# STRUCTURED REWARD RECONSTRUCTION FOR GROUP-NORMALIZED POLICY LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Group-normalized policy updates couple the rewards of all sampled completions. This coupling makes selective verification difficult: an unknown reward changes the update weights even for verified answers. For tasks with one canonical correct answer, we show that the complete update has a finite representation with one state per distinct answer class and one none-correct state. The representation makes classical prediction-correction estimators applicable to this nonlinear target and supports an optimal projection on its observation channel. Public coefficient vectors guide query selection, and detached reconstruction weights implement the estimate through an ordinary language-model loss. Online calibration adapts the reference distribution under a statewise covariance constraint. Experiments on GSM8K use three models, five seeds and a common training recipe. At 400 updates, generation-matched comparisons show fewer new checks than full feedback under shared stopping and answer-cache rules. Half-batch full feedback offers a competitive lower-generation alternative. Fixed-history analyses distinguish query-design effects from prediction correction and statewise covariance control; cost models quantify when fewer checks can offset additional computation.

## 1 INTRODUCTION

A policy can generate several candidate answers before their correctness is known. Verifying every candidate gives a complete learning signal, but selective verification provides another way to allocate supervision: generate a group, check a subset, and estimate the update that full feedback would have produced. The difficulty is that group-relative normalization couples rewards. Replacing an unobserved reward by zero can change the weight of every completion, including those whose rewards were observed.

We study this problem for questions with a single canonical answer. Consider the generated answers $(42, 42, 17, \text{invalid})$. There are only three possible reward vectors: $(1, 1, 0, 0)$, $(0, 0, 1, 0)$, and $(0, 0, 0, 0)$. Checking one occurrence of 42 verifies both duplicate answers. A positive result identifies the entire reward vector; a negative result excludes one of its three possibilities. The nonlinear reward normalization therefore has only three possible target values.

This finite representation makes the target linear in mutually exclusive state indicators. A generalized-difference estimator can therefore correct predictions using selected class labels, even though the original update is nonlinear in the reward vector. A projection on the full observation channel characterizes the minimum reference error among all unbiased reconstructions on that channel. The reference guides variance reduction; the expectation remains correct for each individual reward state.

The procedure is inexpensive to integrate into policy learning. Once a query outcome determines the coefficients, their detached weighted log likelihood gives the estimated update in ordinary weighted backpropagation. Query design uses a public coefficient matrix instead of extracting all individual completion gradients. Online calibration updates its reference from answers already verified during training; a covariance constraint controls how far calibration changes the current estimator.

Our experiments fix optimizer steps and generated completions for the main comparison, and include full-feedback controls with smaller groups and batches. Fixed-history calculations separate



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

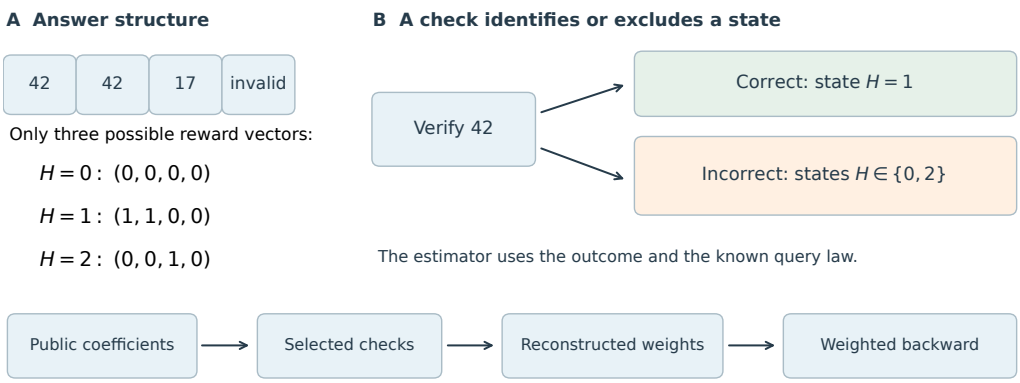


Figure 1: Selective verification with mutually exclusive answers. Duplicate completions share one label. A successful check identifies the reward state; a failed check excludes a state. The randomized observation is mapped to unbiased slot weights and implemented by ordinary weighted backpropagation.

prediction correction, query dependence and calibration. Learning accuracy, feedback calls and model computation are evaluated together, including cases where simpler full feedback costs less.

The contribution is a structured reconstruction method for the complete group-normalized update, together with a practical implementation and an empirical comparison under explicit resource budgets. Its mathematical foundation combines finite-channel projection and optimal experimental design with the mutually exclusive answer structure (Horvitz & Thompson, 1952; Cohen & Kaplan, 2011; Sagnol & Pronzato, 2023).

## 2 A FINITE STATE SPACE FOR THE COMPLETE UPDATE

Fix one question, $K$ sampled completions $x_1, \dots, x_K$, and policy parameters $\theta$. Write $s_i = \nabla_\theta \log \pi_\theta(x_i)$ for the completion-sum score, and stack these scores in $S \in \mathbb{R}^{K \times d}$. The full-feedback target is

$$f(r) = \sum_{i=1}^{K} c_i(r) s_i, \qquad c_i(r) = \frac{r_i - \bar{r}}{K(\sigma(r) + \epsilon)}, \qquad \sigma(r)^2 = K^{-1} \sum_i (r_i - \bar{r})^2, \tag{1}$$

with $\epsilon = 10^{-6}$. This is the current-policy score update with group-relative reward normalization (Shao et al., 2024). A constant-reward group has zero target. The gradient is subsequently clipped and passed to AdamW in the experiments.

**Answer classes.** Canonicalization partitions valid answers into classes $I_1, \dots, I_N$; invalid answers have known reward zero. Assume that at most one class is correct. Let $H = 0$ denote no correct class and $H = h$ denote that class $h$ is correct. The reward support is

$$r \in \{0, \mathbf{1}_{I_1}, \dots, \mathbf{1}_{I_N}\}.$$

Define the public coefficient matrix $B \in \mathbb{R}^{N \times K}$ by $B_{h,:} = c(\mathbf{1}_{I_h})^\top$ and the corresponding update $v_h = S^\top B_{h,:}^\top$, with $v_0 = 0$.

**Proposition 1** (Complete-update representation)**.** *For every admissible reward state,*

$$f(r) = v_H = \sum_{h=1}^{N} \mathbf{1}\{H = h\} v_h. \tag{2}$$

The matrix $B$ is available from answer membership alone. If $\nu_h = |I_h|/K$, then

$$B_{hi} = \frac{\mathbf{1}\{i \in I_h\} - \nu_h}{K(\sqrt{\nu_h(1 - \nu_h)} + \epsilon)}. \tag{3}$$

All-invalid groups and groups consisting of one repeated valid answer have zero update in every state allowed by their class membership. We call these groups structurally zero; they request no labels. All other groups are active. In particular, a valid class mixed with invalid answers can have a nonzero target.

# 3 SELECTIVE VERIFICATION AND RECONSTRUCTION

## 3.1 RECONSTRUCTING COEFFICIENTS FROM HITS AND EXCLUSIONS

Condition on the generated completions and feedback from earlier batches, denoted by $\mathcal{F}$. A query law samples a class subset $A$ with probability $w_A$. One representative per selected class is verified. The observation $O$ contains $A$ and either its correct class or the fact that all queried classes are incorrect. The latter excludes the queried states without identifying the unqueried ones.

Choose a positive reference distribution $p = (p_0, \dots, p_N)$ before querying. Let $z_o$ be the posterior state vector under this reference after observing $o$. With $T = [0; I_N]$, $P = \mathrm{diag}(p)$ and $C = PT$, define

$$M = \mathbb{E}_p[z_O z_O^\top], \qquad a_*(o)^\top = z_o^\top M^{-1} C, \qquad \widehat{v}_*(o) = \sum_h a_{*,h}(o) v_h. \tag{4}$$

This solves an $(N+1)$-dimensional system. To compute $M$, let $p_{\bar{A}}$ equal $p$ with the queried class coordinates zeroed, retaining coordinate zero. Then

$$M = \sum_A w_A \left[ \sum_{h \in A} p_h e_h e_h^\top + \frac{p_{\bar{A}} p_{\bar{A}}^\top}{\mathbf{1}^\top p_{\bar{A}}} \right]. \tag{5}$$

The first term represents hits; the second represents all-negative observations. Positive inclusion of every class makes $M$ invertible. The reference need not equal the true state probabilities for the estimate to be unbiased.

**A two-class example.** Return to the answers $(42, 42, 17, \text{invalid})$. Name their two valid classes 1 and 2, and take $p = (1/2, 1/4, 1/4)$. Query nothing with probability $1/2$, class 1 with probability $1/4$, and class 2 with probability $1/4$. The expected number of checks is $1/2$, and each class has positive inclusion. Equation (4) gives the reconstruction below, where the returned update is $a_1 v_1 + a_2 v_2$.

| Observation | $a_1$ | $a_2$ |
|---|---|---|
| No class queried | $1/4$ | $1/4$ |
| Class 1 queried, incorrect | $-3/4$ | $1/4$ |
| Class 1 queried, correct | $13/4$ | $1/4$ |
| Class 2 queried, incorrect | $1/4$ | $-3/4$ |
| Class 2 queried, correct | $1/4$ | $13/4$ |

If class 1 is correct, the three possible observations have mean coefficient vector

$$\tfrac{1}{2}(1/4, 1/4) + \tfrac{1}{4}(13/4, 1/4) + \tfrac{1}{4}(1/4, -3/4) = (1, 0).$$

The same calculation gives $(0, 1)$ when class 2 is correct and $(0, 0)$ when neither is correct. Thus the coefficients are correction weights, not probabilities: a negative answer can produce a negative weight, and an empty query can still produce an update. A hit identifies the reward state, but its returned update also compensates for the other possible observations under the query law. In this example the projection coincides with a generalized-difference estimator. Its reference-risk reduction relative to HT comes from correcting a prediction using both positive and negative labels. When neither class is correct, HT is identically zero while projection fluctuates around zero. Reference optimality averages the errors over $p$; norm clipping can break this zero-mean cancellation.

## 3.2 CHOOSING WHICH ANSWERS TO VERIFY

We use an expected budget of $\rho N$ checks per active group, with $\rho = 1/4$. Classes are ordered by first occurrence. The query dictionary contains consecutive cyclic subsets of sizes $\lfloor \rho N \rfloor$ and $\lceil \rho N \rceil$;

duplicate masks are merged. It has at most $2N$ subsets. Uniform rotations and a mixture of the two sizes give an initial law $w^0$ with equal inclusion probability $\rho$.

For a class-update Gram $G_{hk} = v_h^\top v_k$, the reference mean squared error of Equation (4) is

$$R_p(w; G) = \mathrm{tr}(C^\top M(w)^{-1} CG) - \mathrm{tr}(T^\top PTG). \tag{6}$$

The first term is an L-optimal design criterion (Sagnol & Pronzato, 2023). Computing $G$ would require individual scores. We instead optimize coefficient error with the public metric $G_0 = BB^\top$.

The estimator also includes a predictable reliance parameter $t \in [0, 1]$. If $q_h = \Pr(h \in A)$, form the inverse-probability endpoint and its channel mixture:

$$\widehat{v}_{\mathrm{HT}} = \sum_{h \in A} \frac{\mathbf{1}\{H = h\}}{q_h} v_h, \qquad \widehat{v}_t = (1 - t)\widehat{v}_{\mathrm{HT}} + t\widehat{v}_*. \tag{7}$$

The HT reference risk is $R_p^{\mathrm{HT}}(w; G) = \sum_h p_h(q_h^{-1} - 1)G_{hh}$. For a fixed reference and law, the mixture risk is

$$R_{p,t}^{\mathrm{mix}} = (2t - t^2)R_p + (1 - t)^2 R_p^{\mathrm{HT}}.$$

We choose $w$ to minimize this risk under $G_0$, subject to

$$\mathbf{1}^\top w = 1, \qquad w \geq .01w^0, \qquad \sum_A w_A |A| \leq \rho N. \tag{8}$$

The floor gives every class positive inclusion. The solver starts at $w^0$ and uses it if a numerical solution fails the feasibility or objective check.

### 3.3 Learning the reference and reliance online

The initial reference assigns probability $1/2$ to no correct answer and distributes the rest in proportion to class frequency. A six-feature log-linear update uses class frequency, completion length and truncation, with class and none-correct intercepts. For each active group, its feedback is an unbiased state-feature reconstruction from the same selected answers. Appendix B gives the features and step size.

The training implementation uses eight batches of HT warmup, then chooses $t$ from accumulated past coefficient moments with a scale-dependent ridge. Appendix B.4 gives the exact rule. This rule adapts to past prediction error; Proposition 3 characterizes its reference-risk tradeoff. All references, query laws and masks are fixed before either group reveals labels.

### 3.4 A weighted language-model update

Given observed class coefficients $a$, set $u = B^\top a$ and detach this vector. The ascent objective

$$\mathcal{L}(\theta) = \sum_i \mathrm{stopgrad}(u_i) \log \pi_\theta(x_i) \quad \Longrightarrow \quad \nabla_\theta \mathcal{L} = S^\top u \tag{9}$$

implements the reconstructed update. It requires ordinary weighted backpropagation, without extracting any individual $s_i$. Two groups are averaged; microbatches accumulate before one clipped AdamW step. Zero coefficients can be omitted from scoring without changing the denominator. If all coefficients are zero, an explicit zero-gradient step advances the optimizer's moments. The identical convention is used by the full-feedback controls.

### 3.5 Calibrating the reference with a covariance constraint

We refine the reference using nine experts that scale the none-correct odds by $\kappa \in \{1/4, 1, 4\}$ and the class concentration by $\beta \in \{1/2, 1, 2\}$. The identity expert starts with weight $1/2$ and each alternative with $1/16$. Their weights are updated by the likelihood of the observed hit or exclusion event. For two groups,

$$Z_j = \sum_e \omega_{je} \prod_{g=1}^{2} \ell_{jge}, \qquad \omega_{j+1,e} = \frac{\omega_{je} \prod_g \ell_{jge}}{Z_j}. \tag{10}$$

The marginal mixture supplies a second reference $p^{\mathrm{mix}}$. The query law stays fixed. Write $u_{\mathrm{HT}}$, $u_0$ and $u_1$ for the HT, base-reference and calibrated-reference slot endpoints on that law. We interpolate

$$u_\alpha = u_{\mathrm{HT}} + t\{(1-\alpha)u_0 + \alpha u_1 - u_{\mathrm{HT}}\}, \quad \alpha \in [0,1]. \tag{11}$$

Let $c_h = B^\top_{h,:}$ for $h > 0$ and $c_0 = 0$, and define

$$V_h(\alpha) = \mathbb{E}[(u_\alpha - c_h)(u_\alpha - c_h)^\top \mid H = h, \mathcal{F}]. \tag{12}$$

We minimize $\sum_h p_h^{\mathrm{mix}} \operatorname{tr} V_h(\alpha)$ subject to $V_h(\alpha) \preceq (1+\delta)V_h(0)$ for every state. The allowance $\delta \geq 0$ is fixed at .1 in training. These are small public matrices. The feasible values form an interval containing zero, so one scalar search suffices. This calibration uses no additional verification or model scores. Its expert formulas and numerical tolerances are specified in Appendices C and D.

