, Wenbo Su, Jiamang Wang, Lin Qu, Bo Zheng, and Wei Wang. Rollpacker: Mitigating long-tail rollouts for fast, synchronous rl post-training. *arXiv preprint arXiv:2509.21009*, 2025.

Zetao Hong, Song Yuan, Yuanhao Ding, Yibo Zhu, Daxin Jiang, Zhibin Wang, and Chen Tian. Scheduling mixed rl rollouts beyond prefix locality. *arXiv preprint arXiv:2608.11152*, 2026. URL https://arxiv.org/abs/2608.11152.

Zhenyu Hou, Yujiang Li, Jie Tang, and Yuxiao Dong. Single-rollout asynchronous optimization for agentic reinforcement learning. *arXiv preprint arXiv:2607.07508*, 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Tianhao Hu, Xiangcheng Liu, Youshao Xiao, Yang Zheng, Xuan Huang, Jinrui Ding, Yufei Zhang, Tao Liang, Hongyu Zang, Quan Chen, Yueqing Sun, Wenjie Shi, Chao Zhang, Wei Wang, Qi Gu, Yerui Sun, Yucheng Xie, and Xunliang Cai. Dora: A scalable asynchronous reinforcement learning system for language model training. *arXiv preprint arXiv:2604.26256*, 2026.

Luke Huang, Zhuoyang Zhang, Qinghao Hu, Shang Yang, and Song Han. Stable asynchrony: Variance-controlled off-policy rl for llms. *arXiv preprint arXiv:2602.17616*, 2026.

Tianjin Huang, Ziquan Zhu, Gaojie Jin, Lu Liu, Zhangyang Wang, and Shiwei Liu. SPAM: Spike-aware adam with momentum reset for stable llm training. *arXiv preprint arXiv:2501.06842*, 2025. URL https://arxiv.org/abs/2501.06842.

Kang Liu and Suyan Li. Finite-horizon input-output dynamics of minibatch perturbations in adamw. *arXiv preprint arXiv:2608.19762*, 2026. URL https://arxiv.org/abs/2608.19762.

Zhizhao Liu, Zhiliang Tian, Xi Wang, Zhihua Wen, Yihang Xiong, Zhiquan Lai, and Dongsheng Li. Efficient rlvr scheduling via graph-structured online difficulty estimation. *arXiv preprint arXiv:2608.17941*, 2026. URL https://arxiv.org/abs/2608.17941.

Han Lu, Zichen Liu, Shaopan Xiong, Yancheng He, Wei Gao, Yanan Wu, Weixun Wang, Jiashun Liu, Yang Li, Haizhou Zhao, Ju Huang, Siran Yang, Xiaoyang Li, Yijia Luo, Zihe Liu, Ling Pan, Junchi Yan, Wei Wang, Wenbo Su, Jiamang Wang, Lin Qu, and Bo Zheng. Part ii: Roll flash—accelerating rlvr and agentic training with asynchrony. *arXiv preprint arXiv:2510.11345*, 2025.

Jian Lu. Periodic asynchrony: An effective method for accelerating on-policy reinforcement learning. *arXiv preprint arXiv:2511.18871*, 2025.

Ammar Mahran, Artavazd Maranjyan, and Peter Richtárik. Rescaled asynchronous sgd: Optimal distributed optimization under data and system heterogeneity. *arXiv preprint arXiv:2605.13434*, 2026.

Lam M. Nguyen, Quoc Tran-Dinh, Dzung T. Phan, Phuong Ha Nguyen, and Marten van Dijk. A unified convergence analysis for shuffling-type gradient methods. *Journal of Machine Learning Research*, 22(207):1–44, 2021. URL https://www.jmlr.org/papers/v22/20-1238.html.

Michael Noukhovitch, Shengyi Huang, Sophie Xhonneux, Arian Hosseini, Rishabh Agarwal, and Aaron Courville. Asynchronous rlhf: Faster and more efficient off-policy rl for language models. *arXiv preprint arXiv:2410.18252*, 2024.

Qwen Team. Qwen2.5-Coder-1.5B-Instruct model card. Hugging Face, 2024. URL https://huggingface.co/Qwen/Qwen2.5-Coder-1.5B-Instruct.

Yanyu Ren, Xizheng Wang, Xiao Liu, Bowen Lv, Hanchen Zhang, Shudan Zhang, Hanyu Lai, Shuai Wang, Li Chen, Dan Li, and Jie Tang. Tiderl: Boosting agentic rl goodput with readiness-aware scheduling. *arXiv preprint arXiv:2608.10402*, 2026. URL https://arxiv.org/abs/2608.10402.

Itay Safran and Ohad Shamir. How good is sgd with random shuffling? In *Proceedings of the Thirty-Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 3250–3284. PMLR, 2020.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yibo Shen, Xudong Han, Xiaowei Zhu, Gen Li, and Zhenxuan Pan. Routepack: Expert placement and attention-aware data packing for moe reinforcement learning. *arXiv preprint arXiv:2608.12146*, 2026. URL https://arxiv.org/abs/2608.12146.

