WHICH ENVIRONMENTS SHOULD LLMS LEARN FROM? PREDICTING RL UTILITY ACROSS SCALE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards can improve language-model reasoning. Yet equal compute can help, do little, or degrade a policy across mathematical prompt distributions. We ask whether measurements from the starting policy can predict a distribution's held-out utility before observing its RL outcomes. We run 32 controlled Qwen3 trajectories from 1.7B to 14B parameters in a 2×2 design that varies group-signal availability and the admission-cost profile. We evaluate 196 policies on a fixed, disjoint 1,000-prompt OpenR1 evaluation panel. Environment choice changes held-out pass@1 by up to 11.4 percentage points at matched generated-action-token budget. Reward variation alone does not explain this spread. We measure admitted advantage mass per generated token, ρ, and its effective coverage across prompts, H. A compact ρ–H utility model is fit through 8B and frozen before inspecting 14B outcomes. At 14B, it achieves 2.29-point curve RMSE, 77.8% pairwise rank accuracy, and 11/12 cell-budget means inside its empirical envelope. A four-scale leave-one-scale-out audit yields 2.09-point RMSE. Fixed-label and crossover rules remain competitive for allocation, while a frozen uniform-mixture test identifies the current boundary of composition transfer. Together, these results provide evidence that pre-RL measurements combined with earlier-scale outcomes can help predict how environment choice changes held-out improvement across model scale and compute.

1 INTRODUCTION

Reasoning models increasingly rely on reinforcement learning with verifiable rewards (RLVR), where mathematical equivalence or executable tests replace learned reward models (Shao et al., 2024; Guo et al., 2025; Yu et al., 2025). Frontier post-training recipes now scale the tasks, environments, and rollouts that supply experience alongside model size and optimization. The DeepSeek-V4.1-Flash report illustrates this shift. It describes a standard sequence of SFT, RL, and distillation with no algorithmic modification, while its substantive changes come from a progressively scaled pipeline of synthesized agent tasks, environments, and rollouts (DeepSeek-AI, 2026).

This raises a basic question: when the algorithm is fixed, which environments convert rollout compute into useful learning? For a given policy, some prompts are solved, some remain out of reach, and others produce reward variation only through long responses excluded from the loss. More rollout compute can therefore amplify unusable experience and yield little policy improvement.

Here, an environment is a prompt distribution under a fixed rollout, verifier, and optimization contract. We cross group-signal availability with the admission-cost profile. The resulting HH, HL, LH, and LL cells are shown in Figure 1 and use the same colors throughout the paper.

We study a concrete decision:

Given a starting policy and a generated-action-token budget, which measurable environment distribution will produce the most held-out improvement?

Figure 1 shows the full study: probe experience under the starting policy, construct controlled cells, measure usable signal through ρ and H, and predict finite-budget utility before observing RL outcomes at a new scale.

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.

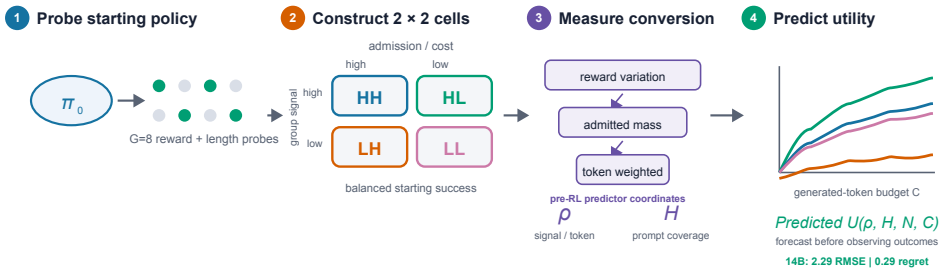

Figure 1: **We measure environments before RL and forecast their finite-budget utility at a new scale.** Policy-relative probes construct four controlled cells. The predictor uses their measured ρ and H coordinates. The pipeline is tested prospectively at 8B and, after recalibration, on untouched 14B.

The key distinction is between *available* and *usable* signal. A prompt group provides group-relative learning signal only when its rewards vary. The loss contract can still remove that signal or concentrate it within a small fraction of prompts. We therefore measure two continuous coordinates. Usable signal density ρ is the admitted leave-one-out advantage mass per generated token. Effective coverage H is the normalized effective sample size of that admitted mass across prompts. An independent pre-RL rollout protocol measures both quantities without using held-out outcomes.

Our study has three parts. First, we establish that environment choice changes learning trajectories under a fixed trainer. At 4B, the best and worst cells differ by 11.4 points at the largest shared generated-token budget. Second, we test whether pre-RL measurements predict that utility across scale. An initial model fit at 1.7B and 4B misses all three 8B best-environment decisions. After scoring that test, we fit a revised model on 1.7B, 4B, and 8B and freeze it before inspecting 14B. It predicts the untouched 14B utility curves with 2.29-point RMSE. Third, we test the model on a frozen uniform mixture assembled from the four cells with an outcome-blind hash rule. That candidate exposes the boundary between transfer across scale and transfer to a new composition.

Prospective chronology is essential because retrospective fit alone cannot distinguish a useful allocator from a fragile explanation. We therefore report each fit before its corresponding scale test. Our four-scale study provides:

1. a controlled demonstration that policy-relative environment choice materially changes held-out RL utility,
2. a measurement decomposition from reward variation to admission and effective prompt coverage,
3. two temporally separated prospective scale tests, including a successful untouched 14B prediction after an informative 8B calibration failure,
4. a compact environment-utility model with low four-scale curve error, together with a frozen uniform-mixture stress test that identifies its current generalization boundary.

Our evidence establishes a predictive environment-utility relationship for one verifiable domain and defines the experimental path toward testing it in additional domains.

2 Related Work

We complement recipe-level scaling by holding the trainer fixed and modeling how environment choice changes finite-budget returns. Prior work selects or generates prompts, modifies optimization, scores individual examples, or fits curves across recipes. Our study measures a fixed candidate distribution under the starting policy and predicts its held-out utility at a new model scale.

Selection, curricula, and self-evolving environments. Data selection, curricula, and task generation address which experience to train on. LIMR and DEPO curate subsets or filter for online explorability. Prompt Curriculum Learning, HIVE, and CoDaPO adapt selection to policy-relative difficulty or reward history (Li et al., 2025; Tang et al., 2025; Gao et al., 2025; Wu et al., 2026; Zhou et al., 2026). Self-Evolving Curriculum and Absolute Zero generate or select verifiable tasks online

(Chen et al., 2025; Zhao et al., 2025a). Our study forecasts fixed candidate distributions before RL and disables dynamic sampling so each intervention remains fixed.