## 4 Statistical Guarantees

The guarantees condition on the current public information $\mathcal{F}$ and treat the true reward state as fixed. Expectations are over the randomized verification. Appendix A gives complete proofs.

**Theorem 2** (Unbiased reconstruction and fixed-law optimality)**.** *For a positive reference and positive inclusion of every answer class, Equation (4) satisfies $\mathbb{E}[\widehat{v}_* \mid H = h, \mathcal{F}] = v_h$ for each state. Every estimator $g(O)$ on the same observation channel with the same statewise means obeys*

$$\mathbb{E}_p[(g - v_H)(g - v_H)^\top] - \mathbb{E}_p[(\widehat{v}_* - v_H)(\widehat{v}_* - v_H)^\top] = \mathbb{E}_p[(g - \widehat{v}_*)(g - \widehat{v}_*)^\top] \succeq 0. \tag{13}$$

The proof expresses unbiasedness as a linear constraint on observation-dependent outputs. A weighted least-squares projection satisfies this constraint, and its orthogonality removes the cross term in Equation (13). The result also holds when the class updates are linearly dependent. Optimality is under the specified reference and fixed query law; unbiasedness holds separately in every state.

**Proposition 3** (Predictable mixtures)**.** *Selecting $t$ from past information preserves statewise unbiasedness. On a fixed law and under the construction reference,*

$$R_p(\widehat{v}_t) = R_p(\widehat{v}_*) + (1-t)^2\{R_p(\widehat{v}_{\mathrm{HT}}) - R_p(\widehat{v}_*)\}. \tag{14}$$

This identity explains the query-design objective. The online rule chooses reliance using past coefficient moments, while calibrated interpolation controls its current-state covariance.

**Theorem 4** (Statewise covariance control in exact arithmetic)**.** *For a fixed allowance $\delta \geq 0$, current history, query law and $t$, any feasible predictable $\alpha$ in Equation (11) is statewise unbiased and satisfies, for every fixed score map $S$,*

$$\operatorname{Cov}(S^\top u_\alpha \mid H = h, \mathcal{F}) \preceq (1+\delta)\operatorname{Cov}(S^\top u_{\alpha=0} \mid H = h, \mathcal{F}). \tag{15}$$

*The same bound holds for the average of independently queried groups, conditional on their reward states.*

The covariance is a matrix-convex quadratic in $\alpha$. Its feasible set contains zero, and congruence by $S$ transfers the coefficient bound to gradient space. This guarantee requires no accuracy assumption on the reference and no approximation assumption on $BB^\top$. It applies to the raw update before gradient clipping and AdamW; the experiments measure the resulting policy learning.

## 5 Learning with a Common Training Configuration

We evaluate Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct (Qwen, 2024) and SmolLM2-1.7B-Instruct (Ben Allal et al., 2025) on GSM8K (Cobbe et al., 2021). The study contains 105 runs: 90 runs of six $K = 8$ methods and 15 additional small-group controls. Each method uses the same 256 training questions, fresh rank-16 LoRA adapters (Hu et al., 2021), five seeds, 400 optimizer updates and greedy evaluation on all 1,319 test questions. Each main-comparison batch has two questions and eight completions per question, giving 6,400 generated completions per run. AdamW (Loshchilov & Hutter, 2017) uses learning rate $10^{-5}$ for Qwen 7B and $3 \cdot 10^{-5}$ otherwise,

with gradient norm clipping at one. Prompts, decoding, adapter modules, seed rules and training indices are specified in Appendix E.

Initial-policy accuracy is 70.13%, 80.97%, 32.52% for Qwen 3B, Qwen 7B and Smol, respectively. The initial and trained policies use the same 1,319-question greedy evaluation. All training seeds of a model begin with this common adapter initialization. At update 400, the trained Structured policies are 10.10, 6.41 and 19.61 percentage points above initialization. The comparisons below assess them against other trained policies.

**Comparisons.** *Structured* is the method in Sections 3–3.5, including online calibration and its covariance constraint. On active groups, *Full feedback* determines the complete reward state by checking answer classes in descending frequency order and stopping at the first correct class. *Uniform HT* independently verifies each class with probability $1/4$ and applies inverse-probability weighting. *Independent* uses independently allocated checks and a generalized-difference reconstruction, with the same online predictor and calibration components (Appendix B). *No expert calibration* removes the expert mixture and covariance-constrained interpolation while keeping the base query design, feature prior and reliance rule. Every method shares answer canonicalization, duplicate-label reuse and structurally zero targets. We also vary the amount of generation in simple full-feedback controls, keeping the rest of the training recipe fixed.

**Statistical comparison.** We compare final accuracy using 50,000 paired crossed-bootstrap draws over five seeds and 1,319 test questions. A single Bonferroni family contains 21 contrasts: Structured against six trained controls and the common initialization, for each model. Five seeds give an approximate sampling distribution; intervals containing zero do not establish equivalence. Appendix G gives every contrast. Appendix I.3 gives paired seed-level $t_4$ intervals conditional on the fixed test set as a sensitivity check.

### 5.1 Accuracy and verification at matched generation

Table 1: Same training budget: 400 updates and 6,400 completions. Accuracy is the mean and sample standard deviation over five seeds, evaluated on all 1,319 GSM8K test questions. Checks are mean external verification calls, computed exactly from the recorded responses under a shared hit-and-stop rule and per-run answer cache. The same service rules apply to every method.

| | Qwen 3B | | Qwen 7B | | Smol 1.7B | |
|---|---|---|---|---|---|---|
| Method | Accuracy (%) | Checks | Accuracy (%) | Checks | Accuracy (%) | Checks |
| Structured (ours) | $80.23 \pm 1.65$ | 136.6 | $87.38 \pm 0.80$ | 64.4 | $52.13 \pm 0.91$ | 482.0 |
| Full feedback | $77.24 \pm 5.04$ | 176.6 | $87.64 \pm 1.08$ | 109.2 | $52.05 \pm 0.56$ | 737.8 |
| Uniform HT | $74.77 \pm 3.99$ | 140.2 | $87.13 \pm 0.30$ | 63.2 | $49.04 \pm 2.63$ | 499.2 |
| Independent | $78.47 \pm 2.82$ | 122.2 | $87.67 \pm 0.77$ | 59.2 | $51.78 \pm 0.99$ | 455.0 |
| No expert calibration | $80.12 \pm 1.40$ | 123.2 | $87.16 \pm 1.39$ | 64.0 | $51.72 \pm 0.56$ | 483.0 |

Table 1 fixes both optimizer updates and generated completions. All methods use the same verification service: check requested classes by decreasing frequency, stop after a correct answer, and cache positive and negative answers for each training question. Each run starts with an empty cache. A known positive answer determines the labels of all subsequent answers to that question; a previously rejected canonical answer also requires no new call.

The chosen $\rho = 1/4$ controls requested labels before stopping and caching. The external-call reductions below additionally depend on answer repetition, hits and the resulting policy trajectories. Uniform HT has the same conditional update distribution as independently sampling each whole group with probability $\rho$ and weighting its complete update by $1/\rho$; their stopping and cache costs can differ (Appendix I.1).

Label-service replay preserves every returned label and hence the recorded policy trajectories. Structured uses 136.6, 64.4 and 482.0 mean new checks on Qwen 3B, Qwen 7B and Smol, compared with 176.6, 109.2 and 737.8 for full feedback: mean reductions of 22.7%, 41.0% and 34.7%. The corresponding absolute call savings are $40.0\,[-17.2, 97.2]$, $44.8\,[19.6, 70.0]$ and $255.8\,[184.2, 327.4]$, with intervals adjusted over the 18 cost comparisons (Table 20); the Qwen 3B interval includes zero.

The family-adjusted accuracy intervals against all trained controls include zero (Table 9). Table 11 separates label requests, stopping, and caching.

### 5.2 Simple reductions in generation

Table 2: Full-feedback controls with fewer generated completions. Every row uses 400 updates and five seeds. All methods use the same hit-and-stop and answer-cache service for query accounting. Accuracy is mean ± sample standard deviation.

| Method | Completions per run | Qwen 3B Accuracy (%) | Qwen 3B Checks | Qwen 7B Accuracy (%) | Qwen 7B Checks | Smol 1.7B Accuracy (%) | Smol 1.7B Checks |
|---|---|---|---|---|---|---|---|
| Structured (ours) | 6,400 | $80.23 \pm 1.65$ | 136.6 | $87.38 \pm 0.80$ | 64.4 | $52.13 \pm 0.91$ | 482.0 |
| Half-batch full | 3,200 | $80.03 \pm 0.83$ | 134.6 | $87.78 \pm 0.40$ | 72.8 | $50.48 \pm 3.00$ | 510.2 |
| Small-group full | 1,600 | $69.67 \pm 16.35$ | 158.0 | $87.38 \pm 0.70$ | 68.2 | $48.57 \pm 2.12$ | 534.6 |

The half-batch control uses one question and eight completions per update, halving generation while keeping 400 optimizer steps. The small-group control uses two questions and two completions per question, reducing generation to one quarter. Its normalized target uses the same formula evaluated within the smaller group. Both the initialization and model-specific learning rate remain fixed. Table 2 reports accuracy and new verification calls; all paired accuracy intervals appear in Table 9.

For cost per generated completion $c_{\text{gen}}$ and per new verification $c_{\text{ver}}$, consider

$$C = c_{\text{gen}} n_{\text{gen}} + c_{\text{ver}} n_{\text{ver}}. \tag{16}$$

Relative to half-batch full, Structured uses 3,200 more completions. Its mean call savings are $-2.0$, $8.4$ and $28.2$, so the Qwen 7B and Smol break-even ratios $c_{\text{ver}}/c_{\text{gen}}$ are approximately 381 and 113. Qwen 3B has no positive-cost crossover. These are equal-price completion scenarios; token lengths, optimizer work and accuracy also matter. Independent uses fewer checks than Structured on all three models. Appendix J includes all four methods, recorded-time scenarios and call uncertainty. With caching, none of the half-batch call-saving intervals excludes zero, so these plug-in thresholds are imprecise.

### 5.3 Learning trajectories and resource accounting

Figure 2 compares Structured, full feedback and half-batch full along both resource axes. The common initial policy is evaluated once per model. Training checkpoints were evaluated when requested class labels first crossed 64, 128 or 256 and at update 400. Each marker uses its actual update count and the cumulative new external calls under the common verification service.

Evaluation uses one deterministic completion and verification per test question, without training-cache access. It is accounted for separately. Table 12 reports generated tokens, backward calls and training time, which can differ at a fixed completion count.

Structured takes 90.2, 64.2 and 83.5 recorded training minutes, versus 82.5, 63.4 and 65.6 for full feedback. These totals include generation, query selection, verification, gradients and controller updates on NVIDIA L20D GPUs, excluding loading, evaluation, serialization and queueing. They measure the original batched service; stopping and cache counts use exact label-service replay. Verification-count reductions therefore coexist with higher recorded training time.

Clipping is active on 99.28–100% of steps with nonzero slot coefficients across methods. For Structured, clipping occurs on 56.50%, 34.50% and 95.50% of all optimizer steps. Appendix I.2 reports both denominators and pre-clipping norms. These results separate the raw-estimator guarantees from learning under norm-rescaled updates and AdamW.

### 5.4 Design and calibration components

The uniform estimator uses the same expected fraction of class checks and gives accuracies of 74.77%, 87.13% and 49.04%. The structured estimates are higher by 5.46, 0.26 and 3.09 percentage points in the observed means. Independent query allocation and the calibration ablation provide

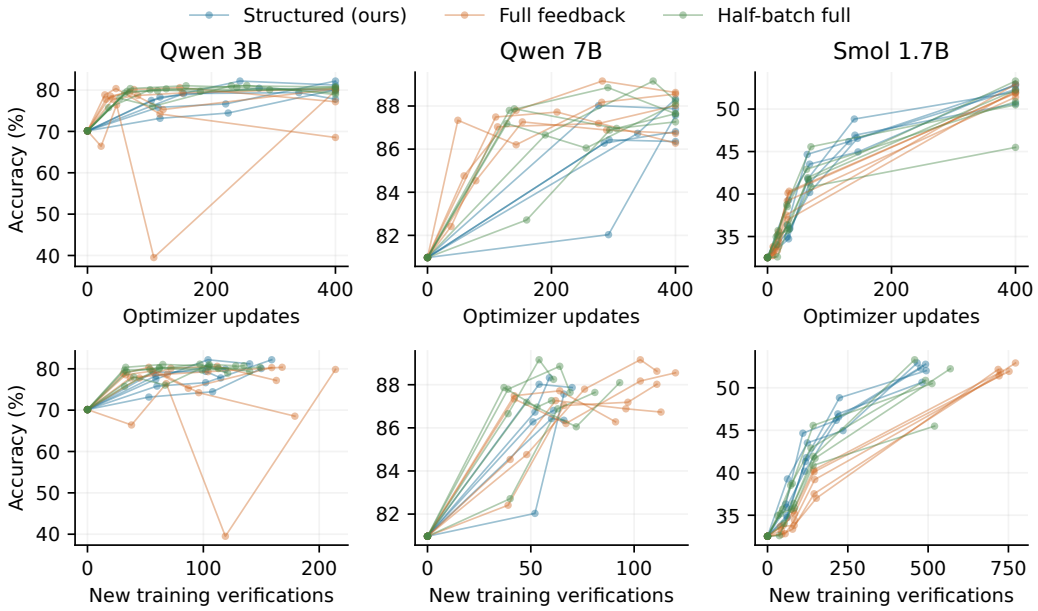


Figure 2: Measured learning trajectories with the same stopping and answer-cache rules. Each line is one training seed; markers are evaluated checkpoints, including the shared initialization at update zero. The upper row uses optimizer updates, the lower row new external verification calls. Structured and full feedback generate 16 completions per update; half-batch full generates eight. Lines connect measured checkpoints.

comparisons for the two additional choices, with every seed and the paired intervals reported in Appendix G.

To isolate reconstruction from query design, we enumerate all possible observations for the 5,701 active groups in the Structured runs, holding the generated completions, query law and prequery history fixed. Channel projection and generalized-difference reconstruction have nearly identical true-state coefficient errors: their pooled error ratios are 0.999221, 1.000001 and 0.999190. Both substantially reduce this error relative to HT (Appendix H).