Jingwei Song, Haofeng Xu, Jie Xiao, Chengke Bao, Jingwei Shi, Pengbin Feng, Weixun Wang, Yuhang Han, Chuan Wu, Linfeng Zhang, and Bill Shi. Staleness–learning rate scaling laws for asynchronous RLHF. *arXiv preprint arXiv:2607.01083*, 2026. URL https://arxiv.org/abs/2607.01083.

John Sweeney. Optimizer memory makes shuffle order a first-order source of fine-tuning noise. *arXiv preprint arXiv:2606.29554*, 2026. URL https://arxiv.org/abs/2606.29554.

Yujie Wang, Siwei Chen, Longzan Luo, Xinyi Liu, Xupeng Miao, Fangcheng Fu, and Bin Cui. Darts: Distribution-aware active rollout trajectory shaping for accelerating llm reinforcement learning. *arXiv preprint arXiv:2605.30859*, 2026.

Ryan Xu, Atlas Zhao, David Bao, and Frank Du. War: Workload-aware rollouts for synchronous agentic reinforcement learning. *arXiv preprint arXiv:2607.17299*, 2026. URL https://arxiv.org/abs/2607.17299.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

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

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3911–3921, 2018. doi: 10.18653/v1/D18-1425. URL https://aclanthology.org/D18-1425/.

Yiqi Zhang, Huiqiang Jiang, Xufang Luo, Zhihe Yang, Chengruidong Zhang, Yifei Shen, Dongsheng Li, Yuqing Yang, Lili Qiu, and Yang You. Sortedrl: Accelerating rl training for llms through online length-aware scheduling. *arXiv preprint arXiv:2603.23414*, 2026a.

Zili Zhang, Yinmin Zhong, Chengxu Yang, Chao Jin, Bingyang Wu, Xinming Wei, Yuliang Liu, and Xin Jin. Heddle: A distributed orchestration system for agentic rl rollout. *arXiv preprint arXiv:2603.28101*, 2026b.

Peilin Zhao and Tong Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1–9. PMLR, 2015.

Linghao Zhu, Yiran Guan, Dingkang Liang, Jianzhong Ju, Zhenbo Luo, Bin Qin, Jian Luan, Yuliang Liu, and Xiang Bai. Shuffle-r1: Efficient rl framework for multimodal large language models via data-centric dynamic shuffle. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026a. URL https://arxiv.org/abs/2508.05612.

Ximo Zhu, Ruiqi Liu, Rong Wang, Ping Wu, Xiang Zheng, Wenzhuo Xu, Xubin Yao, Zhiyuan Yan, Bo Li, Jun Gao, and Xiaolei Lv. Reorder-opd: Reliability-aware prompt ordering for on-policy distillation. *arXiv preprint arXiv:2608.10905*, 2026b. URL https://arxiv.org/abs/2608.10905.

Appendix guide. Appendix A gives the formal guarantees; Appendix B supplies the controlled-learning protocols and full results; Appendix C describes admission and measured traces; Appendix D reports every material failed or inconclusive extension. Appendix E maps these claims to the anonymous artifact. Extended execution histories are in `paper/artifact_details/README.md`; their relocation changes no outcome.

A Formal guarantees and their limits

A.1 Exact batch-path decomposition

Let $x_t = (\theta_t, o_t)$ contain the parameters and all optimizer state, including Adam moments and step count. Conditional on matched per-update randomness, write an update on complete batch B as $x_{t+1} = F_t(x_t; B) = x_t + u_t(x_t; B)$. Consider batch paths P and Q with a common x_0, and define

$$\begin{aligned}\delta_t &= x_t^Q - x_t^P,\\ D_t &= u_t(x_t^P; B_t^Q) - u_t(x_t^P; B_t^P),\\ R_t &= u_t(x_t^Q; B_t^Q) - u_t(x_t^P; B_t^Q).\end{aligned}$$

Subtracting the two state recursions gives

$$\delta_{t+1} - \delta_t = u_t(x_t^Q; B_t^Q) - u_t(x_t^P; B_t^P).$$

Adding and subtracting $u_t(x_t^P; B_t^Q)$ proves the exact identity

$$\delta_{t+1} = \delta_t + D_t + R_t, \qquad \delta_T = \sum_{t=0}^{T-1} (D_t + R_t). \tag{6}$$

D_t is a matched-state batch-replacement term and R_t is feedback through the changed learner state. If $u_t(\cdot; B_t^Q)$ is L_t-Lipschitz, then $\|R_t\| \le L_t \|\delta_t\|$ and hence $\|\delta_{t+1}\| \le (1 + L_t)\|\delta_t\| + \|D_t\|$. Induction from $\delta_0 = 0$ gives

$$\|\delta_T\| \le \sum_{t=0}^{T-1} \|D_t\| \prod_{s=t+1}^{T-1} (1 + L_s). \tag{7}$$

Equal eventual membership, $\biguplus_t B_t^P = \biguplus_t B_t^Q$, does not force either sum to vanish. Conversely, if corresponding canonicalized batches are identical, then every $D_t = 0$; since $\delta_0 = 0$, induction gives identical extended learner state at every checkpoint. This is the condition implemented by our canonicalized fixed-batch control. The decomposition neither predicts a capability direction nor claims that D_t and R_t were separately measured.