Verifiable-reward post-training. DeepSeekMath introduced GRPO for mathematical reasoning, and DeepSeek-R1 demonstrated reasoning RL at scale (Shao et al., 2024; Guo et al., 2025). DAPO studies dynamic sampling, token-level aggregation, and overlong responses, while Dr.GRPO identifies length and difficulty biases from normalization (Yu et al., 2025; Liu et al., 2025b). DeepSeek-V4.1-Flash keeps the algorithm fixed while scaling synthesized tasks, environments, and rollouts (DeepSeek-AI, 2026). These results motivate separating reward variation from the signal retained by admission and token weighting.

Prompt informativeness and influence. Prompt-level methods use uncertainty, reward history, or group variance to estimate immediate informativeness (Zhao et al., 2025b; Zheng et al., 2025; Jiang et al., 2026), while RFT-Inf estimates retrospective influence along an optimization trajectory (Tan et al., 2025a). Our measurements summarize a distribution before RL by tracking how much group-relative signal survives admission and how broadly that signal is distributed.

Controlled environment allocation. Controlled studies relate environment properties to learning outcomes. Zhu et al. cross reasoning depth with environment complexity, while RLVE studies policy-adaptive difficulty and environment scaling (Zhu et al., 2026; Zeng et al., 2026). Our 2×2 intervention crosses group signal with admission/cost, balances starting success and prompt attributes, then replaces labels with continuous measurements.

Predicting data decisions across scale. DoReMi and RegMix use proxy optimization or regression to choose pretraining mixtures (Xie et al., 2023; Liu et al., 2025a). DataDecide finds that none of eight tested scaling methods improves the compute–decision frontier over single-scale rankings, motivating explicit comparisons with simple decision rules (Magnusson et al., 2025). RL scaling work studies environment dependence, model-size and compute interactions, seed variation, and sigmoidal response curves (Hilton et al., 2023; Tan et al., 2025b; Choshen et al., 2025; Khatri et al., 2026). We fix the recipe, vary the environment, and evaluate both curve error and allocation regret at held-out scales.

3 Environment utility as experience conversion

3.1 Utility and the compute axis

For environment distribution E, starting policy $\pi_0(N)$ with parameter scale N, and cumulative generated-action-token budget C, define

$$U(E, N, C) = P_{\text{heldout}}(\pi_{E,N,C}) - P_{\text{heldout}}(\pi_0(N)). \tag{1}$$

Utility is the change in held-out pass@1 relative to each scale's warm start. Generated action tokens include capped and loss-masked responses. This budget captures realized rollout cost, whereas optimizer steps obscure response-length differences and wall-clock time depends on the system. End-to-end FLOPs and selection cost are reported separately. Matching admitted tokens would condition on a treatment consequence, so we report them as a mechanism diagnostic. By definition, $U(E, N, 0) = 0$. Comparisons at common C measure which environment yields greater held-out improvement from equal generated-token budgets.

3.2 From group signal to usable experience

For prompt group g with $G = 8$ binary rewards r_{gi}, our NeMo contract uses the unnormalized leave-one-out advantage

$$a_{gi} = r_{gi} - \frac{1}{G-1} \sum_{j \neq i} r_{gj}. \tag{2}$$

All-success and all-failure groups have zero advantage. For G conditionally independent responses with prompt success probability p_g, the mixed-group probability is

$$q_{\text{mixed},g} = 1 - p_g^G - (1 - p_g)^G. \tag{3}$$

Equation 3 characterizes group-signal availability. Group reward variance can guide rollout allocation (Jiang et al., 2026). Admission and token weighting then determine how much of that signal survives and how broadly it is distributed. Available signal may be masked, diluted by long trajectories, or concentrated on a few prompts, with different consequences for held-out improvement.

Let $m_{gi} \in \{0, 1\}$ be the admission indicator induced by the fixed loss contract and ℓ_{gi} the generated length. Define raw and admitted absolute advantage mass

$$A_g^{\mathrm{raw}} = \sum_i |a_{gi}|, \qquad A_g^{\mathrm{adm}} = \sum_i m_{gi}|a_{gi}|. \tag{4}$$

For a finite empirical support of n_E equally weighted prompt groups, our primary pre-RL coordinates are

$$\rho(E) = \frac{10^6 \sum_g A_g^{\mathrm{adm}}}{\sum_{g,i} \ell_{gi}}, \tag{5}$$

$$H(E) = \frac{1}{n_E} \frac{(\sum_g A_g^{\mathrm{adm}})^2}{\sum_g (A_g^{\mathrm{adm}})^2}. \tag{6}$$

ρ measures admitted group-relative signal per million generated tokens, while H measures effective prompt coverage. We set $H = 0$ when total admitted mass is zero. All measured environments have $\rho > 0$, so $\log \rho$ is well-defined. Both summarize the scalar advantage coefficients that enter the update, leaving parameter-gradient geometry outside the representation.

3.3 A COMPACT ENVIRONMENT-UTILITY MODEL

To predict how efficiently a measured environment converts RL compute into held-out improvement, we fit

$$\widehat{U} = \beta_0 + \beta_N x_N + \beta_C s(C) + \beta_\rho s(C) z(\log \rho(E)) + \beta_H s(C) z(H(E)), \tag{7}$$

where $s(C) = C/(C + C_0)$, x_N is log model scale, and $z(\cdot)$ standardizes coordinates inside each training fold. For the prospective 14B forecast, $C_0 = 14.36651$M tokens is the earlier-scale median budget, fixed while fitting the five coefficients. We fit Equation 7 over positive generated-token budgets. Its environment contrasts vanish at zero compute, while the exact warm-start anchor $U(E, N, 0) = 0$ is defined separately. Cell labels are not features.

For a given environment, the shared saturation factor produces an increasing curve when the fitted coefficient multiplying $s(C)$ is positive and a decreasing curve when it is negative. Its shared timescale fixes environment ordering within each scale, so Equation 7 cannot forecast budget-dependent preference reversals. Appendix D evaluates anchored and budget-flexible sensitivities to this restriction.

The first three terms describe the shared response to scale and compute. The final two model how admitted signal density (ρ) and effective coverage (H) change that response. Because both coordinates derive from admitted advantage mass and are correlated in the constructed grid, we interpret their coefficients jointly. Pre-RL measurements thus specify finite-budget utility curves for predicting held-out learning.

4 CONTROLLED FOUR-SCALE STUDY

We repeat the 2×2 experiment at four model scales. Within each scale, the warm start, trainer, rollout group size, decoding, verifier, horizon, and held-out panel are fixed. Only the policy-relative prompt distribution changes. We construct cells from each scale's starting policy because a prompt can be solved, unsolved, or informative for different models. Balanced factorial contrasts estimate within-scale environment effects, while continuous pre-RL measurements test transfer across scale and budget. Two seeds capture trajectory variation. Checkpoints trace utility against realized tokens and remain repeated observations.