Table 3: Reconstruction at fixed generated groups and query laws. Entries are ratios of summed true-state coefficient errors across all five seeds; smaller is better. Projection/GD compares pure endpoints. The calibration columns compare the guarded and unconstrained interpolations to the base mixed estimator.

| Model | Projection/GD | Guarded/base | Unconstrained/base |
|---|---|---|---|
| Qwen 3B | 0.999221 | 0.95292 | 0.74965 |
| Qwen 7B | 1.000001 | 0.95974 | 0.81943 |
| Smol 1.7B | 0.999190 | 0.97664 | 0.81897 |

Guarded expert calibration reduces true-state coefficient error by 4.71%, 4.03% and 2.34% relative to the base mixed estimator. Removing the covariance constraint yields larger average reductions but violates its statewise cap on 95.3–97.0% of active groups. The guarded outputs satisfy the cap within the specified numerical tolerance. Unconstrained calibration also increases error at the actual reward state in 204, 106 and 942 active groups; Table 22 quantifies what the constraint limits. Query design and endpoint construction take 3.2–12.7 milliseconds per batch (Table 5).

The unconstrained comparator minimizes the same reference-risk quadratic over $[0, 1]$, keeping the query law and recorded reliance fixed. Its lower pooled error shows the cost of statewise protection. Appendix I.4 varies the allowance over $\{0, .05, .1, .5\}$ on these same histories: larger allowances permit larger average error reductions, while zero allowance returns the base estimator in this study.

On the same histories, cyclic GD has 12.0%, 14.7% and 8.5% lower pooled error than allocated independent GD (Table 21). This isolates a query-law effect; the separate learning runs produce different answers and cache histories.

## 6 Related work

**Partial feedback and active supervision.** Horvitz–Thompson and generalized-difference estimators correct selective observations via inclusion probabilities (Horvitz & Thompson, 1952; Särndal et al., 1992); partial-information estimators also use exclusions (Cohen & Kaplan, 2011). Prediction-powered and active inference reduce labeling through predictions (Angelopoulos et al., 2023a;b; Li et al., 2025). For a generated batch's complete policy update, answer classes define a finite observation channel; public coefficients enable tractable query design (Sagnol & Pronzato, 2023).

**Reward normalization and efficient policy learning.** GRPO normalizes group rewards without learning a value function (Shao et al., 2024). Dr. GRPO removes reward-standard-deviation and response-length normalization to address optimization biases (Liu et al., 2025). Mean-centering yields a linear target admitting HT or generalized-difference estimation (Appendix F); our learning comparisons retain group normalization.

DAPO dynamically replaces zero-variance groups (Yu et al., 2025), using feedback to determine contributing prompts. We estimate a prescribed batch's update, skipping structurally zero groups before checking. TTRL uses model-majority rewards (Zuo et al., 2025) instead of external verification. Active reward acquisition and label-efficient RLHF allocate limited feedback through preference or reward models with different observation spaces (Lee et al., 2021; Christiano et al., 2017).

## 7 Limitations and analysis scope

The emphasis on update 400, the 21-comparison accuracy family and stopping/cache accounting was chosen after the initial analysis. Reanalyses use completed 400-update trajectories; service replay preserves their labels and policies. The GSM8K test set was used during method development.

Answer classes must preserve correctness and be mutually exclusive. GSM8K verification is inexpensive; measured training times show no net speedup. Persistent caching suits repeated questions; uncached stopping reduces calls by 42.8%, 59.5% and 53.7% against full feedback. Neither replay measures costly-verifier latency. Raw-estimator guarantees apply before clipping and AdamW; update-direction effects of frequent clipping and a trained Dr. GRPO comparison remain unmeasured.

## 8 Conclusion

Mutually exclusive answer classes represent the complete group-normalized update by a small set of known coefficient vectors. Structured verification selects checks, reconstructs updates and implements them through a weighted loss. Predictable calibration preserves statewise unbiasedness and constrains the impact of reference changes. A shared training configuration and explicit feedback, generation and optimizer costs quantify the learning tradeoff.

## AI Use Statement

AI tools were used solely for language polishing of the manuscript. The authors are responsible for checking the submitted claims and results.

## Reproducibility Statement

The appendix contains proofs, the full algorithm, numerical settings, answer parsing, prompts, training indices, per-seed results and evaluation procedures; source and machine-readable results accompany the submission.

## REFERENCES


Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 382(6671):669–674, 2023a. doi: 10.1126/science.adi6000. URL https://doi.org/10.1126/science.adi6000.

Anastasios N. Angelopoulos, John C. Duchi, and Tijana Zrnic. PPI++: Efficient prediction-powered inference. arXiv:2311.01453, 2023b. URL https://arxiv.org/abs/2311.01453.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, et al. SmolLM2: When smol goes big – data-centric training of a small language model, 2025. URL https://arxiv.org/abs/2502.02737.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. doi: 10.1017/CBO9780511546921. URL https://www.cambridge.org/core/books/prediction-learning-and-games/A05C9F6ABC752FAB8954C885D0065C8F.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1706.03741.

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:2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Edith Cohen and Haim Kaplan. Get the most out of your sample: Optimal unbiased estimators using partial information. In *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 13–24, 2011. doi: 10.1145/1989284.1989288. URL https://arxiv.org/abs/1109.1325.

D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952. doi: 10.1080/01621459.1952.10483446.

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. arXiv:2106.09685, 2021. URL https://arxiv.org/abs/2106.09685.

Kimin Lee, Laura Smith, and Pieter Abbeel. PEBBLE: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *International Conference on Machine Learning*, 2021. URL https://arxiv.org/abs/2106.05091.

Puheng Li, Tijana Zrnic, and Emmanuel Candès. Robust sampling for active statistical inference. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2511.08991.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025. URL https://arxiv.org/abs/2503.20783.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. arXiv:1711.05101, 2017. URL https://arxiv.org/abs/1711.05101.

Qwen. Qwen2.5 technical report. arXiv:2412.15115, 2024. URL https://arxiv.org/abs/2412.15115.

Guillaume Sagnol and Luc Pronzato. Fast screening rules for optimal design via quadratic lasso reformulation. *Journal of Machine Learning Research*, 24(307):1–32, 2023. URL https://www.jmlr.org/papers/volume24/22-1305/22-1305.pdf.

Carl-Erik Särndal, Bengt Swensson, and Jan Wretman. *Model Assisted Survey Sampling*. Springer, 1992. URL https://link.springer.com/book/9780387406206.

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

Qiying Yu, Zheng Zhang, Ruofei Zhu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. URL https://arxiv.org/abs/2503.14476.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, Biqing Qi, Youbang Sun, Zhiyuan Ma, Lifan Yuan, Ning Ding, and Bowen Zhou. TTRL: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*, 2025. URL https://arxiv.org/abs/2504.16084.

# A PROOFS

This appendix proves the finite-state estimator and the claims used in the main text. Weighted least-squares projection and L-optimal design are established principles. The derivations make their application to the present reward support explicit, including the distinction between coefficient and actual-vector unbiasedness.

Table 4: Principal notation. All quantities refer to the current generated group unless indexed by a training batch.

| Symbol | Meaning |
|---|---|
| $K, N$ | Number of completions and distinct valid answer classes |
| $H \in \{0, \dots, N\}$ | Correct class, with zero denoting no correct class |
| $B \in \mathbb{R}^{N \times K}$ | Public normalized coefficients for each class state |
| $S \in \mathbb{R}^{K \times d}$ | Completion score vectors stacked as rows |
| $p, p^{\mathrm{mix}}$ | Base and calibrated reference state probabilities |
| $w_A, q_h$ | Probability of querying subset $A$; inclusion of class $h$ |
| $M$ | Reference posterior second-moment matrix |
| $a(O), u(O)$ | Reconstructed class coefficients and slot weights |
| $t, \alpha$ | HT–reconstruction reliance; calibrated interpolation |
| $V_h(\alpha)$ | Slot-error covariance conditional on reward state $h$ |
| $\Lambda = \mathrm{diag}(Qp)$ | Diagonal observation-probability matrix |
| $\xi_O, \psi_j$ | Update difference at observation $O$; batch slot-weight difference |
| $\Delta_X, \Omega_h$ | Calibration direction; quadratic covariance coefficient |

## A.1 SUPPORT REDUCTION AND PUBLIC COEFFICIENTS

Let $n_c = |I_c|$ and $\nu_c = n_c/K$. For a class mask $r = \mathbf{1}_{I_c}$,

$$\bar{r} = \nu_c, \qquad \sigma(r) = \sqrt{\nu_c(1-\nu_c)}, \qquad B_{ci} = \frac{\mathbf{1}\{i \in I_c\} - \nu_c}{K(\sqrt{\nu_c(1-\nu_c)} + \epsilon)}. \tag{17}$$

Every term is known from generated answers alone. When $n_c = K$, the numerator is zero and $B_{c,:} = 0$. When there is no valid class, every reward is known zero. These cases admit an exact zero update without querying. If a single valid class coexists with invalid answers, however, $0 < n_c < K$ and its target need not vanish and the structural zero shortcut does not apply.

*Proof of Proposition 1.* The exhaustive states are $r = 0$, with target zero, and $r = \mathbf{1}_{I_c}$, with target $S^\top B_{c,:}^\top = v_c$, proving Equation (2). For a general target $F$ with $F(0) \neq 0$, the corresponding support identity is $F(r) = F(0) + \sum_c \mathbf{1}\{H = c\}\{F(\mathbf{1}_{I_c}) - F(0)\}$. No assumption that $F$ is linear away from this support is needed. □

Canonicalization must preserve verifier outcomes and at most one valid class can be correct. Stochastic grading or multiple passing programs requires a different state space: a hit then need not identify unqueried rewards.

## A.2 A GENERAL FINITE-CHANNEL PROJECTION

Retain the $m$ observations possible in at least one state. Let $Q_{oh} = \Pr(O = o \mid H = h, \mathcal{F})$ have full column rank and unit column sums. For $p_h > 0$, set $P = \mathrm{diag}(p)$, $\mu = Qp$, and $\Lambda = \mathrm{diag}(\mu) \succ 0$. The rows of $Y \in \mathbb{R}^{(N+1)\times d}$ are the desired targets $v_h^\top$.

An estimator that is measurable with respect to $O$ is a matrix $A \in \mathbb{R}^{m \times d}$ of output rows. Its statewise unbiasedness constraints are exactly

$$Q^\top A = Y. \tag{18}$$

This constraint defines the class of all observation-measurable unbiased outputs. Set $J = Q^\top \Lambda^{-1} Q \succ 0$ and

$$A_* = \Lambda^{-1} Q J^{-1} Y. \tag{19}$$

Then $Q^\top A_* = Y$. For any feasible $A = A_* + \Delta$, the perturbation satisfies $Q^\top \Delta = 0$, and therefore

$$A_*^\top \Lambda \Delta = Y^\top J^{-1} Q^\top \Delta = 0. \tag{20}$$

The reference-integrated error covariance can be expanded without assuming that $p$ is a true conditional prior:

$$\mathcal{R}_p(A) = \sum_{h,o} p_h Q_{oh} (A_o - Y_h)^\top (A_o - Y_h) \tag{21}$$

$$= A^\top \Lambda A - A^\top Q P Y - Y^\top P Q^\top A + Y^\top P Y \tag{22}$$

$$= A^\top \Lambda A - Y^\top P Y. \tag{23}$$

Substituting $A_* + \Delta$ and using Equation (20) gives

$$\mathcal{R}_p(A) - \mathcal{R}_p(A_*) = \Delta^\top \Lambda \Delta \succeq 0. \tag{24}$$

Since $\Lambda \succ 0$, zero excess trace requires $\Delta = 0$: the statewise-unbiased optimum is unique on possible observations.

Conditioning away randomization uninformative about $H$ beyond $(O, \mathcal{F})$ preserves statewise expectation and removes covariance, so it cannot improve the optimum. Additional informative signals require a different $Q$ and are outside this comparison.

### A.3 Posterior coordinates and singular update Grams

The posterior vector under $p$ is $z_o = P Q^\top_{o,:} / \mu_o$. Its uncentered second moment obeys

$$M = \sum_o \mu_o z_o z_o^\top = P Q^\top \Lambda^{-1} Q P = P J P. \tag{25}$$

For coefficient targets $T = [0; I_N]$, stacking the rows in Equation (4) gives

$$A_*^{\text{coef}} = \Lambda^{-1} Q P M^{-1} P T = \Lambda^{-1} Q J^{-1} T. \tag{26}$$

The construction depends only on $Q$ and $p$. If $V \in \mathbb{R}^{N \times d}$ stacks $v_c^\top$, the actual target rows satisfy $Y = TV$, and the optimal output is $A_* = A_*^{\text{coef}} V$.

The direct $Y$ proof covers all *actual-vector*-unbiased competitors, including dependent rows of $V$. Assuming coefficient unbiasedness instead would miss competitors when $V$ has a nontrivial nullspace.

Equation (24) also gives nonnegative excess trace against every $G = VV^\top \succeq 0$. Thus the coefficient rule is optimal for every positive semidefinite metric. A singular $G$ can make different coefficient representations give equal output risk; strict improvement requires a nonzero actual output difference on a positive-probability observation.

### A.4 Structured information and identification

For a fixed selected subset $A$, each $c \in A$ yields a positive observation with probability $p_c$ and posterior $e_c$. The all-negative event has probability $r_A = \mathbf{1}^\top p_{\bar{A}} > 0$ and posterior $p_{\bar{A}} / r_A$. Its second-moment contribution is $p_{\bar{A}} p_{\bar{A}}^\top / r_A$. Adding these contributions and averaging over the sampled subset gives Equation (5).

To establish identification, suppose $x^\top M x = 0$. Every summand in

$$x^\top M x = \sum_A w_A \left\{ \sum_{c \in A} p_c x_c^2 + \frac{(p_{\bar{A}}^\top x)^2}{\mathbf{1}^\top p_{\bar{A}}} \right\} \tag{27}$$

is nonnegative. Positive inclusion of each class forces $x_c = 0$ for $c = 1, \ldots, N$. An all-negative term then forces $p_0 x_0 = 0$, hence $x = 0$. Thus $M \succ 0$, and Equation (25) gives full column rank of $Q$. This condition suffices to identify all state targets. It need not be necessary for a degenerate vector target, including a structurally zero one.

Reference accuracy is not needed for unbiasedness, but poor probabilities can make rare observations receive large coefficients and increase realized-state variance.

### A.5 Reference risk and convex design

Using Equation (26),

$$(A_*^{\mathrm{coef}})^\top \Lambda A_*^{\mathrm{coef}} = T^\top J^{-1} T = C^\top M^{-1} C. \tag{28}$$

Taking the trace of its coefficient-error covariance times $G$ proves Equation (6). Since $M(w) = \sum_A w_A M_A$, its first term is

$$\operatorname{tr}(M(w)^{-1} C G C^\top). \tag{29}$$

This is the L-optimal criterion with a fixed positive semidefinite weight. Convexity follows, for example, by writing $CGC^\top = LL^\top$ and applying convexity of $\ell^\top M^{-1}\ell$ to each column $\ell$ of $L$. The derivative is

$$\frac{\partial R_p}{\partial w_A} = -\operatorname{tr}(C^\top M^{-1} M_A M^{-1} C G) = -\operatorname{tr}(U^\top M_A U G), \qquad U = M^{-1}C. \tag{30}$$

This convex design criterion uses the specified finite dictionary, fixed reference and verification-cost constraint. The numerical solver is described in Appendix E.

