INDEPENDENCE OVER ACCURACY: GENERATOR–VERIFIER ERROR CORRELATION IN RECURSIVE SELF-IMPROVEMENT IN SMALL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Verifier accuracy does not specify whether erroneous feedback repeats or corrects a policy's mistaken judgments. We study this distinction using a binary verdict channel that fixes expected verifier accuracy while varying error correlation with a specified self-judge. Under an explicit conditional-independence assumption, the expected reward difference between correct and incorrect responses with the same self-judgment decreases linearly with correlation when the self-judge error rate is below one half. The resulting correlation tax defines an equivalent independent accuracy by matching conditional reward gaps. At a self-judge error rate of 0.30, a maximally correlated 90% verifier matches the gap of an independent 83.3% verifier, while a maximally correlated 97% verifier retains a 95% equivalent accuracy. Exact analysis identifies the feasible correlation range, the region where independence offsets lower accuracy, and the bounded tax as the self-judge error rate decreases. Reported Monte Carlo channel checks agree with these population calculations. A proposed five-round GRPO–LoRA protocol tests whether the conditional reward gap predicts learning, using controlled and natural verifiers and a frozen control. No language-model training outcome is claimed.

1 INTRODUCTION

Self-improvement methods turn model-generated responses into training data or rewards, but they differ in how those responses are verified. STaR filters and rationalizes generated reasoning, self-rewarding methods ask a model to evaluate its own outputs, and self-play fine-tuning uses another form of model-derived supervision (Zelikman et al., 2022; Yuan et al., 2025; Chen et al., 2024). Other methods use executable environments or agreement among sampled answers (Zhao et al., 2025; Huang et al., 2026; Zuo et al., 2025). These designs make verification part of the learning process. Describing a verifier therefore requires asking both how often it is wrong and which responses receive those mistakes.

Figures 1 and 2 summarize the proposed training loop, controlled intervention and measurement design. Consider a fixed response population whose binary self-judgment is wrong on 30% of examples. Two verifiers can both be 90% accurate while placing their errors differently. One spreads errors independently of the self-judgment; the other concentrates them inside the self-judge's error set. Under the controlled channel studied here, their conditional reward gaps are 0.800 and 0.667. The comparison isolates an error-placement effect at matched expected accuracy.

Theoretical work on sharpening explains how self-training can concentrate probability on selected responses, while generation–verification-gap studies and solver–verifier dynamics examine complementary limits on self-improvement (Huang et al., 2024; Song et al., 2025; Sun et al., 2026). Our object is narrower: correlation between two error indicators on the same response, one from a verifier and one from a specified binary self-judge. This differs from correlation between verifier errors on different responses within a GRPO group. The distinction also prevents treating all same-model or majority-vote judges as identical copies of one self-judgment.

We make three contributions. First, we give a controlled binary channel with explicit feasibility conditions and a closed-form conditional reward-gap identity. Second, we derive its equivalent independent accuracy, crossover boundary and bounded tax, and compare exact calculations with

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.

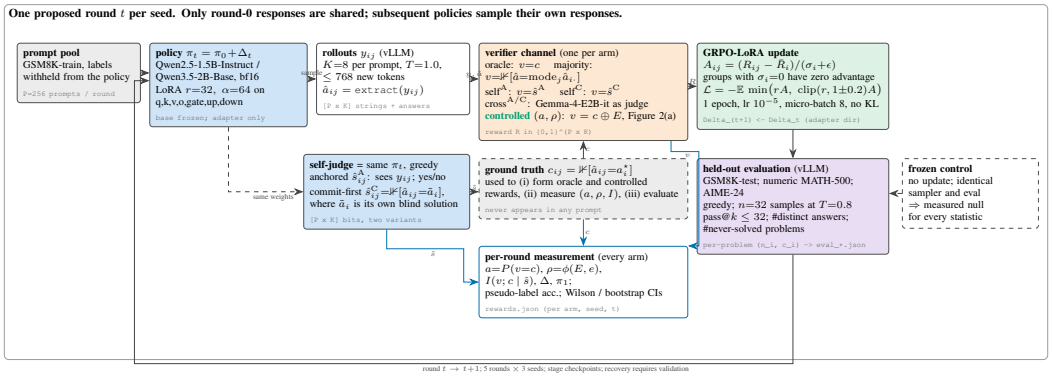

Figure 1: The proposed training and evaluation pipeline. Only initial responses are shared; subsequent policies generate their own responses. Ground truth is used by the oracle, controlled channels, measurements and evaluation. This figure specifies an unexecuted protocol.