A.2 Exact allocation and earliest release

Order the declared strata as $g_1, \dots, g_G$. For a fixed eligibility snapshot, define

$$C_i(k) = \min \sum_{r=1}^{i} |n_{g_r}/K - p_{g_r}|$$

over integer counts satisfying $\ell_{g_r} \le n_{g_r} \le N_{g_r}$ for $r \le i$ and $\sum_{r=1}^{i} n_{g_r} = k$; an empty feasible set has value $+\infty$. The implementation initializes $C_0(0) = 0$ and no other state. For stratum g_i, it enumerates every integer $c \in [\ell_{g_i}, \min\{N_{g_i}, K - k\}]$ and updates

$$C_i(k + c) = \min\{C_i(k + c), C_{i-1}(k) + |c/K - p_{g_i}|\}.$$

Induction on i shows that the retained state for every selected count is the minimum partial ℓ_1 error over all feasible prefixes: the base case is immediate, and the recurrence partitions every feasible length-i allocation by its last count c. Therefore $C_G(K)/2$ is exactly the minimum declared- stratum TV among coverage-preserving size-K allocations. Absence of state K is equivalent to infeasibility. The stored count tuple only resolves equal-cost allocations lexicographically and cannot change the optimum.

For release time, each capacity $N_g(\tau)$ is a nondecreasing step function whose changes occur only when a complete prompt group becomes eligible. The minimum achievable TV and coverage feasibility are consequently constant between consecutive eligibility times. The standalone controller evaluates the minimum-observation boundary, every distinct eligibility time no later than the deadline, and the deadline in chronological order. The first snapshot satisfying coverage and the TV budget is therefore the earliest feasible release; if none satisfies them, the deadline error is correct. The live polling implementation has the same logical guarantee at poll granularity.

A.3 Declared composition and within-stratum selection

At a fixed update, let $\mathcal{P}_t^B$ be an intended law over complete batches and $\mathcal{Q}_t^B$ the law induced by rollout scheduling and admission. For complete update functional $U_\theta(B)$, conceptual drift is

$$\Delta_t(\theta) = \mathbb{E}_{B\sim\mathcal{Q}_t^B} U_\theta(B) - \mathbb{E}_{B\sim\mathcal{P}_t^B} U_\theta(B). \tag{8}$$

This formal object is broader than what our system can measure. We use it to locate the semantic boundary, not to claim estimation of a full batch law.

This is a projection, not a full-law guarantee. Two groups with the same label may differ in content, quality, or availability. A useful decomposition makes the limitation explicit. For atomic unit Z, label $G = s(Z)$, and signal $v(Z)$, the difference between realized and intended expectation is

$$\sum_g (\pi_g^Q - \pi_g^P)\mu_g^P + \sum_g \pi_g^Q(\mu_g^Q - \mu_g^P), \tag{9}$$

where π_g is label mass and μ_g the conditional signal mean. The contract constrains the first term relative to declared target p_t; coarse strata do not control the second. Strata and p_t are versioned parts of the learning specification, not values inferred from treatment outcomes or a certificate that balance is optimal or fair.

B Controlled learning: protocols and full results

B.1 Tasks, learner, and matched experience

Both arithmetic studies use a fixed Qwen3-0.6B source adapter and the same two deterministic word-problem templates. Product-chain problems ask for the remaining items after multiplying crates, boxes, and items per box ($abc-d$); shared-rate problems combine two daily production rates over a common number of days and subtract discarded items ($ad+bd-c$). Each prompt includes an irrelevant time quantity. The model supplies a calculator expression; the verifier executes bounded integer arithmetic and checks the target result. These are two synthetic task templates, not broad reasoning benchmarks.

Within a training seed, all paths share the source checkpoint, task prompts, four recorded completions per prompt, rewards, token likelihoods, initialization, and total token exposure. Sixteen complete groups form eight two-group updates. Table 2 records the common settings and the two independent evaluation cohorts. Source model and adapter hashes, initialization seeds, prompts, and complete trajectories are included in the artifact.

Recorded FP32 full-softmax scores are a surrogate source density, not exact propensities under the top-p sampling distribution. Each paired comparison holds this convention fixed and concerns the implemented update; it does not establish exact off-policy correction for filtered sampling. Historical source snapshots preserve the learner implementations used by the respective studies.

B.2 Prospective order reversal

The follow-up protocol freezes six unused training seeds and three paths: balanced mixed batches, product→shared blocks, and shared→product blocks. Within seed, all paths use the same 16 prompt groups, 64 recorded trajectories, behavior/reference policy, total active-token exposure, eight optimizer steps, initialization rule, and four held-out evaluation panels. Per-update randomness is matched by step. The two blocked paths reverse only task-family order; the balanced path changes both partition and order relative to either blocked path. Every path assigns two groups to each step

Table 2: Settings shared by the arithmetic order and first-moment studies, with distinct training seeds and evaluation panels.