Direct evaluation avoids subtracting large second moments. For the realized state $h$, let $o(A, h)$ be the observable hit/negative branch. Then the exact conditional error is

$$R_h = \sum_A w_A \big(a_*(o(A,h)) - T_h^\top\big)^\top G \big(a_*(o(A,h)) - T_h^\top\big). \tag{31}$$

We use this nonnegative expression to evaluate conditional risk without subtracting large second moments.

### A.6 Mixture risk

Write $\xi_O = \widehat{v}_*(O) - \widehat{v}_{\mathrm{HT}}(O)$ and $E_O = \widehat{v}_{\mathrm{HT}}(O) - v_H$. For fixed $t$,

$$\mathbb{E}\,\|E_O + t\xi_O\|^2 = \zeta_0 + 2\zeta_1 t + \zeta_2 t^2, \quad (\zeta_2, \zeta_1, \zeta_0) = \big(\mathbb{E}\,\|\xi_O\|^2, \mathbb{E}\langle E_O, \xi_O\rangle, \mathbb{E}\,\|E_O\|^2\big). \tag{32}$$

For $\zeta_2 > 0$, minimizing this quadratic over $[0,1]$ gives $t^* = \mathrm{clip}(-\zeta_1/\zeta_2, 0, 1)$. If $\zeta_2 = 0$, then $\xi_O = 0$ in the evaluation metric almost surely, so $\zeta_1 = 0$ and all values tie; our convention takes zero. This is the quadratic minimized for a fixed reference and observation law.

Both endpoints are statewise unbiased and $t$ is fixed conditional on $\mathcal{F}$, proving the first claim of Proposition 3. Under the construction reference, the mixture is $\widehat{v}_* - (1-t)\xi_O$. Its projection cross term vanishes, leaving excess risk $(1-t)^2\mathbb{E}_p\,\|\xi_O\|^2$ and proving Equation (14). A data-dependent $t$ selected after the current observation does not follow this proof.

### A.7 Aggregate gradients and batching

For fixed sampled token sequences, detachment makes $u = B^\top a_t$ constant in the differentiated objective. Differentiating the weighted sum gives Equation (9). For a batch of $J$ questions the target is $J^{-1}\sum_j \widehat{v}_j$; splitting its token loss into microbatches preserves the same gradient if all partial gradients accumulate before the optimizer step. Removing zero coefficients does not change the original factor $1/J$.

For two groups with independent query randomizations and conditional errors $e_1, e_2$ of mean zero,

$$\mathbb{E}\,\|(e_1 + e_2)/2\|^2 = \tfrac{1}{4}\{\mathbb{E}\,\|e_1\|^2 + \mathbb{E}\,\|e_2\|^2\}, \tag{33}$$

because the cross expectation vanishes. Shared calibration does not invalidate this identity after conditioning on it. Shared query randomness or a biased estimator would require the cross term and cannot reuse this formula without checking it.

Padding, microbatching, reduction order, and precision can change floating-point results despite real-arithmetic equality. Clipping and optimizer transformations do not generally commute with expectation, motivating separate backward and learning measurements.

**Algorithm 1: a guarded, warmup-scaled two-group policy batch**

1. Generate both groups, canonicalize classes, construct $B$, and identify structurally zero targets.
2. Snapshot the past-only feature prior, expert weights and $t$. Construct each base query law and calibrated endpoint, solve the covariance-constrained interpolation, then sample both independent masks. Freeze these quantities before either group's labels.
3. Verify the selected class representatives. From each pre-query channel and the observed hit or exclusions, obtain the HT and guarded correction coefficients. Update the base prior and expert weights for future use only; keep current coefficients unchanged.
4. Detach the mixed slot weights, score nonzero-weight records in sampling order, and accumulate microbatches of four with denominator two. Clip at one and advance AdamW, including an explicit zero-gradient step when all coefficients vanish.
5. Only after that policy step, incorporate the observed endpoint coefficient moments into Equation (35). Checkpoints follow completed policy and controller transitions.

Figure 3: The algorithm uses ordinary training labels and public coefficients, including during warmup. One-group full feedback samples one task per step and uses denominator one; all controls use the same evaluation procedure.

# B Complete algorithm and baselines

## B.1 Two-group training algorithm

For $N > 0$, initialize $p^0 = (1/2, \nu_1/(2\sum_c \nu_c), \ldots, \nu_N/(2\sum_c \nu_c))$ and $\varphi = 0$. Here $\ell_i$ is completion token length and $\tau_i$ marks truncation at the token limit; all overbars in $x_c$ average only over slots $i \in I_c$. The none-state feature is $(1, 0, 0, 0, 0, 0)$; a class feature is

$$x_c = (0, 1, \mathrm{clip}_{[-2,0]}(\log \nu_c/4), \nu_c, \mathrm{clip}_{[0,2]}(\overline{\log(1+\ell)}/6), \mathrm{clip}_{[0,1]}(\bar{\tau})).$$

Let $X_g \in \mathbb{R}^{(N_g+1)\times 6}$ stack the state-feature rows for group $g$. Use $p_h \propto p_h^0 \exp(x_h^\top \varphi)$, which retains full support. Before sampling, the same channel inverse constructs every possible state-feature estimate $\widehat{x}(O)$. Each group's predictor feedback is $d_g^{\mathrm{pred}} = p_g^\top X_g - \widehat{x}_g(O)$; average both groups, assigning zero to an unqueried structurally zero target. For group $g$, put $L_g = \max_o \|p_g^\top X_g - \widehat{x}_g(o)\|_2$, maximizing over its possible observations, and set $L_g = 0$ for a structurally zero target. With $L = (L_1 + L_2)/2$, use $\eta = \min(.05, .05/\max(1, L))$ and project $\varphi - \eta(d_1^{\mathrm{pred}} + d_2^{\mathrm{pred}})/2$ onto the radius-three Euclidean ball. The pre-query step cap is independent of the realized hit. Batches whose targets are both structurally zero leave this prior unchanged. For an active group, this is a statewise-unbiased estimate of $p^\top X - x_H$. Structurally zero groups receive weight zero, so the batch prediction objective averages active-group residuals with the fixed denominator two.

For independent queries, the endpoint is Equation (34) with $m = p_{1:N}$. Mixing with HT gives effective baseline $tp_{1:N}$. Allocate $q_c$ using $d_c = [p_c - (2t - t^2)p_c^2](BB^\top)_{cc}$ under the same expected quarter-class budget and positive inclusion bounds. This allocation changes with the pre-query $t$. Cyclic querying uses the current mixture objective and the same feasibility floor as in the main text. Both arms use only selected-class feedback.

Full G2 feedback and uniform class HT use two groups; full G1 feedback uses one. Full feedback queries all valid classes unless the target is structurally zero. Uniform HT samples each valid class with probability .25. Invalid-answer zeros, duplicate reuse, and free structurally zero targets are shared. For full-feedback hit-and-stop, order classes by decreasing frequency with first occurrence breaking ties, and stop after the first positive result. This determines the same complete update. For all methods, a per-run cache serves repeated answers without a new external call; Appendix G defines the service and verifies its label equivalence.

### B.2 Independent and uniform verification

Let $y_h = \mathbf{1}\{H = h\}$. For public baselines $m_h$ and inclusion probabilities $q_h$, the generalized-difference estimator is

$$\widehat{v}_m = \sum_h m_h v_h + \sum_h \frac{\mathbf{1}\{h \in A\}}{q_h}(y_h - m_h)v_h. \tag{34}$$

For independent Bernoulli queries its conditional risk is $\sum_h (q_h^{-1} - 1)(y_h - m_h)^2 G_{hh}$. Before expert calibration, the independent method uses $m_h = tp_h$. Covariance-constrained interpolation changes this to $m_h = t\{(1-\alpha)p_h + \alpha p_h^{\mathrm{mix}}\}$. It uses the same online predictor, warmup, ridge and calibration procedure, and its own training history. Query allocation is computed first using the base reference $p$ and the effective baseline $tp$, with $10^{-6} \le q_h \le 1$ and $\sum_h q_h = \rho N$. Expert calibration and its interpolation then keep these $q_h$ fixed; they do not redesign the queries. Uniform HT uses $m_h = 0$ and $q_h = \rho$. Full feedback and these controls all share duplicate-answer reuse and structurally zero targets.

### B.3 Design implementation

Let $\lambda = \rho N - \lfloor \rho N \rfloor$. The initial law assigns each cyclic subset of size $\lfloor \rho N \rfloor$ weight $(1-\lambda)/N$ and each subset of the next size weight $\lambda/N$. Merge identical subsets before solving. Each information-matrix construction and solve costs $O(|\mathcal{A}|N^2 + N^3)$ dense arithmetic. The final numerical query solver and independent scalar allocation are specified in Appendix E.

### B.4 Predictable reliance rule

The first eight batches use $t = 0$. For batch $j$, let $H_j$ and $\psi_j$ concatenate the two groups' HT slot weights and reconstruction-minus-HT slot weights, each divided by two. The reconstruction includes the calibration defined below. Accumulate $a_j = \|\psi_j\|^2$, $d_j = H_j^\top \psi_j$ and $h_j^2 = \|H_j\|^2$. At the first $n \ge 8$ with positive accumulated energy, freeze $s^2 = n^{-1}\sum_{j \le n}(a_j + h_j^2)$; subsequent batches use

$$t_{n+1} = \mathrm{clip}_{[0,1]} \frac{-\sum_{j \le n} d_j}{.125 s^2 + \sum_{j \le n} a_j}. \tag{35}$$

Until this scale exists, $t = 0$. Both groups' references, query laws and masks are fixed before either group's labels arrive. All predictor and reliance updates affect only future batches.

## C Online reference calibration

### C.1 Expert probabilities and observations

For $\kappa \in \{1/4, 1, 4\}$ and $\beta \in \{1/2, 1, 2\}$, define

$$p_0^{(\kappa,\beta)} = \frac{\kappa p_0^{\mathrm{base}}}{1 - p_0^{\mathrm{base}} + \kappa p_0^{\mathrm{base}}}, \qquad p_c^{(\kappa,\beta)} = (1 - p_0^{(\kappa,\beta)}) \frac{(p_c^{\mathrm{base}})^\beta}{\sum_{k>0}(p_k^{\mathrm{base}})^\beta}. \tag{36}$$

For $N > 0$, Equation (36) scales none-versus-answer-class odds and applies a positive power to the conditional class probabilities. When $N = 0$, every expert is the one-state distribution $(1)$ and the target is structurally zero. The implementation uses log probabilities and log-sum-exp. It returns the identity probabilities unchanged and floors an otherwise unrepresentable positive tail only at the smallest positive floating-point value. This can create ties but does not reverse class order.

For a selected class set $A$ and observed event $o$, define

$$\ell_e(o; A) = \begin{cases} p_c^{(e)}, & \text{a queried class } c \in A \text{ is correct,} \\ p_0^{(e)} + \sum_{c \notin A} p_c^{(e)}, & \text{all queried classes are incorrect.} \end{cases} \tag{37}$$

An empty set supplies probability one and no correctness information. A negative event never supplies an individual unqueried label. For a structurally zero target, the policy queries no class, so this group's likelihood factor is one. Both expert arrays, mixture weights, and query masks are fixed before either group's verification feedback.

### C.2 THE FINITE-MIXTURE IDENTITY

Let $\lambda_{je} = \prod_{g=1}^{2} \ell_{jge} > 0$ and define the cumulative expert loss $L_{Je} = -\sum_{j=1}^{J} \log \lambda_{je}$. Predictions may depend on every previous observation; experts need not be fixed probability vectors over time. Their values at the current batch must be predictable. Equation (10) implies, by induction,

$$\omega_{j,e} = \frac{\omega_{1,e} \prod_{i<j} \lambda_{ie}}{\sum_k \omega_{1,k} \prod_{i<j} \lambda_{ik}}. \tag{38}$$

The normalizing factors telescope. In exact arithmetic,

$$L_J^{\mathrm{mix}} = -\sum_{j=1}^{J} \log Z_j = -\log\left(\sum_e \omega_{1,e} \exp(-L_{Je})\right), \tag{39}$$

$$L_J^{\mathrm{mix}} \le L_{Je} + \log(1/\omega_{1,e}) \quad \text{for every } e. \tag{40}$$

Hence the identity expert incurs a $\log 2$ comparison penalty and each other expert a $\log 16$ penalty. This is the usual finite-expert log-loss argument (Cesa-Bianchi & Lugosi, 2006), applied to coarsened observations. The product components define a forecasting model; this algebra does not assume that the two actual reward groups are statistically independent.

The bound compares forecasts on the realized observation sequence. The mixture's predictable marginal reference yields statewise-unbiased channel estimates by Theorem 2, independently of its forecasting accuracy.

## D COVARIANCE CONTROL ON THE BASE OBSERVATION CHANNEL

### D.1 ENDPOINT CONSTRUCTION AND PREDICTABILITY

Condition on $\mathcal{F}$ and the current base predictor, reliance $t$, public class coefficients and query law. Let $Q_{oh} = \Pr(O = o \mid H = h, \mathcal{F})$, and let $U$ have rows $c_h^\top$. Any slot endpoint $X$ satisfying $Q^\top X = U$ is statewise unbiased. The base and calibrated cyclic endpoints solve this same linear constraint with different positive reference weights. The independent endpoints instead use their respective prior-weighted residuals with the same fixed Bernoulli inclusions. Both constructions satisfy the same constraint. In particular, calibrating an endpoint here does not redesign the query masks.

Let $X_0 = u_{\mathrm{HT}} + t(u_0 - u_{\mathrm{HT}})$ and $\Delta_X = t(u_1 - u_0)$. Then $Q^\top \Delta_X = 0$ and $X_\alpha = X_0 + \alpha \Delta_X$. For $e_{0h}(o) = X_0(o) - c_h$, define

$$J_h = \sum_o Q_{oh} \{e_{0h}(o) \Delta_X(o)^\top + \Delta_X(o) e_{0h}(o)^\top\}, \tag{41}$$

$$\Omega_h = \sum_o Q_{oh} \Delta_X(o) \Delta_X(o)^\top \succeq 0. \tag{42}$$

Every quantity is determined before the current labels. Direct expansion gives

$$V_h(\alpha) = V_h(0) + \alpha J_h + \alpha^2 \Omega_h. \tag{43}$$

Every matrix is computable from the public coefficients, references and query law before feedback.