Models and warm starts. We study dense Qwen3 Base models at 1.7B, 4B, 8B, and 14B parameters (Yang et al., 2025). Each receives the same response-only full-parameter warm-start recipe. The ordered SFT set contains 1,024 mathematics and 1,024 executable code examples, with a separate 256-example validation set. A preregistered gate covering format, coverage, truncation, repetition, and diversity selects the common step-512 checkpoint without using held-out RL accuracy.

Prompt pool and verifier. RL prompts come from a pinned revision of OpenR1-Math-220k (Hugging Face Open R1 Team, 2025a;b). We remove SFT, previous RL, monitoring, and evaluation identities by normalized prompt hash. Mathematical correctness is scored by one frozen Math-Verify contract (Kydlíček & Hugging Face, 2025). Incorrect but parseable answers remain ordinary zero rewards. Verifier infrastructure errors remain distinct from incorrect answers. The held-out panel is exactly hash-disjoint from training, while a lexical audit finds a small number of close problem variants within the shared source family (Appendix A).

Factorial environments. At each scale, policy-relative $G = 8$ measurements construct four disjoint cells of 256 prompts. The first letter denotes group-signal availability (high or low), and the second denotes admission/cost (high admission and lower cost, or low admission and higher cost). Thus HH is high on both axes, HL has high group signal but low admission/higher cost, LH has low group signal but high admission/lower cost, and LL is low on both axes. Cells are balanced on mean starting success, prompt length, source, problem type, parseability, and verifier validity. The admission/cost factor remains coupled, and the design identifies only their joint contrast. The cells create controlled variation in prompt distributions. Equations 5–7 replace these categorical labels with continuous pre-RL measurements for prediction.

Selection chronology. Each scale starts from 4,000 prompts and uses three independent $G = 8$ construction probes followed by one untouched confirmation. Broad non-ranked bands, deterministic hash caps, and joint cell selection are fixed before RL. Appendix A gives the exact rules.

RL contract. Each cell is trained for two fixed passes, 128 optimizer steps, with two independent training seeds. Every update uses four prompts and eight responses per prompt. We use NeMo RL v0.6.0 (NVIDIA, 2026a;b), unnormalized leave-one-out advantages, token-level loss, importance correction, symmetric ratio clipping at 0.2, and no reference KL penalty. Dynamic sampling is disabled. Overlong filtering is enabled, so capped responses remain in the reward baseline but have zero policy-loss weight, matching the documented NeMo semantics (NVIDIA, 2026b). Sampling uses temperature 0.6, top-p 0.95, top-k 20, a 12,288-token response cap, and a 16,384-token total sequence limit. The optimizer is AdamW with learning rate 2×10^{-6}. No adaptive stopping, prompt redraw, or outcome-driven seed replacement is used.

Evaluation and statistical units. The held-out outcome uses one score-blind 1,000-prompt OpenR1 panel with the same prompt template, sampler, response cap, and verifier at every checkpoint. It measures within-source mathematical performance. The primary $G = 1$ matrix contains 196 policies and 196,000 responses. A fresh native $G = 4$ endpoint evaluation contains 36 policies and 144,000 responses. We treat the 16 scale-by-environment cells as primary experimental units, 32 trajectories as seed replicates, and checkpoints as repeated budget observations. Two physical shards are recombined before inference.

5 Environment choice changes utility

Figure 2 shows seed-mean utility at four actual generated-action-token budgets, anchored by the definition-implied warm-start point $U(E, N, 0) = 0$. Within each scale, observations correspond to 25%, 50%, 75%, and 100% of the largest budget jointly supported by all four cells, and each panel retains its physical token range. No single cell dominates across scales and budgets. At the common maximum budget, HH is best at 1.7B and 8B, while HL is best at 4B and 14B. Best-to-worst spreads are 3.74, 11.45, 4.32, and 5.96 points, respectively.

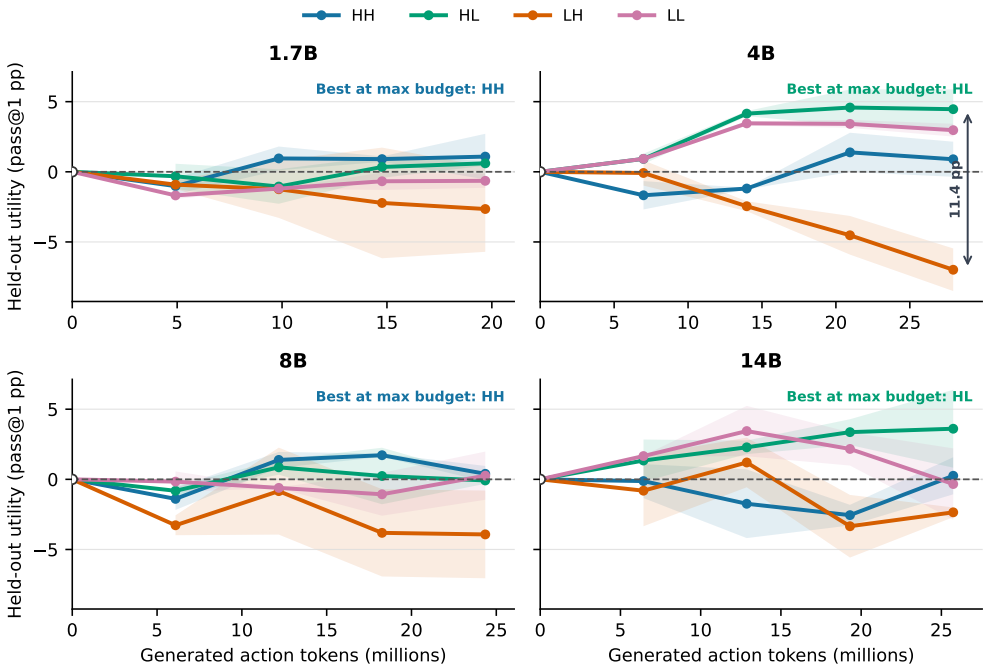

Figure 2: **Environment choice produces large, scale-dependent differences in held-out RL utility.** HH/HL denote high group signal, and HH/LH denote high admission and lower cost. Cells are selected separately under each starting policy. Lines show two-seed means, with bands spanning the observed seeds. Zero-budget anchors are definition-implied and excluded from fitting. Each panel ends at the largest jointly supported budget.

The 4B surface provides the clearest replicated contrast. At step 128, HL improves 5.6 points over the 38.4% warm start while LH loses 7.1 points. At matched generated-token budget the separation remains 11.4 points, with both seeds reproducing the direction. In contrast, one 1.7B LH trajectory degenerates into long, capped, repetitive responses. At its endpoint, 86.8% of responses hit the cap and pass@1 falls to 11.9%. The decline develops across checkpoints and both evaluation shards. Integrity checks reveal no technical explanation, so we retain it as an observed stochastic policy failure.

On a fixed 1,000-prompt Polaris panel, HL remains best, LH remains worst, and the four cell effects correlate 0.991 with their OpenR1 counterparts (Appendix Table 2, panel B).