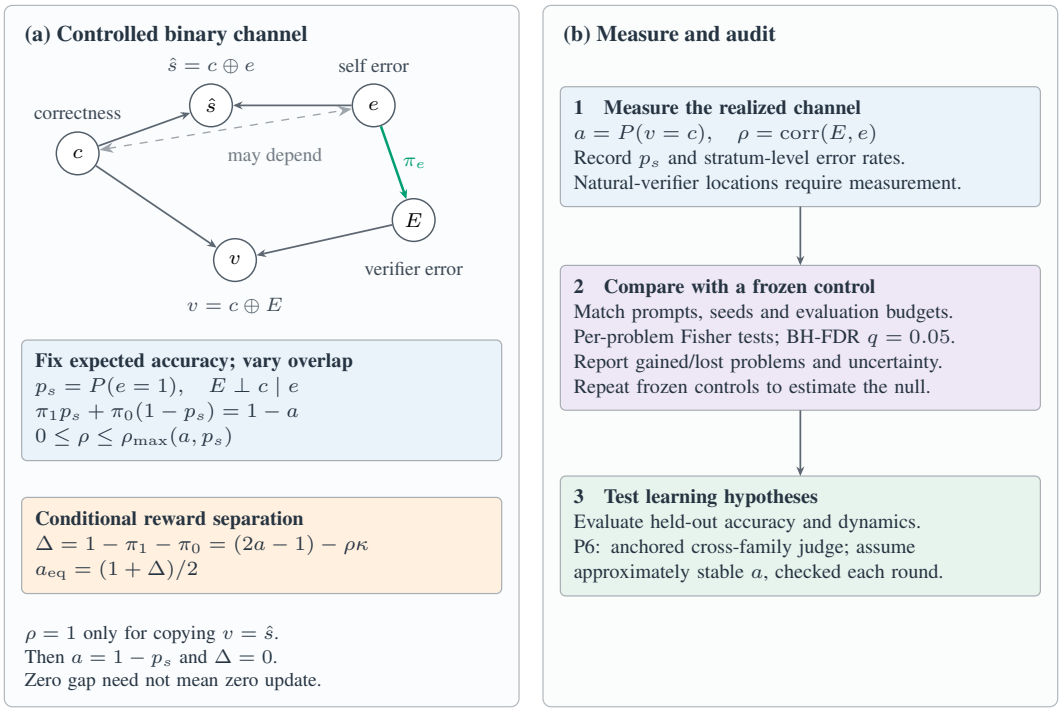

Figure 2: Core verification framework. (a) The controlled channel fixes expected accuracy while varying error overlap with the designated self-judge. (b) Measurement and the proposed frozen-control audit connect channel diagnostics to empirical training hypotheses. The copying verifier is the specific $\rho = 1$ endpoint; natural-verifier locations require measurement.

Monte Carlo channel checks. Third, we specify empirical training hypotheses and a matched-control protocol that keep reward-channel algebra separate from actual learning outcomes. The complete proofs, additional channel diagnostics, detailed protocol, prompts and compute accounting appear in the appendices.

The central implication is conditional: independence can compensate for lower accuracy inside a calculable region. Outside that region, the more accurate correlated verifier retains the larger conditional reward gap. Whether either ordering predicts GRPO learning requires an additional empirical test, because the optimizer acts on normalized rewards and policy-dependent gradient features.

2 Related work

Self-generated supervision includes reasoning bootstrapping, self-rewarding and self-play fine-tuning (Zelikman et al., 2022; Yuan et al., 2025; Chen et al., 2024). Executable verification and self-evolving task generation provide other feedback mechanisms (Zhao et al., 2025; Huang et al., 2026). Internal feedback also extends beyond binary judgments: TTRL, Intuitor and confidence maximization use distinct training signals (Zuo et al., 2025; Zhao et al., 2026; Prabhudesai et al., 2025). Their relation to the specified self-judge is an empirical quantity.