### D.2 FEASIBLE INTERVAL AND SCALAR OBJECTIVE

For $x, y \in [0,1]$ and $\lambda \in [0,1]$,

$$\lambda V_h(x) + (1-\lambda) V_h(y) - V_h(\lambda x + (1-\lambda) y) = \lambda(1-\lambda)(x-y)^2 \Omega_h \succeq 0. \tag{44}$$

The set satisfying all the covariance inequalities is therefore convex. It is closed, bounded, and contains zero, so it equals $[0, \bar{\alpha}]$ for some $\bar{\alpha} \in [0,1]$. Write

$$\eta_2 = \sum_h p_h^{\mathrm{mix}} \operatorname{tr} \Omega_h \ge 0, \qquad \eta_1 = \sum_h p_h^{\mathrm{mix}} \operatorname{tr} J_h. \tag{45}$$

The reference risk is $R(0) + \eta_1\alpha + \eta_2\alpha^2$. If $\eta_2 > 0$, its exact constrained minimizer is

$$\alpha^\star = \min\{\bar{\alpha}, \max\{0, -\eta_1/(2\eta_2)\}\}. \tag{46}$$

If $\eta_2 = 0$, strict positivity of the reference and nonnegativity of every $\operatorname{tr}\Omega_h$ imply that $\Delta_X$ is zero on possible observations. Then $\eta_1 = 0$ and zero is a minimizer. If $\eta_1 \geq 0$, zero also minimizes the risk on the nonnegative interval. In every case the attainable reference risk is no larger than $R(0)$.

### D.3 Proof of Theorem 4

The shared constraint $Q^\top X_\alpha = U$ proves the stated conditional mean. Consequently, $V_h(\alpha)$ is both conditional error second moment and conditional covariance. Multiplying the feasible matrix inequality on the left by $S^\top$ and on the right by $S$ preserves positive semidefiniteness. This proves Equation (15), even for singular $S$ or covariance matrices.

For groups $g = 1, 2$, condition additionally on both fixed hidden states. Their query masks are independent under the executed product law, and their coefficient errors each have mean zero. Hence the raw batch-gradient covariance is

$$\operatorname{Cov}\left(\frac{1}{2}\sum_{g=1}^{2} S_g^\top u_{\alpha_g,g} \,\middle|\, H_1, H_2, \mathcal{F}\right) = \frac{1}{4}\sum_{g=1}^{2} S_g^\top V_{g,H_g}(\alpha_g) S_g. \tag{47}$$

Applying the same $\delta$ to both summands yields the theorem. Shared predictable forecasts do not change this conditional independence of query draws.

### D.4 Numerical implementation

The implementation uses the same $\delta = .1$ across all models, designs and seeds. When $\eta_2 > 0$ and $\eta_1 < 0$, it first tests $\alpha = 1$ and otherwise uses 64 bisection steps to find the feasible interval endpoint. It takes $\min\{\bar{\alpha}, -\eta_1/(2\eta_2)\}(1 - 10^{-6})$. Values below $10^{-8}$ become zero. For each state the eigenslack is $2{\cdot}10^{-12}\max\{\|V_h(0)\|_2, 10^{-300}\}$; the covariance recomputed from the returned coefficients is checked with twice this slack. Reference risk is checked with $2{\cdot}10^{-12}\max\{R(0), 10^{-300}\}$. If a returned-coefficient check fails, the interpolation is halved and rechecked, reverting to zero below the same threshold. These tolerances remain fixed throughout the search. The implementation records interpolation, contractions, each state tolerance and minimum eigenvalue, and both reference risks.

The returned-matrix acceptance criterion is

$$V_h(\alpha) \preceq (1+\delta)V_h(0) + \tau_h I_K, \qquad \tau_h = 4\cdot 10^{-12}\max\{\|V_h(0)\|_2, 10^{-300}\}. \tag{48}$$

Congruence by $S$ transfers the additive term to $\tau_h S^\top S$. This states the numerical tolerance explicitly. Theorem 4 concerns the exact-arithmetic constraint; an additive tolerance need not be equivalent to a multiplicative allowance when $V_h(0)$ is singular.

At $t = 0$ the calibration direction is zero. A structurally zero target is returned without a query.

## E Training and evaluation details

**Exact answer grammar and class construction.** Decode completion tokens with special tokens removed. Take the text after the last `####` marker and strip surrounding whitespace. Remove at most one final period, then require the entire remaining string to be a signed decimal number. An integer part may contain commas only in groups of three after an initial one-to-three-digit group. A fractional part requires at least one digit after the decimal point; a leading decimal point is allowed. Fractions, exponents, units, currency symbols, explanations, NaN and infinity are invalid. Parse with exact decimal arithmetic, remove thousands separators, write fixed-point notation, remove fractional trailing zeros and normalize every signed zero to `0`. Thus `#### 42`, `#### 42.0` and `#### +0042.` share a class; `#### 1,000` and `#### 1000.00` share another. `#### 1/2`, `#### 1e3` and `#### 42 dollars` are invalid. Missing markers are invalid even when another number appears in the reasoning. The same parser processes the gold answers. Correctness is equality of these canonical strings, so two different valid classes cannot both pass. Classes are numbered by first

occurrence, and a selected class queries its first completion. Invalid completions have known zero reward. If all completions are invalid, or all belong to one valid class, every admissible normalized update is zero and no training query is acquired. A valid class mixed with invalid completions does not satisfy this shortcut.

**Prompts and decoding.** Use each model's chat template with a generation prompt. The system message is "You are a helpful assistant who solves mathematics problems." The user message is the GSM8K question followed by two newlines and:

> Solve the problem carefully and concisely. Use a plain decimal number without units or a currency symbol. End your response with a final line in this format:
> `#### <number>`

Training samples from the full-vocabulary policy at temperature one with top-$p = 1$ and top-$k = 0$. Evaluation is greedy (one completion per question), in batches of eight, without self-consistency or best-of-$n$ selection. Both use at most 256 new tokens, repetition penalty one, the tokenizer's EOS token, left padding and a maximum prompt length of 2,048. EOS is included in the scored completion; padding is excluded. Evaluation counts a verifier call even for an invalid completion.

**Trainable parameters and update.** Load the instruction-tuned base model in bfloat16 with scaled-dot-product attention. Attach fresh LoRA adapters to `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj` and `down_proj`, with rank 16, scaling parameter 32 and dropout zero. Initialization seed 25000 is shared within each model. Only adapters are trainable; no additional SFT or KL anchor is used in these experiments. The model remains in evaluation mode for scoring, while autograd stays enabled. Accumulate the negative detached-weighted sum of completion log probabilities over microbatches, divide by the batch's number of groups, clip the joint gradient norm at one, then take one AdamW step. The default Adam moments are $(0.9, 0.999)$ with epsilon $10^{-8}$ and zero weight decay. The learning rate is constant for all 400 updates. Explicit zero gradients still advance the optimizer, including its accumulated moments.

**Randomness and training partition.** For training seed $s$, the task generator uses NumPy's default generator seeded by `SeedSequence([s,20])`; it draws task indices with replacement. Generation at step $j$ uses the first generated integer from `SeedSequence([s,30,j])` as the Torch seed. Group $g$ uses an independent query generator initialized with `SeedSequence([s,40,j,g])`. Two-group arms draw two tasks per step; the single-group control draws one. The 256 training questions were selected without reading answers. The ordered zero-based GSM8K train indices below specify the partition directly. The evaluator uses all test indices 0 through 1318 in order.

> 468, 2201, 3400, 3036, 3171, 4417, 7097, 6534, 4894, 6504, 1087, 2924, 7108, 6380,
> 4765, 1940, 3194, 3604, 1191, 224, 616, 6048, 6712, 5817, 557, 5562, 3930, 279, 932,
> 5171, 5658, 1142, 4924, 1536, 6404, 7327, 6023, 5692, 2588, 5745, 288, 229, 2060, 7268,
> 3634, 7011, 3328, 2474, 7136, 6987, 3503, 1793, 5121, 6510, 387, 6501, 6064, 6584,
> 3148, 4415, 7416, 3098, 3457, 4038, 7178, 6, 4442, 599, 6279, 2086, 6715, 4538, 7392,
> 6127, 2219, 6464, 1403, 5353, 7044, 5433, 7051, 855, 5921, 4067, 6029, 7134, 2732, 946,
> 6411, 6063, 1754, 1973, 2761, 4214, 1084, 4885, 1604, 3102, 2410, 717, 3446, 1875,
> 2620, 3582, 2579, 7083, 3798, 7338, 786, 3579, 6636, 1893, 3283, 2463, 2130, 5892,
> 3365, 5290, 2398, 6882, 7309, 5029, 2307, 3983, 1652, 46, 901, 253, 5670, 4652, 4537,
> 2883, 7234, 4456, 4201, 6947, 6533, 4724, 2817, 4840, 6297, 5627, 6499, 2403, 3368,
> 7165, 3449, 4889, 2392, 7402, 6438, 6755, 6906, 5527, 6007, 6632, 1933, 5805, 3838,
> 4034, 3380, 4295, 2313, 996, 6585, 956, 3296, 6778, 4862, 5022, 5789, 6323, 1983, 1770,
> 4251, 5193, 4168, 3382, 2306, 2781, 528, 3046, 3718, 3674, 6860, 2773, 1646, 7307,
> 3338, 3386, 7218, 7276, 203, 1243, 87, 596, 1432, 3669, 2867, 4598, 885, 5589, 6005,
> 907, 5944, 522, 1740, 3174, 3394, 5224, 971, 1984, 1201, 3180, 5915, 5480, 5690, 3341,
> 6976, 618, 2103, 5056, 1680, 150, 1482, 1655, 6009, 2833, 2902, 4463, 2987, 5807, 4775,
> 5628, 214, 6605, 3547, 5657, 6814, 6977, 7223, 4711, 1668, 4852, 6929, 2157, 5853,
> 2412, 1677, 3001, 3721, 5934, 1064, 4359, 5929, 209

**Numerical environment.** The runtime uses Python 3.12.7, PyTorch 2.8.0, Transformers 4.57.1 and PEFT 0.17.1 on NVIDIA L20D GPUs. Deterministic algorithms are enabled for generation and score backpropagation. Attention uses the PyTorch math SDPA backend; flash, memory-

efficient and cuDNN attention kernels and reduced-precision math-SDPA reductions are disabled. The cuBLAS workspace configuration is `:4096:8`, with cuDNN benchmarking disabled.

**Online numerical design.** The cyclic solver initializes at the uniform-rotation law and uses SLSQP with analytic derivatives, objective tolerance $10^{-11}$, at most 100 iterations, the stated exploration floor and expected-cost constraint. It accepts only feasible laws whose reference objective does not increase; a numerical failure uses the original feasible law. Independent allocation solves the scalar water-filling problem stated in Appendix B. The independent inclusion floor is $10^{-6}$; a 100-step scalar bisection enforces the budget, assigning any residual budget to zero-importance classes after positive-importance classes saturate. Both channels retain positive inclusion. The separate guard uses the bisection, contraction and tolerance rules in Appendix D.

The analysis chronology and test-set reuse are stated in Section 7.

# F Mean-centered rewards without standard-deviation normalization

With the same score convention, remove $\sigma(r)$ from Equation (1). Let $\bar{s} = K^{-1}\sum_i s_i$. The resulting target is

$$f_{\mathrm{mean}}(r) = \frac{1}{K}\sum_i (r_i - \bar{r})s_i = \frac{1}{K}\sum_i r_i(s_i - \bar{s}). \tag{49}$$

If slot $i$ is verified with probability $q_i > 0$, the estimator

$$\widehat{f}_{\mathrm{mean}} = \frac{1}{K}\sum_i \frac{\mathbf{1}\{i \text{ verified}\}}{q_i} r_i(s_i - \bar{s}) \tag{50}$$

is unbiased without a restriction on joint reward support. A public prediction $m_i$ yields the corresponding generalized difference by adding $K^{-1}\sum_i m_i(s_i - \bar{s})$ and replacing $r_i$ by $r_i - m_i$ in the correction. This is the direct linear alternative relevant to Dr. GRPO's removal of reward-standard-deviation normalization (Liu et al., 2025); the complete Dr. GRPO training recipe also specifies token normalization and optimization conventions.

When mutually exclusive classes are available, set $B^{\mathrm{mean}}_{hi} = (\mathbf{1}\{i \in I_h\} - \nu_h)/K$. A queried positive class $h$ contributes $B^{\mathrm{mean}}_{h,:}/q_h$; an all-negative observation contributes zero. This class-level HT estimator is unbiased. The same finite-channel projection can also be applied to $B^{\mathrm{mean}}$. By Theorem 2, it minimizes reference covariance among unbiased outputs on the fixed observation channel. Thus removing the standard deviation makes a simple reward-linear estimator available, while projection remains a variance-design choice.

# G Complete results for the common training configuration

## G.1 Calibration diagnostics

For each active generated group $g$, define $R_g(\alpha) = \sum_h p^{\mathrm{mix}}_{g,h} \operatorname{tr} V_{g,h}(\alpha)$. Both endpoints are evaluated under the same current calibrated reference, query law and reliance value. The pooled risk reduction is

$$100\left(1 - \frac{\sum_g R_g(\alpha_g)}{\sum_g R_g(0)}\right),$$

where each sum includes all active groups across the five training seeds of one model. Structurally zero groups have zero risk and contribute zero to either sum. Nonzero-interpolation percentages and mean interpolation are calculated over active groups. Public decision time includes query design and construction of all endpoint tables; it excludes generation, backpropagation and future-feedback processing.

## G.2 Per-seed accuracy and verification

Seeds are 32101–32105. Each row reports accuracy on the same complete test set after 400 updates. Query counts use frequency-ordered hit-and-stop and a separate initially empty answer cache for

Table 5: Operation of calibration in the structured method. Each model has 4,000 generated groups across five seeds. Risk reduction refers to the pooled reference coefficient metric defined above.

| Model | Active groups | $\alpha > 0$ (%) | Mean $\alpha$ | Risk drop (%) | ms/batch |
|---|---|---|---|---|---|
| Qwen 3B | 1420 | 96.41 | 0.120 | 3.56 | 4.45 |
| Qwen 7B | 805 | 95.28 | 0.156 | 2.83 | 3.18 |
| Smol 1.7B | 3476 | 97.09 | 0.081 | 1.60 | 12.66 |

each run; every returned training label is identical to the original observation. The half-batch control generates 3,200 completions. The small-group control uses two completions per question and generates 1,600. All main-comparison rows generate 6,400.