Neither high mixedness nor low cost alone recovers the scale- and budget-dependent ordering.

6 From available signal to effective coverage

Figure 3 traces normalized effective coverage before admission, after admission, and after token weighting. The predictor coordinate H uses admitted advantage mass from Equation 6. The final diagnostic additionally weights that mass by active token count. This training-log audit covers 1.7B, 4B, and 8B because it preceded the 14B experiment. High-mixed cells retain post-token effective-coverage fractions of roughly 0.60–0.70. Low-mixed cells fall to 0.11–0.18. The heaviest 10% of prompt groups carry 64–86% of the importance-weighted update-coefficient mass in low-mixed environments, compared with 22–27% in high-mixed environments.

The decomposition exposes three bottlenecks. Reward variation determines whether a group has relative signal, admission determines which response-level coefficients remain active, and token weighting determines how broadly they are distributed. Figure 3 reports the resulting coverage loss and concentration. A product of marginal averages such as $q_{\text{mixed}} p_{\text{adm}} / c$ predicts utility less

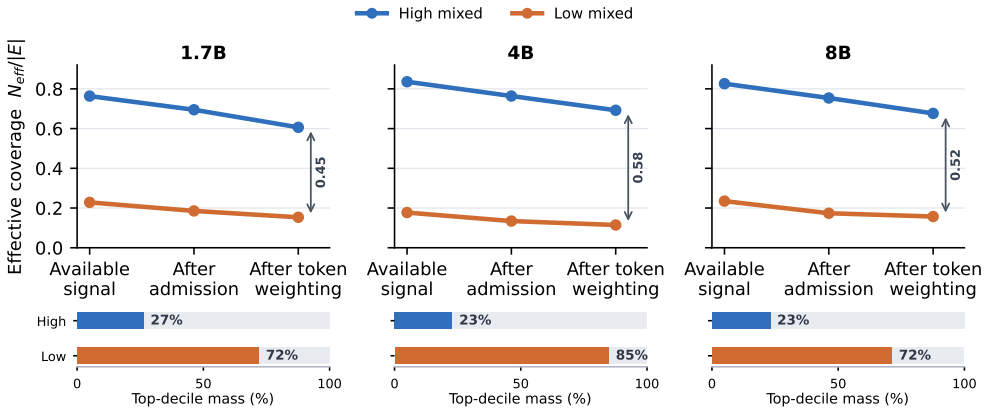

Figure 3: **Admission and token weighting narrow usable signal and concentrate low-mixed updates.** Top: normalized effective prompt coverage across the three stages, with brackets marking the post-token gap. Bottom: importance-weighted update-coefficient mass carried by the heaviest prompt decile. High-mixed environments retain broader coverage, while low-mixed environments concentrate update mass.

accurately than the measured ρ–H representation. Across the 16 cells, however, H correlates 0.999 with q_{mixed} and 0.903 with ρ. We therefore interpret the ρ–H coordinates jointly.

Training reward alone therefore omits both signal survival and concentration. High-mixed environments produce larger reward variation by construction, yet their held-out gains are not uniformly larger. The results support measuring how reward variation is converted into admissible, broadly distributed update signal.

7 Prospective Prediction Across Scale

7.1 The 8B Miss Identifies the Calibration Problem

We first fit prespecified predictors on 1.7B and 4B and froze a cost/admission forecast before observing 8B RL outcomes. The forecast retained 72.2% pairwise rank accuracy and 0.63-point mean regret, but it selected the wrong best cell at all three budgets. Its 90% intervals covered 0/12 cell-budget means. All eight trajectories passed integrity checks. Seed gaps of 3.1–6.1 points were larger than the sub-point predicted decision margins.

The failed transfer identified a calibration defect that retrospective ranking and regret did not reveal. We retained a small mean function, replaced marginal cost/admission proxies with ρ and H measured by the same independent pre-RL $G = 8$ protocol at every scale, and calibrated uncertainty on leave-one-scale-out residual curves. The 8B forecast and its score remain unchanged.

7.2 The Revised Predictor Transfers to Untouched 14B

Equation 7, its coefficients, 14B curves, rankings, and empirical envelope were frozen before inspecting 14B outcomes. Figure 4 compares the two prospective tests. The revised predictor achieves 2.29-point curve RMSE, 77.8% pairwise rank accuracy, and contains 11/12 cell-budget means within its envelope. It selects HL at all three budgets, correctly at 8M and 24M, while LL leads by 0.863 points at 16M. Mean regret is 0.288 points, and fixed HL and the crossover make the same decisions. The observed best-over-runner-up margins are 0.325, 0.863, and 3.643 points, and the seeds agree on the winner only at 24M (Table 4). The prospective result combines accurate curve transfer with competitive allocation decisions, while simple selectors remain equally strong on these three choices.

At step 128, 14B HL reaches 63.35% pass@1, 5.65 points above its 57.7% warm start. It has the smallest observed two-seed endpoint gap, 1.3 points, while HH has a 13.9-point gap. We therefore

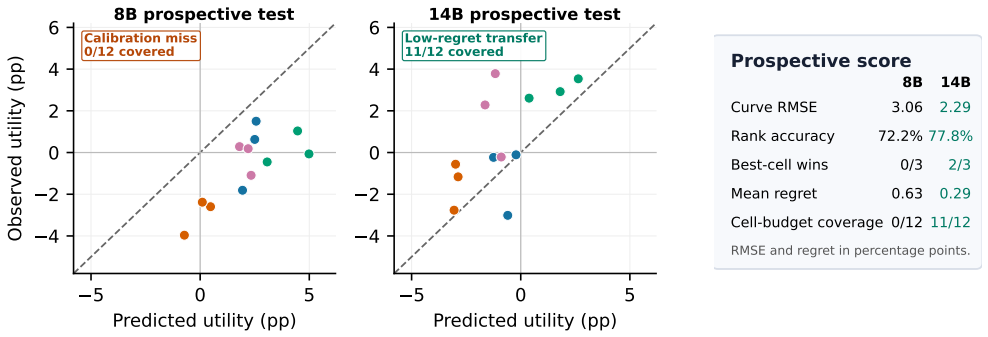

Figure 4: **The revised ρ–H predictor transfers to untouched 14B with low curve error.** The initial 8B test exposes interval miscalibration. The subsequent frozen model performs better on 14B. The tests use different mean models, so their comparison establishes prospective chronology across the full recalibration.

report seed means together with curve-level uncertainty and leave cell-specific stability unresolved with two seeds.

In the native $G = 4$ endpoint check across nine 14B policies and 36,000 responses, per-response pass@1 correlates 0.956 with the primary $G = 1$ surface. HL again leads per-response accuracy, improving 5.90 points. Native pass@4 remains below the warm start for every trained arm, so the observed benefit appears in single-rollout reliability while four-sample problem coverage does not improve.