Sharpening theory, generation–verification gaps and solver–verifier dynamics concern selection and training trajectories (Huang et al., 2024; Song et al., 2025; Sun et al., 2026). Noisy-reward analyses and trained verifiers address imperfect supervision from other perspectives (Cai et al., 2026; Cobbe et al., 2021; Lightman et al., 2023). Our identity characterizes how error-placement interventions change conditional reward separation. Reports on correlated verifier errors and self-judge reward hacking motivate the measurements, while measured-null auditing motivates the frozen control (Xin, 2026; Zhou, 2026a; Xu et al., 2026). Extended discussion of these research directions and scaffold-level recursion is retained in the appendix.

3 A controlled model of verifier error placement

3.1 Setting and assumptions

Let $c \in \{0,1\}$ denote response correctness and $\hat{s} \in \{0,1\}$ a specified self-judge's verdict. Define the self-judge error $e = \mathbf{1}[\hat{s} \neq c]$ and its rate $p_s = P(e = 1)$. A verifier returns a binary reward v, with error $E = \mathbf{1}[v \neq c]$. For nondegenerate marginals,

$$a = P(v = c), \qquad \rho = \frac{P(E = 1, e = 1) - (1 - a)p_s}{\sqrt{a(1 - a)p_s(1 - p_s)}}. \tag{1}$$

Thus ρ measures overlap between verifier and self-judge errors on the same response. It is distinct from within-group error correlation, denoted ρ_w in the appendix. The quantity p_s is a binary judgment error rate, not a probability-calibration measure. The oracle has $a = 1$ and undefined error correlation.

The intervention draws E from e alone, so $E \perp c \mid e$, and returns $v = c \oplus E$. This assumption is a restriction on the controlled channel; natural verifiers can have additional dependence on correctness, difficulty, length or answer format. We do not assume that e is independent of c for the reward-gap derivation. The worked information calculations state that additional population assumption separately.

3.2 Constructing feasible interventions

Write $\pi_1 = P(E = 1 \mid e = 1)$ and $\pi_0 = P(E = 1 \mid e = 0)$. Matching expected accuracy imposes $p_s\pi_1 + (1 - p_s)\pi_0 = 1 - a$. Solving this constraint together with Equation (1) gives

$$\pi_1 = (1 - a) + \rho\sqrt{\frac{a(1 - a)(1 - p_s)}{p_s}}, \qquad \pi_0 = \frac{(1 - a) - p_s\pi_1}{1 - p_s}. \tag{2}$$

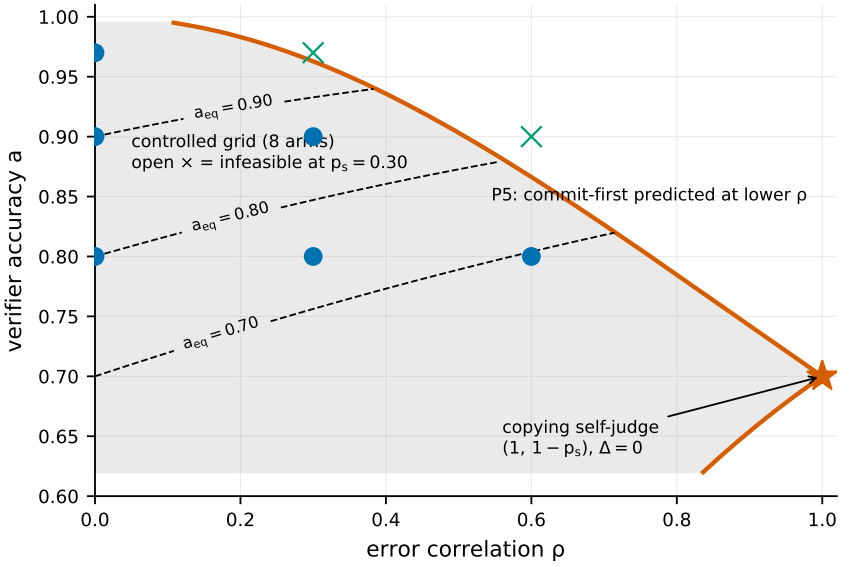

Figure 3: The accuracy–correlation plane at $p_s = 0.30$. On the plotted axes, the copying self-judge occupies $(\rho, a) = (1, 1 - p_s)$; other natural verifiers require measurement. Nominal feasible and infeasible grid targets are distinguished. The a_{eq} contours describe controlled-channel reward gaps, whose relationship to training outcomes is tested by P7.