Table 6: Qwen 3B: every training seed. Entries are accuracy (%) / checks.

| Method | 32101 | 32102 | 32103 | 32104 | 32105 |
|---|---|---|---|---|---|
| Structured (ours) | 77.79 / 115 | 79.91 / 119 | 82.18 / 159 | 81.20 / 140 | 80.06 / 150 |
| Full feedback | 80.29 / 159 | 68.54 / 179 | 79.83 / 214 | 80.36 / 168 | 77.18 / 163 |
| Uniform HT | 77.41 / 125 | 76.04 / 179 | 67.93 / 127 | 74.83 / 137 | 77.63 / 133 |
| Independent | 79.30 / 137 | 81.58 / 123 | 80.44 / 105 | 75.21 / 116 | 75.82 / 130 |
| No expert calibration | 79.53 / 108 | 81.43 / 118 | 78.70 / 139 | 81.80 / 134 | 79.15 / 117 |
| Half-batch full | 79.38 / 140 | 78.92 / 121 | 80.74 / 135 | 80.36 / 149 | 80.74 / 128 |
| Small-group full | 79.76 / 137 | 78.17 / 165 | 72.78 / 140 | 40.79 / 172 | 76.88 / 176 |

Table 7: Qwen 7B: every training seed. Entries are accuracy (%) / checks.

| Method | 32101 | 32102 | 32103 | 32104 | 32105 |
|---|---|---|---|---|---|
| Structured (ours) | 88.32 / 59 | 87.87 / 70 | 87.57 / 66 | 86.35 / 66 | 86.81 / 61 |
| Full feedback | 86.28 / 91 | 88.02 / 111 | 88.55 / 120 | 86.73 / 113 | 88.63 / 111 |
| Uniform HT | 86.96 / 62 | 87.11 / 56 | 86.96 / 70 | 87.64 / 60 | 86.96 / 68 |
| Independent | 88.10 / 63 | 88.40 / 58 | 87.04 / 58 | 88.17 / 61 | 86.66 / 56 |
| No expert calibration | 88.48 / 68 | 87.26 / 72 | 87.87 / 63 | 87.34 / 55 | 84.84 / 62 |
| Half-batch full | 88.25 / 61 | 87.64 / 69 | 87.64 / 81 | 87.26 / 60 | 88.10 / 93 |
| Small-group full | 86.20 / 72 | 87.79 / 65 | 87.41 / 68 | 87.49 / 64 | 88.02 / 72 |

Table 8: Smol 1.7B: every training seed. Entries are accuracy (%) / checks.

| Method | 32101 | 32102 | 32103 | 32104 | 32105 |
|---|---|---|---|---|---|
| Structured (ours) | 52.01 / 494 | 52.92 / 466 | 52.31 / 477 | 52.77 / 492 | 50.64 / 481 |
| Full feedback | 52.92 / 772 | 51.93 / 752 | 51.40 / 721 | 51.86 / 726 | 52.16 / 718 |
| Uniform HT | 50.57 / 561 | 48.52 / 413 | 51.10 / 498 | 44.66 / 539 | 50.34 / 485 |
| Independent | 52.08 / 443 | 52.84 / 461 | 52.31 / 437 | 51.40 / 465 | 50.27 / 469 |
| No expert calibration | 52.62 / 496 | 51.63 / 480 | 51.25 / 504 | 51.25 / 467 | 51.86 / 468 |
| Half-batch full | 52.24 / 569 | 50.49 / 512 | 50.87 / 492 | 53.30 / 458 | 45.49 / 520 |
| Small-group full | 50.19 / 501 | 45.11 / 563 | 49.43 / 553 | 47.99 / 530 | 50.11 / 526 |

### G.3 Accuracy-difference intervals

We use 50,000 paired crossed-bootstrap draws. Each draw resamples five seed indices and 1,319 question indices independently; both methods use the same sampled indices. The reported difference is averaged across the resulting crossed array. Intervals use linear quantiles, nominal familywise error rate $0.05$, and percentile endpoints $0.05/(2\times 21)$ and $1-0.05/(2\times 21)$. They are approximate intervals from five training seeds and the question sample, rather than an equivalence test. The initial policy is shared by all five seeds, so its one evaluation vector is repeated across seed rows.

Table 9: Structured minus each comparison at update 400, in percentage points. All intervals use one family of 21 comparisons.

| Model | Comparison | Difference [interval] |
|---|---|---|
| Qwen 3B | Full feedback | $+2.99$ $[-2.49, +10.36]$ |
| Qwen 3B | Uniform HT | $+5.46$ $[-0.05, +13.21]$ |
| Qwen 3B | Independent | $+1.76$ $[-2.86, +6.60]$ |
| Qwen 3B | No expert calibration | $+0.11$ $[-3.12, +3.81]$ |
| Qwen 3B | Half-batch full | $+0.20$ $[-2.62, +2.84]$ |
| Qwen 3B | Small-group full | $+10.55$ $[-1.85, +34.55]$ |
| Qwen 3B | Initial policy | $+10.10$ $[+6.05, +14.04]$ |
| Qwen 7B | Full feedback | $-0.26$ $[-2.56, +2.27]$ |
| Qwen 7B | Uniform HT | $+0.26$ $[-1.83, +2.27]$ |
| Qwen 7B | Independent | $-0.29$ $[-2.39, +1.49]$ |
| Qwen 7B | No expert calibration | $+0.23$ $[-1.74, +2.48]$ |
| Qwen 7B | Half-batch full | $-0.39$ $[-2.17, +1.44]$ |
| Qwen 7B | Small-group full | $+0.00$ $[-2.22, +2.45]$ |
| Qwen 7B | Initial policy | $+6.41$ $[+3.22, +9.51]$ |
| Smol 1.7B | Full feedback | $+0.08$ $[-2.86, +2.99]$ |
| Smol 1.7B | Uniform HT | $+3.09$ $[-0.84, +8.34]$ |
| Smol 1.7B | Independent | $+0.35$ $[-2.24, +2.96]$ |
| Smol 1.7B | No expert calibration | $+0.41$ $[-2.50, +3.26]$ |
| Smol 1.7B | Half-batch full | $+1.65$ $[-2.02, +5.82]$ |
| Smol 1.7B | Small-group full | $+3.56$ $[-0.43, +8.43]$ |
| Smol 1.7B | Initial policy | $+19.61$ $[+15.42, +23.82]$ |

### G.4 Exact hit-and-stop calculation

For each full-feedback group, sort the distinct valid classes by decreasing number of occurrences, breaking ties by first occurrence. Verify one representative at a time. A positive result determines the whole reward state, so checking stops immediately. If none is positive, all $N$ classes are checked. A structurally zero target needs no check. If the positive class has rank $j$ in this order, the cost is $j$; otherwise it is $N$.

The complete class labels from each full-feedback batch determine this cost exactly. We reconstructed every full-feedback batch coefficient vector and checked that it equals the original vector. Consequently the policy update, AdamW moments, future generated trajectory and evaluation are unchanged. The calculation changes query accounting only and does not measure a sequential verifier's wall-clock execution. Table 10 also gives first-occurrence ordering.

Table 10: Full-feedback verification calls at 400 updates. All entries are five-seed means.

| Model | Groups / size | All classes | Stop: first | Stop: frequency |
|---|---|---|---|---|
| Qwen 3B | 2 / 8 | 626.2 | 369.2 | 320.2 |
| Qwen 3B | 1 / 8 | 381.6 | 228.0 | 196.4 |
| Qwen 3B | 2 / 2 | 264.4 | 226.6 | 226.6 |
| Qwen 7B | 2 / 8 | 328.6 | 218.8 | 197.6 |
| Qwen 7B | 1 / 8 | 163.4 | 108.2 | 98.2 |
| Qwen 7B | 2 / 2 | 112.8 | 97.6 | 97.6 |
| Smol 1.7B | 2 / 8 | 2576.6 | 1619.0 | 1413.0 |
| Smol 1.7B | 1 / 8 | 1373.0 | 848.2 | 737.2 |
| Smol 1.7B | 2 / 2 | 907.8 | 813.4 | 813.4 |

### G.5 Verification service with a per-run answer cache

Each run starts with an empty cache keyed by training question. A verified negative answer is stored by its canonical string. Once any answer is verified positive, the unique correct canonical string is known, so every future answer to that question can be judged from the cache. No cache entries are shared across methods, seeds, models or with test evaluation.

The original method still chooses its randomized subset before feedback. We serve these requested classes in descending frequency order, with first-occurrence ties, and groups in batch order. A cached answer needs no external call. A fresh positive answer determines all remaining requested labels; fresh negatives are stored. Structurally zero targets make no request. The label vector is returned in the original request order. Every returned label is checked against the recorded observation. Thus the estimator, optimizer trajectory and accuracy stay fixed while the number of new external calls is determined exactly. For a run, that number equals the number of distinct verified negative pairs plus the number of questions whose positive answer was first acquired.

Table 11 separates requested class labels, within-group stopping without a persistent cache, and stopping with the cache. These are alternative label-service counts on the same trajectories, not measurements of a sequential verifier's elapsed time.

Table 11: Training verification calls at 400 updates, averaged across five seeds. The last column is used in the main comparisons.

| Model | Method | Requested | Stop only | Stop + cache |
|---|---|---|---|---|
| Qwen 3B | Structured (ours) | 184.0 | 183.2 | 136.6 |
| Qwen 3B | Full feedback | 626.2 | 320.2 | 176.6 |
| Qwen 3B | Uniform HT | 203.6 | 182.0 | 140.2 |
| Qwen 3B | Independent | 181.4 | 157.4 | 122.2 |
| Qwen 3B | No expert calibration | 166.6 | 166.0 | 123.2 |
| Qwen 3B | Half-batch full | 381.6 | 196.4 | 134.6 |
| Qwen 3B | Small-group full | 264.4 | 226.6 | 158.0 |
| Qwen 7B | Structured (ours) | 80.0 | 80.0 | 64.4 |
| Qwen 7B | Full feedback | 328.6 | 197.6 | 109.2 |
| Qwen 7B | Uniform HT | 87.2 | 79.6 | 63.2 |
| Qwen 7B | Independent | 81.8 | 75.4 | 59.2 |
| Qwen 7B | No expert calibration | 82.8 | 82.8 | 64.0 |
| Qwen 7B | Half-batch full | 163.4 | 98.2 | 72.8 |
| Qwen 7B | Small-group full | 112.8 | 97.6 | 68.2 |
| Smol 1.7B | Structured (ours) | 663.0 | 654.0 | 482.0 |
| Smol 1.7B | Full feedback | 2576.6 | 1413.0 | 737.8 |
| Smol 1.7B | Uniform HT | 734.2 | 650.4 | 499.2 |
| Smol 1.7B | Independent | 678.8 | 598.4 | 455.0 |
| Smol 1.7B | No expert calibration | 668.0 | 658.4 | 483.0 |
| Smol 1.7B | Half-batch full | 1373.0 | 737.2 | 510.2 |
| Smol 1.7B | Small-group full | 907.8 | 813.4 | 534.6 |

### G.6 Generation and optimization cost

Time below is the sum of recorded generation, query-decision, verification, gradient-update and controller-feedback timings. It excludes test evaluation, model loading, serialization and queueing. Hardware is NVIDIA L20D. The times describe the original batched label service; latency of the stopping/cache service is not measured. Completion count matches across the main $G = 2$, $K = 8$ methods, while generated token lengths and nonzero-score work can vary.

## H Reconstruction and calibration on fixed query channels

### H.1 Exact comparisons on a fixed observation channel

The learning comparisons change the policy trajectory as well as the estimator. To isolate reconstruction, we instead fix each generated group, query law and prequery history from the 15 Structured runs. This gives 12,000 groups, of which 5,701 are active. For every active group we enumerate its finite set of observations and evaluate each estimator on the same channel.

Table 12: Training cost at 400 updates. Mean generated tokens (thousands), backward calls and recorded training minutes across five seeds.

| Model | Method | Tokens (k) | Backwards | Minutes |
|---|---|---|---|---|
| Qwen 3B | Structured (ours) | 823.7 | 554.0 | 90.2 |
| Qwen 3B | Full feedback | 764.5 | 448.0 | 82.5 |
| Qwen 3B | Uniform HT | 804.3 | 144.0 | 85.9 |
| Qwen 3B | Independent | 732.9 | 526.4 | 79.6 |
| Qwen 3B | No expert calibration | 799.7 | 505.2 | 91.5 |
| Qwen 3B | Half-batch full | 416.4 | 280.0 | 69.8 |
| Qwen 3B | Small-group full | 222.2 | 104.0 | 83.6 |
| Qwen 7B | Structured (ours) | 743.4 | 310.0 | 64.2 |
| Qwen 7B | Full feedback | 760.0 | 293.6 | 63.4 |
| Qwen 7B | Uniform HT | 803.1 | 86.8 | 75.8 |
| Qwen 7B | Independent | 769.3 | 309.2 | 71.6 |
| Qwen 7B | No expert calibration | 765.5 | 298.4 | 68.2 |
| Qwen 7B | Half-batch full | 379.1 | 155.2 | 49.8 |
| Qwen 7B | Small-group full | 206.1 | 60.6 | 64.7 |
| Smol 1.7B | Structured (ours) | 911.8 | 1364.8 | 83.5 |
| Smol 1.7B | Full feedback | 856.6 | 1062.0 | 65.6 |
| Smol 1.7B | Uniform HT | 861.5 | 294.8 | 65.9 |
| Smol 1.7B | Independent | 865.2 | 1354.8 | 77.3 |
| Smol 1.7B | No expert calibration | 882.7 | 1353.6 | 68.5 |
| Smol 1.7B | Half-batch full | 459.2 | 565.2 | 54.8 |
| Smol 1.7B | Small-group full | 220.8 | 211.4 | 60.9 |

Let $Q_{oh} = \Pr(O = o \mid H = h, \mathcal{F})$ and let $c_h$ be the slot target. For an observation-dependent slot output $X$, define

$$\mathcal{R}_h(X) = \sum_o Q_{oh} \|X(o) - c_h\|^2, \qquad \mathcal{R}_p(X) = \sum_h p_h \mathcal{R}_h(X). \tag{51}$$

The true-state error uses the recorded group's canonical correct class, or state zero if none is correct. Those labels are read after training for this diagnostic and are never supplied to an online controller. Structurally zero groups contribute zero to every risk sum. No Monte Carlo sampling of verification masks is needed.

The generalized-difference output on the same, possibly dependent, subset law is

$$X_{\mathrm{GD}}(A, H) = \sum_{h=1}^{N} p_h c_h + \sum_{h \in A} \frac{\mathbf{1}\{H = h\} - p_h}{q_h} c_h. \tag{52}$$

It is statewise unbiased because each class has marginal inclusion $q_h$. Dependence between checks affects its covariance, so we evaluate Equation (51) directly. We compare HT, this generalized difference, the channel projection, and both endpoints mixed with HT using the same recorded reliance $t$. The latter comparison holds the query law, reference and reliance fixed while replacing only the reconstruction endpoint.

For each model and metric, the tables use ratios of summed risks across its five training seeds. A reference-risk ratio uses the original learned reference $p$ for both numerator and denominator. A true-state ratio conditions on each group's actual state. These are conditional estimator measurements on the Structured trajectories, with the exact finite-channel expectation taken separately for each group.

The true-state projection-to-generalized-difference error ratios are 0.999221, 1.000001, 0.999190. Under these query laws, the two reconstruction endpoints produce closely matched errors. Their larger reduction relative to HT reflects the predictive correction shared by both estimators.