7.3 Four-scale refit and decision quality

After scoring 14B, we refit the ρ–H family on all four scales, fixing C_0 to the median budget in each training fold. Leave-one-scale-out (LOSO) evaluation yields 2.09-point RMSE, 74.0% pairwise rank accuracy, and 0.48-point mean allocation regret (Table 1). The fit improves utility levels and rankings over budget/scale-only and one-coordinate alternatives, while simple rules remain competitive for allocation.

Table 1: **The ρ–H model improves full-curve prediction while simple selectors remain competitive for allocation.** Panel A evaluates full curves under LOSO. Panel B evaluates decisions at matched generated-token points. RMSE and regret are percentage points.

A. Full-curve prediction

Full utility predictor	LOSO RMSE ↓	Pairwise rank acc. ↑	Mean allocation regret ↓
ρ–H environment-utility model	**2.09**	**74.0%**	**0.48**
Effective coverage (H) only	2.27	63.5%	0.33
Cost/admission	2.51	74.0%	0.48
Budget/scale only	2.40	50.0%	2.05

B. Allocation decisions

Decision rule or portfolio	Mean allocation regret ↓
Success/mixedness crossover	0.36
Training-fold-selected fixed label (HL)	0.48
Raw mixedness	1.13
Baseline success	2.05
Uniform-arm portfolio	2.05

The prespecified success/mixedness crossover yields lower mean allocation regret, 0.36 points, but its 5.65-point LOSO RMSE exceeds the utility model's 2.09. An H-only fit also reaches 0.33-point

regret with weaker curve and ranking accuracy. A fixed-label rule selected within each training fold always chooses HL and matches the utility model's 0.48-point regret. The ρ–H model's advantage lies in full-curve prediction, while simple selectors remain equally strong or better on allocation regret. Following DataDecide, we keep decision quality separate from curve fit and retain these rules as allocation baselines (Magnusson et al., 2025).

Frozen uniform-mixture stress test. We froze a fifth 14B environment by drawing 64 prompts from each cell with an outcome-blind hash rule and training the resulting uniform mixture. The model ranked it last at 8M, 16M, and 24M tokens, predicting negative utility, while observed utility was $+1.36$, $+3.66$, and $+4.58$ points. Its contrast with HL changes sign across seeds, leaving the winner unresolved. The failure of the predicted ordering identifies composition as a missing coordinate (Table 4).

The uncertainty analysis evaluates decision quality under training-seed variation. Using the maximum absolute LOSO residual over each four-budget curve, the empirical curve-level 90% envelope is ± 4.76 points for a two-seed mean curve and ± 5.13 points for one newly trained seed trajectory. These descriptive finite-sample envelopes carry no repeated-test coverage guarantee. Under 20,000 whole-trajectory seed resamples, the ρ–H rule has median mean regret 0.82 points and a 79.7% frequency of regret at or below one point. Two seeds expose consequential variation and leave cell-specific variance unresolved.

8 Implications for changing environments

The four cells provide controlled interventions, while the predictor uses continuous measurements of signal density, effective coverage, and generated cost. Prediction also depends on verifier validity and uncertainty at the intended model scale and budget.

Three principles follow: measure environments under the policy being trained, account for how much signal survives admission and how it is distributed across prompts, and quantify uncertainty. Static difficulty labels leave policy-relative signal unmeasured, and low retrospective regret can coexist with poorly calibrated predictions. The uniform-mixture test shows that successful transfer across model scale can still leave a new composition poorly predicted.

Environment selection is data-mixture allocation with feedback: the policy determines available experience, and training changes that policy. Preferences can change with budget, motivating utility curves that capture these changes. The current model holds environment ordering fixed within each scale. Its predictions can inform allocation when candidates resemble the environments used to fit the model and predicted differences are large relative to training variation.

Extensions to code and agentic tasks require domain-specific rewards, renewed measurement, and re-estimated coefficients. Greater efficiency could also accelerate harmful capability development, so utility predictions must be considered alongside safety criteria.

9 Limitations and conclusion

Our evidence uses one dense model family, one mathematical training source, two mathematical evaluation distributions, one deterministic verifier, and one NeMo GRPO contract. The grid identifies mixed-signal and coupled admission/cost contrasts, requiring joint interpretation of admission and response cost. Two seeds expose substantial trajectory variation and leave a detailed variance model unresolved. The 12,288-token cap defines the result's scope, and effective coverage uses scalar advantage coefficients without parameter-gradient Jacobians. The successful 14B scale test follows one prospective miss, while the uniform-mixture failure limits the allocation claim. Generalization requires independently constructed environments, model families, and domains.

Within this scope, environment choice shifts held-out utility by more than ten points under the same trainer, a spread reward variation alone cannot explain. Usable signal density and effective coverage connect pre-RL experience to an untouched 14B surface with low curve error. The next test is allocation over independently constructed candidates that accounts for measurement cost, followed by transfer to new domains. Together, these results define both the promise and the present boundary of predicting which training experience becomes useful learning at fixed compute.

AI USE STATEMENT

GPT-5.6 Sol was used to assist with debugging training and evaluation infrastructure, analysis and figure-generation scripting, feedback on experimental design and interpretation, and editing for clarity and conciseness. All AI-assisted work was reviewed and verified by the authors, who made all scientific decisions and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

Sections 3–4 specify the estimand, pre-RL measurements, environment construction, training contract, evaluation protocol, and statistical units. Appendix A reports the detailed optimization settings, compute requirements, checkpoint policy, and integrity checks. Appendix B records the prospective chronology, and Appendix C provides additional numerical results. The anonymous supplementary archive contains the exact training and evaluation manifests, claim-bearing NeMo configurations, Stage-D pre-RL measurements, compact evaluation outputs, prospective forecast records, integrity hashes, and scripts used to audit the reported results.

REFERENCES

Xiaoyin Chen, Jiarui Lu, Minsu Kim, Dinghuai Zhang, Jian Tang, Alexandre Piché, Nicolas Gontier, Yoshua Bengio, and Ehsan Kamalloo. Self-evolving curriculum for LLM reasoning. *arXiv preprint arXiv:2505.14970*, 2025. URL https://arxiv.org/abs/2505.14970.

Leshem Choshen, Yang Zhang, and Jacob Andreas. A hitchhiker's guide to scaling law estimation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 10683–10699. PMLR, 2025. URL https://proceedings.mlr.press/v267/choshen25a.html.

DeepSeek-AI. DeepSeek-V4.1-Flash: Pushing the limits of KV cache compression. Technical report, 2026. URL https://huggingface.co/deepseek-ai/DeepSeek-V4.1-Flash/blob/main/DeepSeek_V41_Tech_Report.pdf.