We study nonnegative correlation. In that range, the rates are valid exactly when

$$0 \le \rho \le \rho_{\max}(a, p_s) = \frac{\min(1-a, p_s) - (1-a)p_s}{\sqrt{a(1-a)p_s(1-p_s)}}. \tag{3}$$

At independence, $\pi_1 = \pi_0 = 1 - a$. At maximal correlation, the smaller error set is nested inside the larger. This need not mean that every self-judge error is copied. The copying verifier $v = \hat{s}$ is the specific endpoint $(a, \rho) = (1 - p_s, 1)$.

The finite reward vector remains random: accuracy is fixed in expectation, not by imposing exactly the same realized number of errors in every trial. Realized accuracy and correlation must be reported. Infeasible targets must be rejected or explicitly redefined; independently clipping the two rates does not preserve the target marginal accuracy. Section C records the distinction between nominal and feasible targets in the training design.

The geometry in Figure 3 also makes the reference judgment explicit. The copying point is determined by the chosen $\hat{s}$ and its measured error rate; changing the self-judge changes the reference plane. Natural-verifier coordinates are measured against the designated self-judge on a common response population, using correctness labels and both sets of verdicts.

3.3 The conditional reward-gap tax

Assume both correctness classes have positive probability in each self-judgment stratum being compared. The separation of interest is

$$\Delta_\sigma = \mathbb{E}[v \mid c = 1, \hat{s} = \sigma] - \mathbb{E}[v \mid c = 0, \hat{s} = \sigma], \qquad \sigma \in \{0, 1\}. \tag{4}$$

If $\hat{s} = 1$, the two means are $1 - \pi_0$ and π_1; if $\hat{s} = 0$, they are $1 - \pi_1$ and π_0. Both strata therefore have the same gap:

$$\Delta = 1 - \pi_1 - \pi_0 = (2a - 1) - \rho\kappa(a, p_s),$$
$$\kappa(a, p_s) = \frac{\sqrt{a(1-a)}(1 - 2p_s)}{\sqrt{p_s(1-p_s)}}. \tag{5}$$

Table 1: Exact channel quantities at $p_s = 0.30$. Each row compares independence with maximal feasible correlation. Equivalent accuracy matches conditional reward gaps only.

a	$\rho_{\max}$	$\Delta(0)$	$\Delta(\rho_{\max})$	$a_{\text{eq}}(\rho_{\max})$	Tax
0.80	0.764	0.600	0.333	0.667	0.133
0.90	0.509	0.800	0.667	0.833	0.067
0.97	0.269	0.940	0.900	0.950	0.020
0.70	1.000	0.400	0.000	0.500	0.200

The identity holds for any correctness prevalence $P(c = 1)$ under the stated conditional-independence assumption.

Proposition 1 (Conditional reward-gap tax)**.** *Fix $0 < a < 1$ and $0 < p_s < 1/2$. Within the feasible controlled family, Δ strictly decreases with ρ at slope $-\kappa(a, p_s)$. The independent verifier with the same conditional gap has accuracy*

$$a_{\text{eq}} = \frac{1 + \Delta}{2} = a - \frac{\rho\kappa(a, p_s)}{2}. \tag{6}$$

On the self-judge error set, the expected reward is

$$\mathbb{E}[v \mid c, \hat{s}] = (1 - \pi_1)c + \pi_1\hat{s}, \qquad \hat{s} \neq c. \tag{7}$$

The full proof and boundary cases are in Section A. Equivalent accuracy matches reward separation only; it is not a language-model benchmark score or a guarantee of equal training performance.