### H.2 Expert forecasts and the cost of the covariance constraint

To separate expert calibration from its constraint, retain the same base endpoint $X_0$, direction $\Delta_X = t(u_1 - u_0)$ and calibrated reference $p^{\mathrm{mix}}$ as in Appendix D. The unconstrained comparison minimizes

Table 13: Same-channel coefficient error, divided by HT error. Reference risk uses the original learned reference; true-state risk uses each generated group's actual reward state. All entries are ratios of summed exact risks across five seeds.

| | Qwen 3B | | Qwen 7B | | Smol 1.7B | |
|---|---|---|---|---|---|---|
| Estimator | Reference | True state | Reference | True state | Reference | True state |
| HT | 1.00000 | 1.00000 | 1.00000 | 1.00000 | 1.00000 | 1.00000 |
| Generalized difference | 0.53954 | 0.31899 | 0.51438 | 0.32264 | 0.70140 | 0.49992 |
| Channel projection | 0.53903 | 0.31875 | 0.51429 | 0.32264 | 0.69955 | 0.49952 |
| Mixed generalized difference | 0.54936 | 0.34106 | 0.53303 | 0.36024 | 0.70575 | 0.50808 |
| Mixed channel projection | 0.54885 | 0.34081 | 0.53295 | 0.36025 | 0.70394 | 0.50772 |

the same quadratic risk on $[0, 1]$:

$$\alpha_{\text{free}} = \text{clip}_{[0,1]}\{-\eta_1/(2\eta_2)\}, \qquad X_{\text{free}} = X_0 + \alpha_{\text{free}}\Delta_X, \tag{53}$$

with $\alpha_{\text{free}} = 0$ when $\eta_2 = 0$. The guarded comparison uses the recorded $\alpha$ and the same current history. Thus removing the constraint preserves the interpolation objective and its search interval.

We measure forecast quality by the negative log probability of the true state. Calibration risk is evaluated under the same $p^{\text{mix}}$ for the base, unconstrained and guarded outputs. A group violates the covariance cap if any state has

$$\lambda_{\min}\{(1+\delta)V_h(0) - V_h(\alpha)\} < -4 \cdot 10^{-12} \max\{\|V_h(0)\|_2, 10^{-300}\}.$$

This is the returned-coefficient tolerance used by the implementation, with $\delta = .1$ in the reported training runs. We count the constraint as binding when $\alpha < (1 - 2 \cdot 10^{-6})\alpha_{\text{free}} - 10^{-8}$, excluding the fixed inward numerical contraction. Forecast loss and interpolation averages use active groups; pooled risk ratios include all groups, with zero contributions from structurally zero targets.

Table 14: Forecasts and interpolation on active groups. Mixture loss refers to the calibrated reference. Binding counts use the criterion defined above to exclude the fixed inward numerical step. Log loss is measured at the actual reward state.

| Model | Active | Base loss | Mixture loss | $\alpha_{\text{free}}$ | $\alpha$ | Binding (%) |
|---|---|---|---|---|---|---|
| Qwen 3B | 1420 | 0.830 | 0.601 | 0.964 | 0.120 | 96.41 |
| Qwen 7B | 805 | 0.831 | 0.643 | 0.953 | 0.156 | 95.28 |
| Smol 1.7B | 3476 | 1.131 | 0.973 | 0.971 | 0.081 | 97.01 |

Table 15: Calibration risk and covariance control. Risk ratios use the base mixed estimator as denominator and the same calibrated reference in the numerator and denominator of each reference-risk ratio. Cap violations count active groups with at least one violating reward state.

| Model | Interpolation | Reference ratio | True-state ratio | Cap violations (%) |
|---|---|---|---|---|
| Qwen 3B | Unconstrained | 0.83369 | 0.74965 | 96.41 |
| Qwen 3B | Guarded | 0.96444 | 0.95292 | 0.00 |
| Qwen 7B | Unconstrained | 0.88787 | 0.81943 | 95.28 |
| Qwen 7B | Guarded | 0.97170 | 0.95974 | 0.00 |
| Smol 1.7B | Unconstrained | 0.88894 | 0.81897 | 97.01 |
| Smol 1.7B | Guarded | 0.98395 | 0.97664 | 0.00 |

Guarded calibration reduces pooled true-state coefficient error by 4.71%, 4.03%, 2.34%. Unconstrained interpolation attains lower pooled error on these trajectories while exceeding the prescribed covariance cap for many possible states. The guard therefore trades some average-error improvement for control in each state. These comparisons evaluate the current estimators on fixed histories; separate online runs would be needed to measure the corresponding policy-learning trajectories.

### H.3 PER-SEED CONDITIONAL MEASUREMENTS

The following table reports each training seed as one trajectory unit. Each entry aggregates its 800 generated groups before taking a ratio. No individual group is treated as an independent training run.

Table 16: Per-seed true-state coefficient-risk ratios on Structured trajectories. Projection/GD fixes the query law, prior and pure endpoint; guarded/base and free/base fix the law, prior history and recorded reliance.

| Model | Seed | Active | Projection/GD | Guarded/base | Free/base |
|---|---|---|---|---|---|
| Qwen 3B | 32101 | 243 | 0.999681 | 0.95492 | 0.74702 |
| Qwen 3B | 32102 | 238 | 0.998911 | 0.95766 | 0.79037 |
| Qwen 3B | 32103 | 339 | 0.998431 | 0.94409 | 0.68613 |
| Qwen 3B | 32104 | 288 | 0.999328 | 0.94969 | 0.72395 |
| Qwen 3B | 32105 | 312 | 0.999802 | 0.95894 | 0.80305 |
| Qwen 7B | 32101 | 152 | 0.999624 | 0.95516 | 0.80871 |
| Qwen 7B | 32102 | 166 | 1.000215 | 0.95908 | 0.84875 |
| Qwen 7B | 32103 | 155 | 1.000097 | 0.95720 | 0.78429 |
| Qwen 7B | 32104 | 164 | 1.000082 | 0.95414 | 0.76496 |
| Qwen 7B | 32105 | 168 | 0.999962 | 0.97102 | 0.87858 |
| Smol 1.7B | 32101 | 707 | 0.999635 | 0.97686 | 0.82054 |
| Smol 1.7B | 32102 | 692 | 0.998972 | 0.97590 | 0.79839 |
| Smol 1.7B | 32103 | 674 | 0.998616 | 0.97601 | 0.82348 |
| Smol 1.7B | 32104 | 692 | 0.999239 | 0.97584 | 0.81403 |
| Smol 1.7B | 32105 | 711 | 0.999491 | 0.97853 | 0.83765 |

## I GROUP SAMPLING AND OPTIMIZER DIAGNOSTICS

### I.1 UNIFORM CLASS HT AND WHOLE-GROUP HT

Consider two sampling rules for the same generated group. Uniform class HT independently includes each valid class with probability $\rho$. Whole-group HT draws $Z \sim \mathrm{Bernoulli}(\rho)$, obtains full feedback if $Z = 1$, and returns $Z v_H/\rho$. Under the mutually exclusive reward support, for each fixed $H = h > 0$,

$$\widehat{v}_{\mathrm{class}} = \frac{\mathbf{1}\{h \in A\}}{\rho} v_h \overset{d}{=} \frac{Z}{\rho} v_h = \widehat{v}_{\mathrm{group}}. \tag{54}$$

When $H = 0$, both estimates are zero. This proves equality of their conditional update distributions, including degenerate cases with $v_h = 0$.

With independent draws for each group, their batch-update distributions also agree. Conditional on a shared policy and optimizer state, identical clipping and AdamW rules map these equal distributions to equal next-state distributions. Induction then gives equality in law of the policy trajectories when prompt sampling, generation and optimizer advancement follow the same rules. The full-label draw must be independent for each group; one draw shared across a batch changes the covariance. Both methods must advance explicit zero-gradient steps, and neither may update an additional predictor or sampler from the extra labels. The Uniform HT control meets these conditions. Unweighted subsampling and skipping optimizer steps define different controls.

This equivalence concerns learning signals. The observed subsets, rejected labels, stopping opportunities and persistent cache contents can differ. Before stopping and caching, both rules request $\rho N$ labels in expectation on a fixed active group, but their external-call costs under those service rules need not agree. Consequently, the Uniform HT accuracy control represents the learning law of the weighted whole-group rule; its reported verification count belongs to class sampling.

### I.2 CLIPPING DURING THE COMPLETE TRAINING RUNS

For every one of the 105 runs, we read the recorded global gradient norm immediately before clipping, after all microbatch gradients have accumulated. A step is clipped when this norm exceeds

one. The all-step denominator is 400 per run. A scored step has at least one nonzero reconstruction coefficient; this definition does not assume that such coefficients must produce a nonzero gradient. In these runs the two counts coincide. Explicit zero-gradient AdamW steps remain in the all-step denominator.

Table 17: Gradient clipping at threshold one. All-step rates include explicit zero-gradient AdamW steps. Scored-step rates condition on at least one nonzero slot coefficient. Each model-method row pools five seeds and 2,000 optimizer steps. Norms are measured before clipping, after microbatch accumulation.

| Model | Method | All steps (%) | Scored steps (%) | Median positive norm |
|---|---|---|---|---|
| Qwen 3B | Structured | 56.50 | 99.82 | 38.67 |
| Qwen 3B | Full feedback | 47.90 | 100.00 | 49.63 |
| Qwen 3B | Uniform HT | 17.20 | 100.00 | 158.01 |
| Qwen 3B | Independent | 53.60 | 99.63 | 35.91 |
| Qwen 3B | No expert calibration | 52.05 | 99.71 | 37.08 |
| Qwen 3B | Half-batch full | 35.00 | 100.00 | 76.00 |
| Qwen 3B | Small-group full | 26.00 | 100.00 | 74.00 |
| Qwen 7B | Structured | 34.50 | 99.71 | 32.91 |
| Qwen 7B | Full feedback | 33.10 | 100.00 | 39.02 |
| Qwen 7B | Uniform HT | 10.50 | 100.00 | 136.30 |
| Qwen 7B | Independent | 34.70 | 99.28 | 28.62 |
| Qwen 7B | No expert calibration | 33.35 | 99.55 | 31.01 |
| Qwen 7B | Half-batch full | 19.40 | 100.00 | 71.86 |
| Qwen 7B | Small-group full | 15.15 | 100.00 | 60.39 |
| Smol 1.7B | Structured | 95.50 | 99.84 | 13.41 |
| Smol 1.7B | Full feedback | 87.50 | 100.00 | 15.28 |
| Smol 1.7B | Uniform HT | 33.60 | 100.00 | 41.65 |
| Smol 1.7B | Independent | 95.00 | 99.58 | 12.09 |
| Smol 1.7B | No expert calibration | 95.40 | 99.95 | 12.79 |
| Smol 1.7B | Half-batch full | 70.65 | 100.00 | 22.58 |
| Smol 1.7B | Small-group full | 52.85 | 100.00 | 19.45 |

Across methods and models, 99.28–100% of scored steps are clipped. Thus the learning experiments operate predominantly with norm-rescaled nonzero updates. The unbiasedness and covariance theorems describe the raw estimator before this nonlinear map. The training comparisons measure the resulting clipped AdamW dynamics. A sparse method can have fewer clipped steps among all 400 updates because it returns zero more often, even when every nonzero update is clipped.

### I.3 Seed-level interval sensitivity

For each of the same 21 comparisons, let $d_s$ be the accuracy difference in percentage points on the fixed 1,319 test questions for seed $s$. With five seeds, the paired Student interval is

$$\bar{d} \pm t_{4,1-\gamma/2}\frac{s_d}{\sqrt{5}}, \qquad s_d^2 = \frac{1}{4}\sum_{s=1}^{5}(d_s - \bar{d})^2. \tag{55}$$

We report $\gamma = .05$ and $\gamma = .05/21$. The critical values are 2.77645 and 6.84714. These normality-based intervals describe variation across training seeds conditional on this fixed test set; the crossed bootstrap also resamples questions. They are sensitivity analyses of the same outcomes, not additional independent evaluations.

Every family-adjusted interval against a trained control contains zero. Five seeds provide limited information about the tails of training variation. There are $\binom{9}{4} = 126$ seed-count vectors in a size-five bootstrap resample; their probabilities differ, and question resampling produces additional statistic values. More bootstrap draws reduce Monte Carlo error without adding independent training seeds.

Table 18: Paired seed-level Student intervals for Structured minus each comparison, in percentage points. The test set is fixed. Both columns use four degrees of freedom; the last applies the same 21-comparison Bonferroni family as the crossed-bootstrap analysis. Marginal intervals are descriptive sensitivity checks.

| Model | Comparison | Marginal 95% | Family-adjusted |
|---|---|---|---|
| Qwen 3B | Full feedback | $[-3.39, +9.36]$ | $[-12.74, +18.71]$ |
| Qwen 3B | Uniform HT | $[-1.22, +12.14]$ | $[-11.01, +21.93]$ |
| Qwen 3B | Independent | $[-2.48, +5.99]$ | $[-8.69, +12.21]$ |
| Qwen 3B | No expert calibration | $[-2.57, +2.79]$ | $[-6.50, +6.72]$ |
| Qwen 3B | Half-batch full | $[-1.39, +1.79]$ | $[-3.72, +4.11]$ |
| Qwen 3B | Small-group full | $[-10.79, +31.89]$ | $[-42.07, +63.18]$ |
| Qwen 3B | Initial policy | $[+8.05, +12.14]$ | $[+5.05, +15.14]$ |
| Qwen 7B | Full feedback | $[-2.05, +1.53]$ | $[-4.67, +4.16]$ |
| Qwen 7B | Uniform HT | $[-1.01, +1.52]$ | $[-2.86, +3.38]$ |
| Qwen 7B | Independent | $[-1.46, +0.88]$ | $[-3.17, +2.59]$ |
| Qwen 7B | No expert calibration | $[-1.17, +1.63]$ | $[-3.22, +3.68]$ |
| Qwen 7B | Half-batch full | $[-1.22, +0.43]$ | $[-2.43, +1.65]$ |
| Qwen 7B | Small-group full | $[-1.68, +1.68]$ | $[-4.14, +4.14]$ |
| Qwen 7B | Initial policy | $[+5.42, +7.40]$ | $[+3.97, +8.86]$ |
| Smol 1.7B | Full feedback | $[-1.41, +1.56]$ | $[-3.59, +3.74]$ |
| Smol 1.7B | Uniform HT | $[-0.88, +7.07]$ | $[-6.70, +12.89]$ |
| Smol 1.7B | Independent | $[-0.39, +1.09]$ | $[-1.47, +2.17]$ |
| Smol 1.7B | No expert calibration | $[-1.12, +1.94]$ | $[-3.37, +4.19]$ |
| Smol 1.7B | Half-batch full | $[-1.21, +4.51]$ | $[-5.40, +8.70]$ |
| Smol 1.7B | Small-group full | $[+0.04, +7.09]$ | $[-5.12, +12.25]$ |
| Smol 1.7B | Initial policy | $[+18.48, +20.73]$ | $[+16.83, +22.38]$ |

### I.4 Sensitivity to the covariance allowance

Using the same 5,701 active groups as Appendix H, we hold the generated completions, base and calibrated references, query law and recorded reliance fixed. For each $\delta \in \{0, .05, .1, .5\}$, we solve the scalar reference-risk problem with $V_h(\alpha) \preceq (1+\delta)V_h(0)$ for every state, using the same bisection, inward contraction and returned-matrix tolerances. The zero-allowance case is solved as a constraint problem; its returned interpolation is zero for all these groups.