Component	Frozen setting
Adapter	LoRA rank 8, scaling 16, dropout 0; attention Q/K/V/O projections
Optimizer	Persistent Adam, learning rate 10^{-5}, betas $(.9, .999)$, zero weight decay
GRPO	KL coefficient .01; token-ratio clipping .2; within-prompt centering, no reward-SD scaling
Source generation	Four completions per prompt; 64 tokens; temperature 1.4; top-$p = .98$
Truncation	Exact-budget completions retained and neutralized by the frozen NaN reward mask
Order-reversal cohort	Seeds 4153, 4253, 4357, 4457, 4561, 4663; four common 64-task panels
First-moment cohort	Six unused seeds in Table 4; four new 128-task panels
Endpoint	Update-8 greedy calculator-execution success, balanced by task family

0–7, so marginal source lag is identical; the blocked paths exchange family-conditioned mean lags 1.5 and 5.5, while the balanced path gives each family mean lag 3.5.

Panel-denominator clarification. The frozen protocol prose mistakenly described "four 256-task panels." The already-frozen executable configuration set 256 tasks total across four evaluation seeds, and the unchanged evaluator therefore generated four 64-task panels. The executable configuration and evaluator were unchanged; no seed, task, checkpoint, endpoint, bootstrap rule, or decision rule was altered.

Let $Y^{(8)}_{s,r,c,f}$ be update-8 exact success for training seed s, evaluation panel r, condition c, and family f, and define

$$I_{s,r,c} = Y^{(8)}_{s,r,c,\text{product}} - Y^{(8)}_{s,r,c,\text{shared}}.$$

With $S = 6$ training seeds and $R = 4$ panels, the direction-free registered primary estimand is the finite-sample mean

$$\tau_{\text{rev}} = \frac{1}{SR} \sum_{s=1}^{S} \sum_{r=1}^{R} \left[I_{s,r,\text{shared}\to\text{product}} - I_{s,r,\text{product}\to\text{shared}} \right]. \tag{10}$$

The analysis also reports each blocked-minus-balanced family effect and the prespecified secondary mean of those two contrasts. This mean is descriptive, not an identified pure blocking effect. The point estimate is the finite arithmetic mean in Equation 10. Each of 10,000 crossed empirical bootstrap draws resamples six paired training-seed rows and four common evaluation-panel columns with replacement; a common column resample is retained across paired conditions and families. The interval summarizes stability of the frozen matrix. It does not treat the 24 cells as independent, and it is not a population-generalization confidence interval. The blocked-minus-balanced family effects and their mean are pointwise, multiplicity-unadjusted secondary summaries. Checkpoints 2, 4, and 6 are path diagnostics rather than equal-cumulative-exposure endpoints; equal membership holds at update 8.

Table 3: Prospective update-8 effects in percentage points. Brackets are 95% crossed empirical stability intervals. Secondary family intervals are pointwise and multiplicity-unadjusted. The primary row is the registered reverse-minus-forward change in the product-minus-shared contrast.

Contrast	Product-chain	Shared-rate	Aggregate
Product→shared − balanced	2.9 [0.5, 5.9]	-5.1 [-8.6, -1.8]	-1.1 [-3.4, 1.1]
Shared→product − balanced	-10.5 [-14.6, -6.8]	5.1 [1.7, 9.1]	-2.7 [-4.7, -0.5]
Mean of blocked − balanced contrasts	-3.8 [-6.3, -1.3]	0.0 [-1.8, 1.9]	-1.9 [-2.9, -1.0]

Registered reverse-minus-forward family contrast: -23.6 [-30.5, -16.3].
Seed directions: negative in all 6 training seeds; descriptive exact sign $p = 0.03125$.

The six paired-seed primary means are -18.8, -28.9, -27.3, -22.7, -18.8, -25.0 points. The reversal concerns the schedule-induced deviation from balanced, not the absolute product-minus-shared

ranking, which remains positive in all three conditions. Seed directions are descriptive paired summaries; the 24 seed–panel cells are not treated as independent replicates. The contrast is negative in all 24 seed–panel cells. Seed means span -28.9 to -18.8 points and panel means span -29.2 to -13.0 points. Leave-one-seed-out and leave-one-panel-out means span -24.5 to -22.5 and -27.1 to -21.7 points, respectively. With six nonzero paired seed means, the descriptive exact sign $p = 0.03125$ is the smallest attainable two-sided value. All 96 corresponding canonical fixed-batch checkpoints are byte-identical. Continuous paths differ despite equal final membership, active tokens, initialization, and eight optimizer steps. These controls isolate the total batch path, including family-conditioned learner age and optimizer state, rather than a single optimizer submechanism.

Earlier development cohort. The preceding eight-seed balanced-versus-product-first study motivated the reversal; it is not an independent model or environment replication. Its blocked-minus-balanced product, shared, and aggregate effects are $+4.2$ [1.1, 7.4], -7.9 [-10.8, -4.9], and -1.9 [-4.4, 0.7] points. All 32 corresponding fixed-batch checkpoints are exact. Full seed rows remain in the archived experimental record; the prospective reversal above uses unused seeds and carries the primary order claim.