Zhaolin Gao, Joongwon Kim, Wen Sun, Thorsten Joachims, Sid Wang, Richard Yuanzhe Pang, and Liang Tan. Prompt curriculum learning for efficient LLM post-training. *arXiv preprint arXiv:2510.01135*, 2025. URL https://arxiv.org/abs/2510.01135.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Jacob Hilton, Jie Tang, and John Schulman. Scaling laws for single-agent reinforcement learning. *arXiv preprint arXiv:2301.13442*, 2023. URL https://arxiv.org/abs/2301.13442.

Hugging Face Open R1 Team. Open R1: A fully open reproduction of DeepSeek-R1. GitHub repository, 2025a. URL https://github.com/huggingface/open-r1.

Hugging Face Open R1 Team. OpenR1-Math-220k. Hugging Face dataset, 2025b. URL https://huggingface.co/datasets/open-r1/OpenR1-Math-220k/tree/e4e141ec9dea9f8326f4d347be56105859b2bd68. Revision e4e141ec9dea9f8326f4d347be56105859b2bd68.

Heyang Jiang, Henry Liu, and Baharan Mirzasoleiman. Learning as reasoning unfolds: Progressive rollout allocation for efficient reinforcement learning. *arXiv preprint arXiv:2607.22002*, 2026. URL https://arxiv.org/abs/2607.22002.

Devvrit Khatri, Lovish Madaan, Rishabh Tiwari, Rachit Bansal, Venkata Sai Surya Subramanyam Duvvuri, Manzil Zaheer, Inderjit Dhillon, David Brandfonbrener, and Rishabh Agarwal. The art of scaling reinforcement learning compute for LLMs. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.13786.

Hynek Kydlíček and Hugging Face. Math-Verify: Mathematical expression evaluation for language model outputs. GitHub repository, 2025. URL https://github.com/huggingface/Math-Verify.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. LIMR: Less is more for RL scaling. *arXiv preprint arXiv:2502.11886*, 2025. URL https://arxiv.org/abs/2502.11886.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. RegMix: Data mixture as regression for language model pre-training. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2407.01492.

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*, 2025b. URL https://arxiv.org/abs/2503.20783.

Ian Magnusson, Nguyen Tai, Ben Bogin, David Heineman, Jena D. Hwang, Luca Soldaini, Akshita Bhagia, Jiacheng Liu, Dirk Groeneveld, Oyvind Tafjord, Noah A. Smith, Pang Wei Koh, and Jesse Dodge. DataDecide: How to predict best pretraining data with small experiments. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 42487–42502, 2025. URL https://proceedings.mlr.press/v267/magnusson25a.html.

NVIDIA. NeMo RL: A scalable and efficient post-training library. GitHub release v0.6.0, 2026a. URL https://github.com/NVIDIA-NeMo/RL/releases/tag/v0.6.0.

NVIDIA. An in-depth walkthrough of GRPO in NeMo RL, 2026b. URL https://docs.nvidia.com/nemo/rl/0.6.0/guides/grpo.html. NeMo RL v0.6.0 documentation.

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

Haoru Tan, Xiuzhe Wu, Sitong Wu, Shaofeng Zhang, Yanfeng Chen, Xingwu Sun, Jeanne Shen, and Xiaojuan Qi. Understanding data influence in reinforcement finetuning. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3755a02b1035fbadd5f93a022170e46f-Abstract-Conference.html.

Zelin Tan, Hejia Geng, Xiaohang Yu, Mulei Zhang, Guancheng Wan, Yifan Zhou, Qiang He, Xiangyuan Xue, Heng Zhou, Yutao Fan, et al. Scaling behaviors of LLM reinforcement learning post-training: An empirical study in mathematical reasoning. *arXiv preprint arXiv:2509.25300*, 2025b. URL https://arxiv.org/abs/2509.25300.

Xinyu Tang, Zhenduo Zhang, Yurou Liu, Wayne Xin Zhao, Zujie Wen, Zhiqiang Zhang, and Jun Zhou. Towards high data efficiency in reinforcement learning with verifiable reward. *arXiv preprint arXiv:2509.01321*, 2025. URL https://arxiv.org/abs/2509.01321.

Jiahao Wu, Ning Lu, Shengcai Liu, Kun Wang, Yanting Yang, Baijiong Lin, Chen Jason Zhang, Li Qing, and Ke Tang. Train at moving edge: Online-verified prompt selection for efficient RL training of large reasoning model. *arXiv preprint arXiv:2603.25184*, 2026. URL https://arxiv.org/abs/2603.25184.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.10429.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, 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.

Zhiyuan Zeng, Hamish Ivison, Yiping Wang, Lifan Yuan, Shuyue Stella Li, Zhuorui Ye, Siting Li, Jacqueline He, Runlong Zhou, Tong Chen, Chenyang Zhao, Yulia Tsvetkov, Simon Shaolei Du, Natasha Jaques, Hao Peng, Pang Wei Koh, and Hannaneh Hajishirzi. RLVE: Scaling up reinforcement learning for language models with adaptive verifiable environments. *arXiv preprint arXiv:2511.07317v2*, 2026. URL https://arxiv.org/abs/2511.07317v2.

Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *arXiv preprint arXiv:2505.03335*, 2025a. URL https://arxiv.org/abs/2505.03335.

Yang Zhao, Kai Xiong, Xiao Ding, Li Du, Yangou Ouyang, Zhouhao Sun, Jiannan Guan, Wenbin Zhang, Bin Liu, Dong Hu, Bing Qin, and Ting Liu. UFO-RL: Uncertainty-focused optimization for efficient reinforcement learning data selection. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. URL https://arxiv.org/abs/2505.12457.

Haizhong Zheng, Yang Zhou, Brian R. Bartoldson, Bhavya Kailkhura, Fan Lai, Jiawei Zhao, and Beidi Chen. Act only when it pays: Efficient reinforcement learning for LLM reasoning via selective rollouts. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2506.02177.

Zhanke Zhou, Xiangyu Lu, Chentao Cao, Brando Miranda, Tongliang Liu, Bo Han, and Sanmi Koyejo. The easy, the hard, and the learnable: Confidence and difficulty-adaptive policy optimization for LLM reasoning. *arXiv preprint arXiv:2606.07950*, 2026. URL https://arxiv.org/abs/2606.07950.

Yihua Zhu, Qianying Liu, Fei Cheng, Jiaxin Wang, Akiko Aizawa, Sadao Kurohashi, and Hidetoshi Shimodaira. Reasoning depth and environment complexity: A controlled study of RLVR data allocation across logical reasoning tasks. *arXiv preprint arXiv:2605.26934*, 2026. URL https://arxiv.org/abs/2605.26934.