Table 19: Covariance-allowance sensitivity on fixed Structured histories. Ratios compare each constrained estimator with the same base mixed estimator, using the calibrated reference or the true state. Each allowance reoptimizes the interpolation; no policy is retrained. Every row has zero violating groups under its own cap and the stated numerical tolerance.

| Model | $\delta$ | Mean $\alpha$ | Reference ratio | True-state ratio |
|---|---|---|---|---|
| Qwen 3B | 0.00 | 0.000 | 1.00000 | 1.00000 |
| Qwen 3B | 0.05 | 0.061 | 0.98132 | 0.97545 |
| Qwen 3B | 0.10 | 0.120 | 0.96444 | 0.95292 |
| Qwen 3B | 0.50 | 0.535 | 0.88027 | 0.83045 |
| Qwen 7B | 0.00 | 0.000 | 1.00000 | 1.00000 |
| Qwen 7B | 0.05 | 0.079 | 0.98509 | 0.97906 |
| Qwen 7B | 0.10 | 0.156 | 0.97170 | 0.95974 |
| Qwen 7B | 0.50 | 0.637 | 0.90763 | 0.85871 |
| Smol 1.7B | 0.00 | 0.000 | 1.00000 | 1.00000 |
| Smol 1.7B | 0.05 | 0.041 | 0.99164 | 0.98787 |
| Smol 1.7B | 0.10 | 0.081 | 0.98395 | 0.97664 |
| Smol 1.7B | 0.50 | 0.353 | 0.94151 | 0.91292 |

The $\delta = .1$ calculation reproduces the existing conditional risks to floating-point precision. Increasing the allowance from .05 to .5 reduces pooled true-state coefficient error more strongly on each model, while permitting a larger worst-state covariance increase. The value .1 specifies a 10% covariance allowance throughout training; this sensitivity study characterizes that choice without

selecting a new training setting from test accuracy. These rows are fixed-history estimator comparisons. They do not measure separate learned policies or risk under the actual score Gram.

# J Resource scenarios and query-law diagnostics

## J.1 Resource prices and verification uncertainty

Equation (16) assigns a common cost to each generated completion and each new verifier call. If Structured uses $\Delta n_{\text{gen}} > 0$ additional completions and saves $\Delta n_{\text{ver}} > 0$ calls relative to a control, its break-even price ratio is $\Delta n_{\text{gen}}/\Delta n_{\text{ver}}$. The ratio is conditional on that comparator and service. It does not describe a universal task requirement or enforce an equal-accuracy constraint.

For a separate time-based scenario, let $T_0$ be the recorded training time excluding original verification time and let $\ell$ be additional serial latency per external call. Then $T(\ell) = T_0 + \ell n_{\text{ver}}$. At the observed mean counts, Structured crosses full feedback at $\ell = 11.53$, $1.02$ and $4.19$ seconds for Qwen 3B, Qwen 7B and Smol. It crosses half-batch full at $102.71$ seconds for Qwen 7B and $61.10$ seconds for Smol, and never for Qwen 3B. These scenarios assume unchanged non-verification work, serial calls and no additional cache-management cost. They combine measured training components with replayed call counts; they are not measured speedups. Against Independent, the completion-price model has no positive crossover. In the time model, Structured is cheaper on Qwen 7B below 85.33 seconds per call and more expensive above it; Qwen 3B and Smol have no positive crossover.

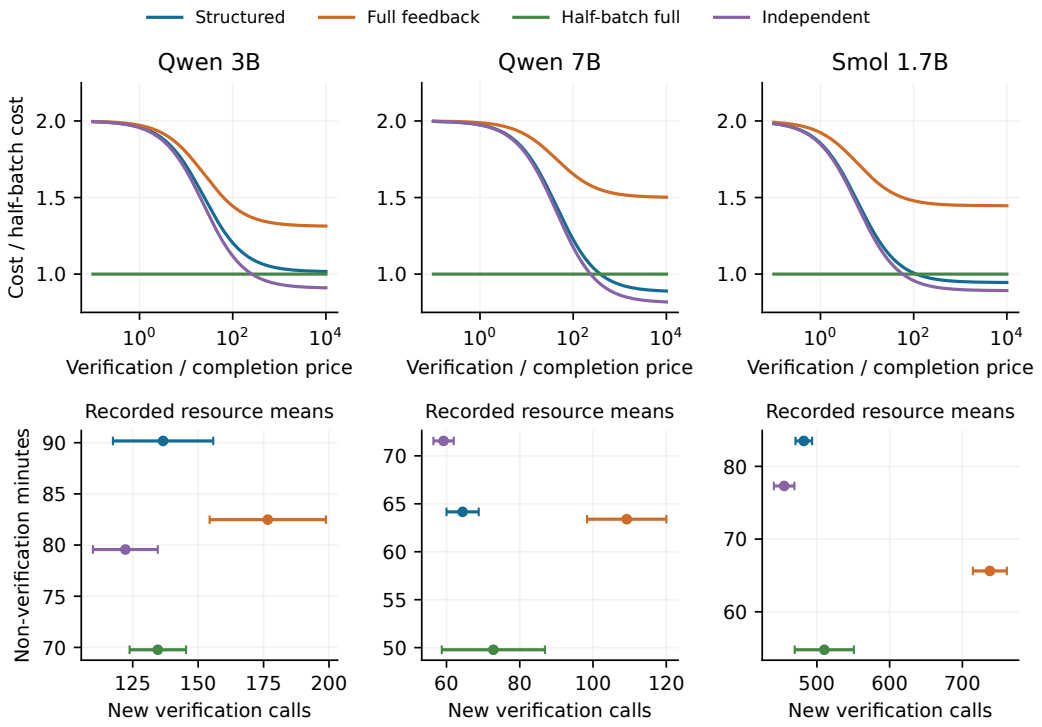


Figure 4: Resource tradeoffs under the shared answer cache. Top: mean-only cost scenarios relative to half-batch full. The crossings are plug-in estimates: cached-call saving intervals include zero, so a finite half-batch break-even threshold is not established (Table 20). Bottom: mean external calls and recorded non-verification training time; horizontal bars show sample standard deviations of call counts across five seeds. Accuracy is reported separately in Tables 1–2.

For each control and service, we form five paired call differences, control minus Structured. Table 20 reports the mean and $t_4$ intervals with critical values $t_{4,.975}$ and $t_{4,1-.05/(2\cdot18)}$. The 18-comparison family contains three models, three controls and two services. It describes this posthoc cost analysis separately from the accuracy family. These model-based intervals remain approximate with five

seeds. All family-adjusted intervals for savings against half-batch full contain zero. For the cached service, every marginal interval also contains zero; a ratio dividing by those uncertain savings has no established finite break-even value.

Table 20: Calls saved by Structured relative to each control: control minus Structured. Intervals are paired seed-level $t_4$ intervals, conditional on this workload. The last column adjusts one family of all 18 model–control–service comparisons; the marginal column is descriptive.

| Model | Control | Service | Saved | Marginal 95% | Family-adjusted |
|---|---|---|---|---|---|
| Qwen 3B | Full feedback | Stop | +137.0 | [+85.5,+188.5] | [+15.0,+259.0] |
| Qwen 3B | Full feedback | Cache | +40.0 | [+15.8,+64.2] | [-17.2,+97.2] |
| Qwen 3B | Half-batch full | Stop | +13.2 | [-40.9,+67.3] | [-114.9,+141.3] |
| Qwen 3B | Half-batch full | Cache | -2.0 | [-28.0,+24.0] | [-63.5,+59.5] |
| Qwen 3B | Independent | Stop | -25.8 | [-85.5,+33.9] | [-167.1,+115.5] |
| Qwen 3B | Independent | Cache | -14.4 | [-50.4,+21.6] | [-99.5,+70.7] |
| Qwen 7B | Full feedback | Stop | +117.6 | [+84.1,+151.1] | [+38.4,+196.8] |
| Qwen 7B | Full feedback | Cache | +44.8 | [+34.1,+55.5] | [+19.6,+70.0] |
| Qwen 7B | Half-batch full | Stop | +18.2 | [+2.7,+33.7] | [-18.6,+55.0] |
| Qwen 7B | Half-batch full | Cache | +8.4 | [-10.6,+27.4] | [-36.6,+53.4] |
| Qwen 7B | Independent | Stop | -4.6 | [-12.9,+3.7] | [-24.3,+15.1] |
| Qwen 7B | Independent | Cache | -5.2 | [-12.5,+2.1] | [-22.5,+12.1] |
| Smol 1.7B | Full feedback | Stop | +759.0 | [+713.1,+804.9] | [+650.5,+867.5] |
| Smol 1.7B | Full feedback | Cache | +255.8 | [+225.6,+286.0] | [+184.2,+327.4] |
| Smol 1.7B | Half-batch full | Stop | +83.2 | [+27.3,+139.1] | [-49.1,+215.5] |
| Smol 1.7B | Half-batch full | Cache | +28.2 | [-22.5,+78.9] | [-91.7,+148.1] |
| Smol 1.7B | Independent | Stop | -55.6 | [-74.4,-36.8] | [-100.2,-11.0] |
| Smol 1.7B | Independent | Cache | -27.0 | [-50.7,-3.3] | [-83.0,+29.0] |

The stop-only service charges repeated answers again; the cached service stores positive and negative answers within each run. A known correct answer also labels future alternatives for that question. Repeated exposure to the 256 training questions makes caching useful here. In a single-pass workload with no repeated question, persistent caching offers no cross-group saving, though duplicate sharing and within-group stopping still apply. Table 11 reports both counts on exactly the same recorded trajectories.

### J.2 Independent query laws on the same histories

For every active Structured group, hold its public coefficient matrix, base reference $p$, recorded reliance $t$ and true state fixed. Compare GD under the recorded cyclic law with independent Bernoulli draws under two marginal choices: the same cyclic inclusion probabilities, or the production allocation

$$q_h = \mathrm{clip}_{[10^{-6},1]}\left(\lambda\sqrt{[p_h - (2t - t^2)p_h^2]\|B_{h,:}\|^2}\right), \qquad \sum_h q_h = \rho N.$$

The scalar $\lambda$ satisfies the budget; calibration does not enter this query allocation. All laws therefore have the same expected requested-label budget, up to the original solver tolerance. For independent draws, the exact risk with effective baseline $m$ is

$$R_H^{\mathrm{ind}} = \sum_h (q_h^{-1} - 1)(\mathbf{1}\{H = h\} - m_h)^2 \|B_{h,:}\|^2.$$

Use $m = p_{1:N}$ for pure GD and $m = tp_{1:N}$ for mixed GD. We independently enumerate all Bernoulli masks and reward states to check these risks and the conditional means. Every table entry divides a pooled sum by the same recorded cyclic HT sum.

Matching the marginals isolates query dependence while keeping the estimator form fixed. The cyclic law has lower pooled true-state error in these histories; reallocating independent marginal probabilities changes this result little. These fixed histories isolate an estimator effect. The separately trained Independent policies generate different answers and cache histories, and request fewer new calls in Table 1.

Table 21: True-state coefficient error on the same Structured histories, divided by the recorded cyclic HT error. GD uses the same base reference for every law. Matched independent draws preserve the cyclic marginal probabilities; allocated independent draws use the production allocation at the recorded reliance. Pure GD and mixed GD share each row's query law.

| Model | Query law | Pure GD | Mixed GD |
|---|---|---|---|
| Qwen 3B | Cyclic | 0.31899 | 0.34106 |
| Qwen 3B | Independent, matched | 0.36272 | 0.38333 |
| Qwen 3B | Independent, allocated | 0.36259 | 0.38320 |
| Qwen 7B | Cyclic | 0.32264 | 0.36024 |
| Qwen 7B | Independent, matched | 0.37838 | 0.41488 |
| Qwen 7B | Independent, allocated | 0.37834 | 0.41484 |
| Smol 1.7B | Cyclic | 0.49992 | 0.50808 |
| Smol 1.7B | Independent, matched | 0.54656 | 0.55400 |
| Smol 1.7B | Independent, allocated | 0.54607 | 0.55352 |

The base mixed projection estimator at the recorded reliance has 6.92%, 11.66% and 1.64% higher pooled true-state error than pure projection on these same laws. Under the construction reference, Proposition 3 already gives a nonnegative mixing penalty relative to projection. The historical-moment rule responds to past prediction error; its precise training definition is in Appendix B.4.

### J.3 What the covariance constraint prevents

Table 22: Conditional error at actual reward states. Counts use all active groups. Each example is selected for the largest absolute increase of unconstrained error over base within its model; it is an extreme case, not a typical group.

| | Qwen 3B | Qwen 7B | Smol 1.7B |
|---|---|---|---|
| Active groups | 1420 | 805 | 3476 |
| Free worse than base | 204 | 106 | 942 |
| Guarded worse than base | 196 | 100 | 924 |
| Free exceeds 1.1 base | 174 | 84 | 816 |
| Example (seed, step, group) | (32105,64,1) | (32103,49,0) | (32105,373,0) |
| True state $H$ | 0 | 0 | 1 |
| Base error | 0.11995 | 0.14123 | 0.37786 |
| Free error | 0.26105 | 0.27265 | 0.55448 |
| Guarded error | 0.13194 | 0.15535 | 0.39342 |

Unconstrained calibration improves pooled error but increases it at the actual reward state for some groups. Table 22 reports all such groups and an explicitly selected extreme case for each model. In the Qwen 3B example, no candidate class is correct: base error is $0.11995$, unconstrained error is $0.26105$, and guarded error is $0.13194$. The guard limits the increase to its 10% allowance; it does not force improvement. The Smol example has a correct generated class and the same qualitative effect. These conditional estimator examples do not show that guarding improves policy accuracy.

The equal Qwen percentages in Tables 5, 14 and 15 count the same observed groups: all 1,369 and 767 groups with positive interpolation also have a binding constraint and an unconstrained cap violation. For Smol, 3,375 groups have positive interpolation, while 3,372 have both a binding constraint and a violation. The indicators were checked group by group. Their agreement on Qwen is empirical, not an identity. The feasible bound still depends on references, query law, reliance and endpoint direction; $\delta$ specifies the permitted covariance increase within that construction.