B.3 Independent first-moment intervention

Motivation and development. Post-hoc inspection of 144 saved Adam states motivates resetting first-moment memory at the family boundary. A frozen two-seed development experiment gives $+13.67$ points of attenuation, with an aggregate cost. Its original finalizers fail because legacy corpus manifests lack two hash fields; a separate compatibility audit fills only missing fields from precommitted hashes. That disclosed exception applies only to development. It neither changes old manifests nor supplies confirmation evidence. The new six-seed study below uses fresh corpora and valid current-schema manifests.

Design and decision rule. The confirmation crosses three schedules with retained continuous Adam, continuous Adam with only its first moment zeroed immediately before update 5, and unmodified canonical controls: 54 branches and 432 updates. The reset preserves parameters, second moments, and optimizer clocks; the balanced path receives the same reset as a matched control. Before learning, all 384 source completions reach EOS and each seed/family has at least three reward-variable groups, exceeding the frozen minimum of two. Source, settings, and likelihood conventions are given in Table 2.

Four new 128-task panels measure update-8 success. For each seed and panel, $D_a = (P - S)_{\mathrm{reverse},a} - (P - S)_{\mathrm{forward},a}$ and $I = D_{\mathrm{reset}} - D_{\mathrm{retain}}$. The registered criteria require mean $I \geq .05$, a positive crossed 95% lower bound, at least five positive seed means, and retained mean $D \leq -.05$ to verify the parent effect. There are 10,000 crossed seed–panel draws with bootstrap seed 24050003. The six seeds are the training replicates; the 24 cells are not independent.

Full results and exact controls. The decision is *confirmed*. Retained and reset contrasts are -35.09 [-41.02, -29.23] and -13.22 [-17.71, -9.11] points. Attenuation is +21.88 [+16.54, +27.21], positive in 6/6 seeds. All 168 declared tensor/optimizer pairs are exact: 96 fixed-batch timing controls and 72 retained/reset controls through update 4. Each reset occurs once and every optimizer clock runs from 1 to 8.

Table 4: First-moment confirmation: every paired training-seed mean.

Seed	Retained contrast (pp)	Reset contrast (pp)	Interaction (pp)
924101	-29.30	-14.84	+14.45
924203	-30.86	-9.77	+21.09
924317	-31.64	-16.80	+14.84
924421	-43.75	-16.02	+27.73
924527	-40.62	-16.41	+24.22
924631	-34.38	-5.47	+28.91

The residual order contrast leaves second-moment, learner-age, and other stateful effects open. This intervention tests first-moment dependence, not a mediated fraction or an aggregate improvement.

Table 5: Reset-minus-retain capability effects. Intervals are pointwise and multiplicity-unadjusted. The first family loses its advantage, without a second-family benefit; aggregate success decreases in every schedule.

Schedule	Endpoint	Reset − retain (pp)	95% lower	95% upper
Balanced	Overall	-6.51	-8.20	-4.72
Balanced	Product	-7.88	-9.70	-5.53
Balanced	Shared	-5.14	-8.07	-2.34
Product → shared	Overall	-7.58	-9.64	-5.44
Product → shared	Product	-13.80	-17.19	-10.55
Product → shared	Shared	-1.37	-2.86	+0.78
Shared → product	Overall	-5.44	-7.03	-3.81
Shared → product	Product	-0.72	-1.56	-0.13
Shared → product	Shared	-10.16	-13.28	-6.90

Order sensitivity and optimizer resets already have substantial prior art (Sweeney, 2026; Asadi et al., 2023; Huang et al., 2025).

C Admission baselines and measured systems

C.1 Simple queues, buffering, and readiness

Six constructed-service 3B traces and their six equal-duration controls each contain 16 immutable groups. Every policy consumes all groups once in eight updates. A capacity-four/eight random buffer fills in arrival order, selects two groups uniformly without replacement, refills, and drains its tail. Each capacity uses 256 fixed seeds, 23900000–23900255; means weight traces equally. These are conditional computational draws, not new live replicates.

For two equally weighted families and batch size two, coverage and cardinality force the unique quota $(1, 1)$. One queue per family with matched stable-hash tie-breaking therefore gives exactly the same selected IDs and earliest readiness as the general allocator in all 12 traces. Wait-all retains those batches but delays them until the last group is ready. Main Table 1 reports TV, missing-family fraction, and the difference in families' mean assigned update indices. Readiness excludes learner compute, checkpoint publication, and live contention.

C.2 Allocator and durable-event costs

The CPU matrix crosses $G \in \{2, 8, 32\}$, $K \in \{32, 64, 128\}$, pools of $2K/8K$, uniform/0.7-head targets, and feasible/missing coverage. On one pinned i7-6700K core, 7,200 decisions over 72 cells have a median of cell medians of 686 μs and worst-cell P99 40.9 ms. Table 6 gives the paper's smaller operating points, each with 800 timed decisions over eight cells. Durable events use one

Table 6: CPU controller latency at the experimental coordinates. Times are μs; P99 is the worst cell.