Proof sketch. Within $\hat{s} = 1$, correctness determines $e = 1 - c$, and the mean rewards for $c = 1$ and $c = 0$ are $1 - \pi_0$ and π_1. Within $\hat{s} = 0$, one has $e = c$, giving means $1 - \pi_1$ and π_0. Subtraction in either stratum yields $1 - \pi_1 - \pi_0$. Substituting Equation (2) gives Equation (5), whose correlation coefficient is negative when $p_s < 1/2$. An independent channel has gap $2a' - 1$, so matching gaps gives $a' = a_{\text{eq}}$. On $e = 1$, the identity $\hat{s} = 1 - c$ gives Equation (7) directly.

The conditions matter at the boundaries. At $p_s = 1/2$, the correlation coefficient in Equation (5) vanishes; above one half its sign reverses. At a degenerate marginal, correlation is undefined. If a correctness class is absent from a self-judgment stratum, the corresponding conditional gap is also undefined. These undefined quantities should be reported as such.

3.4 What the gap says about learning

The gap provides a diagnostic of the reward available to a group-relative update. A shared group baseline cancels in a within-group reward difference. Division by a random group standard deviation, however, prevents identifying Δ with an expected normalized advantage difference. Policy-gradient features, clipping and distribution change further intervene between the reward and the final policy (Shao et al., 2024).

For the copying verifier $v = \hat{s}$, $E = e$, $\rho = 1$ and $\Delta = 0$, but rewards can still differ across self-judgment strata. Learning can still occur through differences across strata. Majority vote, a different self-judge prompt and a cross-family judge need not equal the specified $\hat{s}$; their coordinates are empirical quantities. The training hypotheses in Section 5 test this connection.

4 Analytical results and channel checks

4.1 Matched accuracy, different reward separation

At $p_s = 0.30$, increasing error overlap lowers the gap at each fixed verifier accuracy (Figure 4). The maximal-correlation gaps are 0.333, 0.667 and 0.900 for accuracies 0.80, 0.90 and 0.97. Their equivalent independent accuracies are 0.667, 0.833 and 0.950 (Table 1). The independent and maximally correlated 90% channels differ by 0.133 in conditional reward gap.

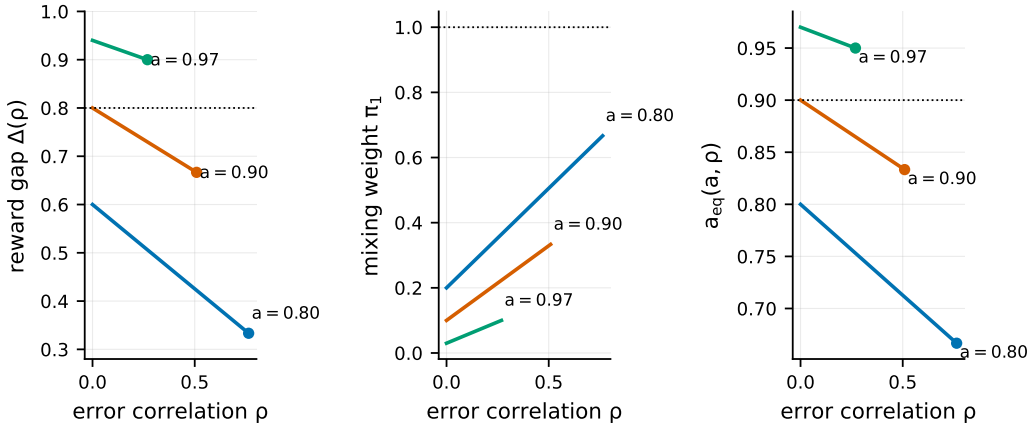

Figure 4: Exact controlled-channel quantities at $p_s = 0.30$: conditional reward gap (left), mixing weight on the specified belief on its error set (middle), and equivalent independent accuracy (right). Dots mark feasible endpoints. The dotted comparator is the independent 90% channel.