A Experimental contract and integrity

All primary runs use the same reasoning template, binary mathematical-equivalence reward, prompt ordering rule, optimizer family, and checkpoint schedule. Checkpoints are saved at steps 16, 32, 48, 64, 96, and 128. Utility is linearly interpolated against cumulative generated tokens between checkpoints and the definition-implied origin. The primary analysis compares 25%, 50%, 75%, and 100% of the largest budget shared by all cells at a scale, without extrapolation; step-matched endpoints are descriptive.

The four cells are disjoint within each scale. Stage A uses broad, recall-oriented bands: mixed prompts satisfy $0 < K_A < 8$, non-mixed prompts have $K_A \in \{0, 8\}$, high admission requires at least three non-capped responses, and low admission permits at most four. A deterministic hash cap retains at most 640 candidates per band without ranking by extremeness. Stages B and C repeat these prompts with independent $G = 8$ seeds. Selection pools response-level success and admission but averages the three independent $G = 8$ mixedness indicators. It limits mean starting-success spread to 0.5 points and targets a 28.5-point admission contrast. Fresh Stage-D confirmation requires success spread at most 3 points, binary-success standardized difference at most 0.10, and the preregistered admission rule. These gates balance cell means and leave the complete prompt-difficulty distributions unconstrained.

Policy-relative selection produces different prompt sets across scale. Among the 1,024-prompt unions, 1.7B and 4B share 193 prompts, 8B and 14B share 350, and all other scale pairs are disjoint. Normalized hashes give zero overlap with the 1,000-prompt held-out panel. A post hoc word-5-gram screen nevertheless finds 14 evaluation prompts with maximum Jaccard similarity at least 0.7 to a

training prompt, including close variants from the shared OpenR1 source family. Excluding these prompts changes pass@1 by at most 0.68 points across all 196 policies, preserves HL as the final-step leader at every scale, and leaves the 4B endpoint best-to-worst spread at 12.68 points. This screen measures lexical overlap and leaves semantic independence untested, so the evaluation remains a within-source test.

The warm-start SFT recipe uses AdamW at 2×10^{-5} with weight decay 0.1, gradient clipping 1.0, 3% linear warmup, and cosine decay. Step 512 gives equal exposure at every scale. RL uses AdamW at 2×10^{-6}, weight decay 0.01, symmetric clipping at 0.2, BF16, activation checkpointing, and sequence packing. The 1.7B and 4B runs use one B200, while 8B and 14B use two with tensor parallelism. Each B200 provides 180 GB of HBM3e. A 128-step trajectory takes approximately 3.1–5.9, 6.8–9.5, 5.1–6.2, and 9.1–11.0 hours at 1.7B, 4B, 8B, and 14B, respectively. The 32 claim-bearing trajectories total approximately 350 B200 GPU-hours. Six-worker G=1 evaluation at 8B or 14B takes approximately eight hours, followed by two hours for native G=4.

Environment measurement is itself a compute cost. Stage D uses 8,192 responses and verifier calls per scale, generating 70.8M, 68.4M, 69.4M, and 73.9M tokens from 1.7B through 14B. At 14B this final probe alone equals 30.0% of the generated tokens in the eight factorial RL trajectories. Because stages A–C, warm-start SFT, evaluation, and failed attempts add further cost, RL-token regret measures only the training allocation. Earlier-scale outcomes inform the fitted response surface, but the policy-relative coordinates must be measured again under a new starting policy. Weights-only checkpoints omit optimizer and dataloader state, so a failed non-resumable trajectory restarts from the common warm start with the same task and seed.

The Qwen3 model repositories, pinned OpenR1-Math-220k dataset, Math-Verify package, and NeMo RL v0.6.0 are distributed under the Apache License 2.0. We credit their creators and use these assets in accordance with their licenses and terms.

The complete G=1 evaluation contains 49 policies per scale: one warm start and six checkpoints for each of eight trajectories, each scored on 1,000 held-out prompts. Native G=4 covers the warm start and eight step-128 endpoints per scale. Prompt memberships, sample indices, source hashes, configs, response hashes, rewards, and completion markers pass integrity checks. Infrastructure retries rerun the identical task and never promote partial shards.

A separate endpoint audit evaluates the 4B warm start and eight step-128 policies on a fixed 1,000-prompt Polaris mathematical panel. It uses the same native $G = 1$ decode, 12,288-token response cap, verifier, and two 500-prompt shards for every policy. These outcomes remain separate from the OpenR1 utility fit.

B PROSPECTIVE CHRONOLOGY

The 8B cost/admission forecast was serialized before 8B RL outcomes and scored before any three-scale refit, and its result remains unchanged. After scoring that test, we developed the ρ–H experience-conversion family using 1.7B, 4B, and 8B outcomes and one independent pre-RL $G = 8$ measurement at every scale. The numerical 14B curves, rankings, and uncertainty envelope were then serialized before 14B held-out evaluation. The uniform-mixture prompt IDs, training contract, and forecast were also frozen before its RL outcomes. The four-scale refit occurs only after scoring the four-cell 14B forecast. This ordering prevents either 14B result from entering its own prediction. The 8B forecast has SHA-256 prefix `747c1c905966`, the frozen 14B forecast has prefix `7fad0619ecae`, and its scored outcome artifact has prefix `8d755d276f08`.

C ADDITIONAL NUMERICAL RESULTS

Table 2: **HL leads the final-step endpoint comparisons.** Panel A reports OpenR1 pass@1 averaged across two training seeds. Panel B reports the 4B Polaris continuity audit under the same native $G = 1$ contract. Parentheses and deltas show change from the corresponding warm start.

A. OpenR1 endpoint pass@1

Scale	HH	HL	LH	LL
1.7B	19.8 (-0.3)	**23.3 (+3.2)**	16.2 (-3.9)	22.1 (+2.0)
4B	38.8 (+0.4)	**44.0 (+5.6)**	31.3 (-7.1)	40.8 (+2.4)
8B	44.7 (-2.1)	**48.4 (+1.6)**	43.2 (-3.7)	48.0 (+1.2)
14B	58.6 (+0.9)	**63.4 (+5.7)**	55.7 (-2.0)	57.9 (+0.2)

B. 4B Polaris endpoint pass@1

Policy	Seed 9001	Seed 9002	Mean	Δ from π_0
π_0	–	–	17.8	–
HH	19.1	17.3	18.2	$+0.4$
HL	19.7	21.5	**20.6**	$+\mathbf{2.8}$
LH	15.4	13.8	14.6	-3.2
LL	18.7	21.0	19.9	$+2.1$

Table 3: **HL improves per-response pass@1 at every scale, while native pass@4 does not always increase.** Endpoint values are averaged across the two training seeds.

Scale	Policy	Per-response pass@1	Native pass@4
1.7B	π_0 / HL	18.07 / 23.54	43.0 / 50.7
4B	π_0 / HL	37.92 / 42.25	67.2 / 69.2
8B	π_0 / HL	45.05 / 48.66	74.7 / 77.5
14B	π_0 / HL	56.98 / 62.88	89.0 / 88.2

Table 4: **The 14B decision audit separates curve transfer from a unique allocation advantage.** Panel A gives the complete observed surface. Panel B shows that ρ–H, fixed HL, the crossover, and raw mixedness make the same decisions; the earlier 8B model had no serialized 14B decision. Panel C shows that the frozen uniform mixture falls outside the learned ordering. Utilities and margins are percentage points.