(G, K)	Median	P99	Hash	Decisions/s
$(4, 16)$	150.1	1067.8	113.9	7,265
$(2, 4)$	46.9	109.5	29.0	23,358

JSON append, flush, and `fsync` per record on local ext4. After 50 warmups, 1,000 appends have median/P95/P99 latencies 2.34/2.65/8.02 ms (393 events/s). Adding a durable batch record preserves selected IDs and adds 2.16–2.27 ms median. One-million-event replay takes 14.598 s with 370.8 MiB peak RSS. Strict recovery rejects an incomplete last record; explicit tail recovery ignores only that final unterminated record. Interior corruption, duplicate events, incomplete lifecycles, and consumption of rejected/cancelled groups remain errors. These local measurements do not establish production-scale throughput; full latency, storage, and replay tables are retained in the artifact documentation.

C.3 Natural multi-turn traces and duration controls

Three Qwen3-8B actors and a Qwen2.5-3B user simulator execute the same 32-task τ^2 panel three times. Each panel contains eight tasks per domain–complexity stratum. These are repeated service measurements on identical task IDs, not independent workload samples; no 8B learner update is performed.

Table 7: Repeated four-GPU Qwen3-8B τ^2 execution (ranges). Coverage is the number of direct orders containing every positive-target stratum. Waste is counterfactual excluded service spent by release.

Policy	Release (s)	Hist. TV	Coverage	Waste (%)
Arrival-first-8	62.71–89.16	0.125–0.375	2/3	19.8–49.8
Bounded TV .125	76.76–89.16	0.125	3/3	19.8–39.5
Zero TV	107.67–124.12	0	3/3	28.1–49.1
Consume all	272.24–328.74	0	3/3	0

For each trace, 4,096 common dispatch orders are replayed with measured, equal, and 1,024 permuted duration assignments using three workers and fixed tie-breaking. Duration permutations preserve the entire duration multiset and break its association with task identity. Exact uniform size-eight sampling already has 27.24% probability of omitting a stratum, so a raw missing-stratum event alone is not a timing effect.

Table 8: Arrival-first-8 duration-null results, in percent except mean-TV difference. Null is the mean over 1,024 duration permutations. All coverage tests have one-sided Monte Carlo $p = 0.0010$. Clear excess requires both coverage and mean TV beyond equal-duration and adverse 95% permutation bounds, at least five coverage-loss points and 0.025 TV beyond the permutation means, and no contradictory median-TV or exact-composition result.

Trace	Obs. loss	Null loss	Excess	TV Δ	Exact
1	35.0	26.9	8.1	0.0207	4.6
2	33.4	26.9	6.5	0.0205	4.0
3	34.2	26.9	7.3	0.0211	4.5

Measured durations add 6.5–8.1 coverage-loss points relative to the permutation means, with $p = 0.0010$ per trace. Mean-TV excess, 0.0205–0.0211, misses the frozen 0.025 rule, and median TV is unchanged. The full joint gate fails in every trace. This supports a bounded duration-associated coverage component, not population prevalence or an 8B learning effect. Waste in Table 7 is service spent by release on excluded or unfinished groups under a counterfactual cancellation; it is not measured training-time savings.

D Independent workloads and unsuccessful extensions

Table 9 separates learning evidence from the gates that enable it. A failed validity rule is not a valid scientific null; a failed support or timing screen supplies no learner comparison. Every frozen outcome is retained, including the operational failures summarized below.

D.1 Measured-service 3B learning

Six unused Qwen2.5-3B seeds each supply 16 complete groups and 64 immutable trajectories. Identity-bound measured calculator/hash service and equal-duration controls cross with FIFO, exact 1:1 composition, and wait-all/fixed batches: 36 branches and 288 updates. The tools construct the service contrast. The update-8 primary is $(\text{fast} - \text{slow})_{\text{FIFO}} - (\text{fast} - \text{slow})_{\text{contract}}$, on 128 new tasks with 4 common stochastic draws. Confirmation requires at least three points, a positive crossed family-stratified-task lower bound, and five positive seed means; only the last criterion passes.

Table 9: Scope of the independent and earlier boundary studies. None supplies a confirmed cross-workload learning benefit. The artifact guide indexes the full protocols and execution histories.

Study	Terminal decision	Interpretation
Measured-service 3B	Not confirmed; metadata deviation	Primary effect misses magnitude and interval rules.
MBPP code pilot	Invalid; numerical screens also fail	Descriptive outcomes cannot establish replication or absence of order effects.
Public Spider	Support and timing no-go	Insufficient mixed-reward groups; no learning launched.
Public math/code mixture	Gradient-premise and timing no-go	Source support passes; accuracy prediction remains untested.
Earlier live 3B	Inconclusive endpoint	Membership and lineage work; quality intervals cross zero.
τ^2 learning path	Reward/semantic no-go	SFT support and partial-reward equivalence fail; no GRPO.
Sequential SQL	Timing and identity no-go	Reward support passes; availability contrast and global-ID gate fail.
3B source V1/V2	Truncation/support no-go	Closed source recipes; no learner or evaluation panel.