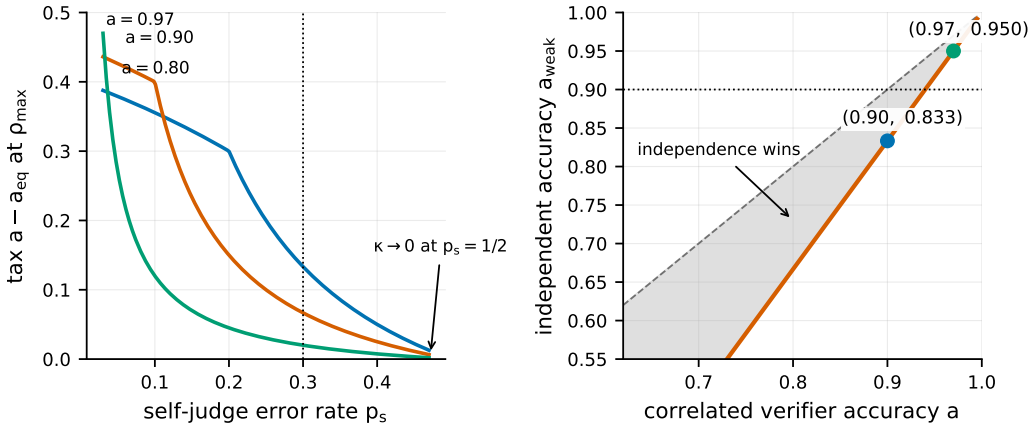

Figure 5: Maximum feasible tax and the independence-winning region. Left: the tax stays bounded as $p_s \to 0$. Right: at $p_s = 0.30$, the shaded band satisfies $a_{\text{eq}} < a_{\text{weak}} < a$, above the orange boundary and below the equal-accuracy diagonal.

4.2 Independence wins only within a region

An independent verifier of accuracy a_{weak} has a larger conditional gap than a correlated verifier (a, ρ) exactly when $a_{\text{weak}} > a_{\text{eq}}(a, \rho)$. For the independent verifier also to be weaker marginally, the region is

$$a_{\text{eq}}(a, \rho) < a_{\text{weak}} < a. \tag{8}$$

Thus the winning band lies above the equivalent-accuracy curve and below the equal-accuracy diagonal (Figure 5). At $p_s = 0.30$, an independent 85% verifier has a larger gap than a maximally correlated 90% verifier; an independent 90% verifier does not exceed the gap of a maximally correlated 97% verifier.

4.3 A bounded tax and a feasible sensitivity example

Writing $q = 1 - a$, the maximal tax for $p_s < 1/2$ is

$$a - a_{\text{eq}}(a, \rho_{\max}) = \begin{cases} q(1 - 2p_s)/(2p_s), & q \le p_s, \\ a(1 - 2p_s)/[2(1 - p_s)], & q \ge p_s. \end{cases} \tag{9}$$

Table 2: Maximum feasible correlation and maximum correlation tax, evaluated at representative self-judge error rates. All entries are exact to the printed precision.

	$\rho_{\max}$			Tax at $\rho_{\max}$		
p_s	$a = 0.80$	$a = 0.90$	$a = 0.97$	$a = 0.80$	$a = 0.90$	$a = 0.97$
0.05	0.459	0.688	0.767	0.379	0.426	0.270
0.10	0.667	1.000	0.528	0.356	0.400	0.120
0.15	0.840	0.793	0.419	0.329	0.233	0.070
0.20	1.000	0.667	0.352	0.300	0.150	0.045
0.25	0.866	0.577	0.305	0.200	0.100	0.030
0.30	0.764	0.509	0.269	0.133	0.067	0.020
0.35	0.681	0.454	0.240	0.086	0.043	0.013
0.40	0.612	0.408	0.215	0.050	0.025	0.008
0.45	0.553	0.369	0.194	0.022	0.011	0.003

Table 3: Monte-Carlo channel check: realized accuracy and correlation, and exact versus plug-in conditional mutual information. Each row uses 4×10^5 synthetic responses; information is in bits. The population has $P(c = 1) = 0.55$, $p_s = 0.30$, and $e \perp c$.