A. Observed utility for every factorial cell

Budget	HH	HL	LH	LL
8M	-0.23	$+2.61$	-1.16	$+2.28$
16M	-3.01	$+2.92$	-0.56	$+3.78$
24M	-0.11	$+3.53$	-2.76	-0.21

B. Decisions and seed-level winners

Budget	Best	Runner-up	Margin	Seed 1	Seed 2	ρ–H/fixed/cross./q_{mixed}
8M	HL	LL	0.325	LL	HL	HL / HL / HL / HL
16M	LL	HL	0.863	HL	LL	HL / HL / HL / HL
24M	HL	HH	3.643	HL	LL	HL / HL / HL / HL

C. Frozen uniform-mixture stress test

Budget	Forecast	Observed	Uniform $-$ HL	Seed contrasts
8M	-1.82	$+1.36$	-1.29	$(+1.99, -4.57)$
16M	-2.20	$+3.66$	$+0.64$	$(+2.84, -1.57)$
24M	-2.42	$+4.58$	$+1.03$	$(-2.12, +4.19)$

D MEASUREMENT PROPERTIES AND RETROSPECTIVE SENSITIVITIES

Table 5: **Pre-RL coordinates for the 16 scale-by-environment cells.** Success, mixedness, admission, and H are fractions; length is mean generated tokens per response; ρ is admitted advantage mass per million generated tokens.

1.7B

Cell	Suc.	q_{mixed}	Adm.	Length	ρ	H
HH	.192	.703	.568	7,252	28.34	.621
HL	.157	.676	.272	10,184	13.52	.603
LH	.182	.242	.545	7,175	8.02	.201
LL	.170	.168	.249	9,975	3.85	.145

4B

Cell	Suc.	q_{mixed}	Adm.	Length	ρ	H
HH	.310	.855	.610	7,653	35.22	.744
HL	.284	.836	.366	9,908	20.44	.754
LH	.307	.145	.618	6,827	4.62	.125
LL	.301	.121	.340	8,994	2.22	.107

8B

Cell	Suc.	q_{mixed}	Adm.	Length	ρ	H
HH	.336	.891	.575	8,108	32.43	.781
HL	.331	.844	.366	9,907	20.11	.783
LH	.333	.211	.596	7,017	5.92	.180
LL	.326	.129	.353	8,867	2.27	.114

14B

Cell	Suc.	q_{mixed}	Adm.	Length	ρ	H
HH	.331	.820	.580	8,562	29.13	.743
HL	.316	.852	.350	10,443	18.63	.776
LH	.309	.219	.582	7,767	5.24	.183
LL	.316	.164	.338	9,306	2.94	.149

For a binary-reward group containing k successes, Equation 2 gives

$$A_g^{\text{raw}} = \frac{2k(G-k)}{G-1}, \qquad \mathbb{E}[A_g^{\text{raw}} \mid p_g] = 2Gp_g(1-p_g). \tag{8}$$

Raw absolute advantage mass therefore encodes binary reward variation. The proposed density further accounts for loss admission and generated cost. Whenever total raw mass is positive,

$$\rho = 10^6 \underbrace{\frac{\sum_g A_g^{\text{raw}}}{\sum_{g,i} \ell_{gi}}}_{\text{raw mass per generated token}} \underbrace{\frac{\sum_g A_g^{\text{adm}}}{\sum_g A_g^{\text{raw}}}}_{\text{advantage-weighted survival}}. \tag{9}$$

The survival factor differs from the unweighted response-admission rate. For normalized masses $w_g = A_g^{\text{adm}} / \sum_h A_h^{\text{adm}}$, effective coverage is $H = (n_E \sum_g w_g^2)^{-1}$. Thus $1/n_E \le H \le 1$ when admitted mass is nonzero, and H is invariant to a common positive rescaling of all masses. These identities describe coefficient mass and concentration. Gradient direction, semantic diversity, and alignment with held-out tasks lie outside this definition.

The prospective 14B model was fit by unregularized ordinary least squares to 48 two-seed mean observations: four environments, four matched budgets, and three earlier scales. Every cell-budget point receives equal weight. The scale coordinate is $x_N = \log(N/1.7)/\log(4/1.7)$. The frozen coefficients, in pass@1 fractions, are $(-0.006221, -0.008124, 0.023489, -0.076673, 0.093314)$ in the order $(\beta_0, \beta_N, \beta_C, \beta_\rho, \beta_H)$. The training-fold standardizers are $(2.280467, 0.975934)$ for $\log \rho$ and $(0.429913, 0.290342)$ for H, reported as mean and standard deviation. No 14B outcome enters these values.

For LOSO evaluation, RMSE gives equal weight to the 64 scale-cell-budget means. Pairwise rank accuracy compares the six unordered cell pairs at every scale-budget point. An exact predicted or observed tie receives one-half credit. Mean allocation regret averages $\max_E U(E, N, C) - U(\widehat{E}, N, C)$ over the 16 scale-budget decisions. If several cells tie for the highest predicted utility, their observed utilities are averaged. The curve-level envelope is calibrated from each held-out cell's maximum absolute residual across its four budgets. The reported 79.7% is a conditional frequency under resampling the 32 observed seed trajectories, with no direct deployment-probability interpretation.

Table 6 reports additional retrospective checks using C_0 computed only from each training fold. This change leaves the selected model's RMSE unchanged to two decimals. An anchored four-parameter form slightly lowers retrospective RMSE and matches its decisions. A two-timescale form allows budget-dependent order changes but does not improve ranking or regret. Both are retrospective sensitivities, and the prospectively scored model remains unchanged.

Table 6: **The compact ρ–H family remains competitive across retrospective sensitivity forms.** Anchored and two-timescale forms were evaluated after 14B and are not prospective results.

Model	RMSE	Pairwise rank acc.	Mean regret
Budget/scale only	2.40	50.0%	2.05
ρ only	2.39	57.3%	2.05
H only	2.27	63.5%	**0.33**
ρ–H	2.09	**74.0%**	0.48
Anchored ρ–H	**2.05**	**74.0%**	0.48
Two-timescale ρ–H	2.09	72.9%	0.60

Coverage alone is competitive for allocation on the four constructed labels, while the combined representation better predicts utility levels and pairwise orderings. A training-fold-selected fixed-label rule chooses HL in every fold and matches the combined model's 0.48-point regret. The crossover rule reaches 0.36 points. These results motivate new candidate distributions and a charged allocation experiment to test whether the model improves decisions beyond simple selectors.