Disclosed validity boundary. The original literal byte gate remains **FAILED**: only 16/144 complete checkpoint pairs match. The remaining pairs differ solely in serialization order of PEFT's set-valued `target_modules`. A separate compatibility audit, frozen after training diagnostics but before held-out evaluation, permits only that permutation; all 144 model/optimizer scientific-state pairs and all six contract/reference held-out controls are exact. It does not convert the strict failure into a pass. All outcomes here carry *`WITH_DISCLOSED_PROTOCOL_DEVIATION`*. The source V1/V2 support failures, scoring-geometry implementation retry, compatibility retry, and their ordering are preserved in the artifact record.

Table 10: 3B outcomes with the disclosed metadata deviation. Primary and secondary effects are FIFO minus contract (pp); secondary intervals are pointwise. Absolute success is percent. Contract/reference outcomes are exact.

Effect	Mean	Crossed 95% interval
Primary fast–slow interaction	+2.08	[-1.11, +5.34]
Fast family	+0.07	[-2.41, +2.73]
Slow family	-2.02	[-4.75, +0.46]
Aggregate	-0.98	[-3.03, +0.94]

Absolute success	Fast	Slow	Aggregate
Source	77.34	82.03	79.69
FIFO	83.01	84.83	83.92
Contract / reference	82.94	86.85	84.90

The six primary seed effects are $+3.52$, -1.56, $+4.69$, $+3.52$, $+1.56$, and $+0.78$ points in registered order. The terminal result is `NOT_CONFIRMED` at the frozen eight-update horizon; no aggregate or time-to-quality benefit is established.

D.2 Public Spider source calibration

Hash selection fixes 32 official Spider training questions across four database domains, labeled by one versus two/three top-level `FROM` tables. Qwen2.5-3B generates four completions each (temperature 1, full softmax, 192 tokens). Gold SQL/results are hidden. A bounded read-only verifier checks result equality on the original database, with ordered rows only when required; this is not official test-suite accuracy. All 128 outputs reach EOS and pass identity/score checks. Success is 60.94%/54.69%, but mixed-reward groups are 1/16 and 2/16, below four per stratum. Measured-minus-equal TV is -0.00446 ($p = 0.9961$); both support and conditional timing gates fail. No learner follows. A separate outcome audit agrees with all 128 rewards.

D.3 PUBLIC PYTHON CODE: AN INVALID LEARNING PILOT

Official MBPP validation IDs calibrate a pinned Qwen2.5-Coder-1.5B source; training IDs and disjoint test panels are selected before learner outcomes (Austin et al., 2021; Qwen Team, 2024). Reference-AST iteration nodes define explicit-iteration (E) and no-explicit-iteration (N) strata; these are syntax labels, not semantic domains or labels of the model's chosen solution. Prompts expose the problem and first assertion, with gold code hidden; success requires all original assertions. This is custom one-shot prompting, not standard leaderboard evaluation; pretraining contamination is possible.

Validation uses 16 tasks per stratum and four completions each (384 tokens, temperature 1, full softmax). Reward support passes, with 8/16 and 5/16 mixed-EOS groups. Conditional timing fails (TV excess -0.004883, $p = 0.9572$). Before any pilot output, an official reference timeout is preserved and a uniform 30-second isolated execution budget is qualified on all 328 selected references; all 128 original calibration rewards remain exact. This verifier qualification does not reopen the failed timing gate.

Four unused corpus seeds each fix 64 training tasks and four completions per task. Balanced and both blocked releases cross with canonical batches, FIFO, and a capacity-eight random buffer: 36 branches, 32 persistent-GRPO steps each, two complete groups per step, rank-8 LoRA, learning rate 10^{-5}, KL .01, and clipping .2. All paths consume the same recorded experience once. The held-out panel has 64 official test tasks. The primary FIFO contrast is $D = (E - N)_{N\text{-first}} - (E - N)_{E\text{-first}}$; advancement requires strict validity, $|\overline{D}| \geq .05$, and three matching signs among four seeds.

Validity remains failed. All 1,152 updates execute, but both original finalizers fail: a balanced-reference reporting lookup and three nested reference-adapter configuration permutations following transactional recovery. All 384 top-level configuration/tensor/Adam pairs match; only 381/384 full artifact pairs do. The frozen protocol forbids a retrospective serialization exception. The original failed manifests and differing bytes are retained. A once-only verifier replay agrees with all 2432 correctness observations, but cannot repair this failed learner-validity gate.

Descriptive outcome and limit. The interaction is +0.781 points, with seed values +0.000, +3.125, -3.125, +3.125. Both numerical screens fail as well. No reserved confirmation, replacement seed, model, or horizon follows. Table 11 retains all family/aggregate summaries. The local buffer mixes neighboring arrivals while retaining most global block order; it shows no aggregate advantage. This invalid pilot establishes neither cross-workload replication nor the absence of code-order effects.

Table 11: Descriptive MBPP success (%) under failed pilot validity. Learned rows average four seeds on the same 64 test tasks. These are not confirmatory estimates. Canonical outcomes equal balanced FIFO in every release condition.

Source or schedule/route	Explicit	No explicit	Aggregate
Base source	43.75	62.50	53.12
Balanced FIFO	44.53	64.06	54.30
$E \to N$ FIFO	44.53	64.84	54.69
$N \to E$ FIFO	44.53	64.06	54.30
Balanced Buffer 8	44.53	64.06	54.30
$E \to N$ Buffer 8	43.75	63.28	53.52
$N \to E$ Buffer 8	43.75	64.06	53.91

D.4 INDEPENDENT GRADIENT PREMISE: STOPPED BEFORE LEARNING

A bounded follow-up asks whether source-gradient geometry could guide the selection of a consequential order experiment. Finite-horizon AdamW response already has a parameter-and-moment treatment (Liu & Li, 2026). Our diagnostic is neither a new optimizer theory nor an accuracy forecast. Development measures 48 arithmetic and 128 MBPP minibatches at reset source states, plus 20 exact gradient/updated-weight repeats. The MBPP learning pilot remains invalid. Current arithmetic measurements differ numerically from the historical learner and are separate development diagnostics.

Split each family's minibatches into odd/even subsets. Let d_o, d_e be the differences of family-mean gradients in those subsets. For B measured minibatch gradients g_j, define $G = \langle d_o, d_e \rangle / (B^{-1} \sum_j \|g_j\|^2)$. Development means are 1.00639 for arithmetic and -0.01556 for MBPP. This split reduces self-inner-product noise but leaves sampling uncertainty; it does not calibrate held-out learning effects.

The independent pinned 3B source combines 16 GSM8K questions rendered as Python `solve()` programs with 16 HumanEval functions, four completions each (Cobbe et al., 2021; Chen et al., 2021). Gold answers/solutions stay hidden; GSM8K uses numeric answer equality and HumanEval its original executable tests. These are custom formats/splits, with possible pretraining contamination. All 328 selected references pass the fixed two-second verifier. Validation success is 76.56%/54.69%, with 7/16 and 9/16 mixed-reward groups; all 128 outputs reach EOS. Source support passes, while the conditional timing gate fails (TV excess -0.000305, $p = 0.45914$). Training and reserved evaluation pools are never model-evaluated in this study.

Before measuring new gradients, the protocol freezes $G \geq .20$ under both unused adapter seeds, 25095101 and 25095203, as a heuristic advancement screen. All 32 unique probes and four exact repeats finish, with valid source-likelihood attestations and zero precision-clipped tokens. Scores are -0.00580 and -0.00231, so the valid decision is a failed gradient premise. Both initializations share one corpus; they are not task replications. No learning or recipe rescue follows. This outcome neither proves order-insensitive learning nor validates a low-effect forecast.

D.5 Earlier boundaries and systems-only evidence

The earlier 3B integration verifies lifecycle and distinct model states, but its deterministic endpoint saturates. A frozen stochastic follow-up is inconclusive: contract minus wait-all is 2.2 [-5.7, 10.4] points; FIFO minus contract is -2.7 [-7.6, 1.4]. Both intervals cross zero. Its critical-path accounting and separate equal-work multi-GPU scaling study establish systems measurements, not admission quality gains; their figures and complete measurements remain in the artifact.

The τ^2 SFT-to-GRPO path stops because no candidate meets reward-support rules and the alternative partial reward assigns full reward to 2/128 episodes that fail official success. Sequential SQL passes reward support but fails all four seeds' frozen timing-excess rule and repeats candidate IDs across seeds, violating the global-ID criterion. Neither launches the proposed learner. These failures are retained separately from the public Spider calibration.

E Reproduction and evidence map

The anonymous supplement contains the implementation, original compact source and evaluation rows, frozen protocols/configurations, source identities, licenses, and exact reproduction commands. Three immutable inventories bind later studies: `REVISION_EVIDENCE.json` covers first-moment confirmation and Spider; `CODE_EVIDENCE.json` covers MBPP and its failed validity; `PREDICTOR_EVIDENCE.json` covers the independent failed gradient premise. Earlier reverse-order and natural-3B evidence retain their own audited bindings.

From the archive root, follow its `README.md` to verify inputs and build `papers/iclr2027_batch_contracts/build.sh` in a clean extraction. The release script checks every generated table, vector figure, and PDF by hash after deleting the derived outputs. This CPU reconstruction reuses the recorded Python and Tectonic runtime; it does not rerun GPU training. Original model and source-adapter revisions, historical learner snapshots, packages, hardware, all seeds, and full GPU launch commands are in the included protocols and run manifests. Large learned and optimizer tensors are hash-indexed and kept separate from the compact archive.

`paper/artifact_details/README.md` maps each paper claim to its protocol, raw data, and analysis, and explains the relocated execution histories. Exact copies of the previous expanded notes are retained there for auditability. Important negative outcomes and deviations remain summarized in Appendix D; no result, threshold, or validity classification changes with this shorter presentation.