Target a	Target ρ	Realized a	Realized ρ	I_{exact}	$I_{\text{plug-in}}$
0.80	0.00	0.7998	-0.0014	0.2342	0.2343
0.80	0.30	0.8001	$+0.2981$	0.1784	0.1792
0.80	0.60	0.7997	$+0.5992$	0.1581	0.1587
0.90	0.00	0.8998	$+0.0005$	0.4522	0.4520
0.90	0.30	0.9002	$+0.2975$	0.4092	0.4108
0.90	0.50	0.9005	$+0.4986$	0.4359	0.4394
0.97	0.00	0.9698	$+0.0004$	0.6958	0.6956
0.97	0.20	0.9697	$+0.2017$	0.6829	0.6827

Although κ diverges as $p_s \to 0$, the feasible correlation interval shrinks. For fixed $0 < a < 1$, the tax tends to $a/2$, and it vanishes as $p_s \to 1/2$. A maximally correlated 97% verifier ties an independent 90% verifier at $p_s = 0.15$; below this threshold the independent comparator has the larger gap.

Sensitivity comparisons must stay inside the feasible region for every population. At $a = 0.90$, κ grows from 0.122 when $p_s = 0.40$ to 0.800 when $p_s = 0.10$. At the common feasible correlation $\rho = 0.3$, the gaps are 0.763 and 0.560.

4.4 Monte Carlo construction checks

The reported channel validation uses $P(c = 1) = 0.55$, $p_s = 0.30$, independent (c, e) and 4×10^5 synthetic responses per tested point. Across the eight feasible targets in Table 3, realized accuracy differs from target by less than 10^{-3} and realized error correlation by less than 3×10^{-3}. Plug-in conditional mutual information differs from the corresponding exact eight-cell calculation by less than 4×10^{-3} bits. These checks validate the construction on synthetic records. Table 2 gives the wider sensitivity grid; estimator details are retained in Section B.

4.5 Information and reward separation describe different properties

For the worked independent-(c, e) population at $a = 0.90$ and $p_s = 0.30$, $I(v; c \mid \hat{s})$ decreases from approximately 0.452 bits at independence to 0.409 near $\rho = 0.33$, then rises to 0.443 at maximal feasible correlation. The reward gap remains linear and decreasing. Mutual information is convex in the channel matrix at a fixed input distribution, allowing this interior minimum. Figure 6 shows the full curves; the derivation is in Section B.

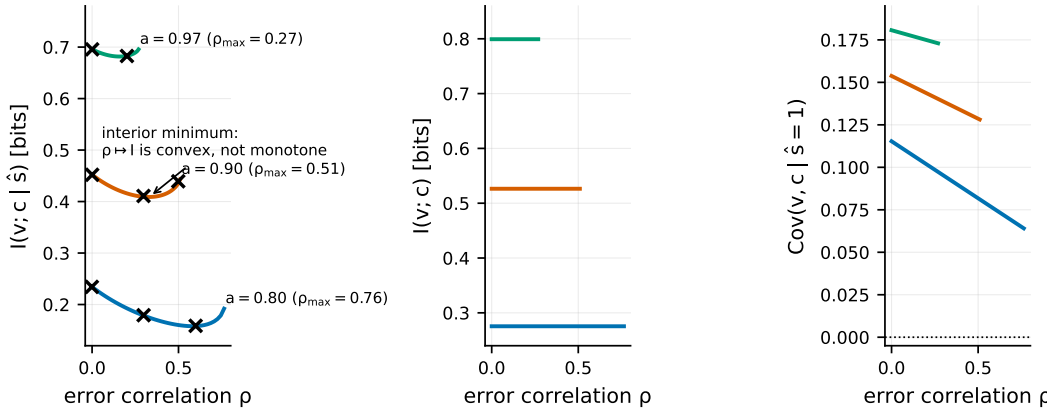

Figure 6: Exact information diagnostics at $P(c = 1) = 0.55$, $p_s = 0.30$, and $e \perp c$. Left: conditional mutual information is convex and can have an interior minimum as correlation increases. Crosses show Monte-Carlo estimates from 4×10^5 responses, agreeing with the exact curves to less than 4×10^{-3} bits. Middle: unconditional mutual information is constant in correlation under the worked-population assumptions. Right: reward–correctness covariance within $\hat{s} = 1$ decreases.

5 Testing the connection to self-improvement

Figure 1 summarizes the proposed training and evaluation loop. The proposed experiment tests whether channel-level reward separation predicts actual learning. The design uses five rounds of GRPO–LoRA on Qwen2.5-1.5B-Instruct, with Qwen3.5-2B-Base as the named replicate, 256 prompts per round and eight responses per prompt. Ground-truth answers are withheld from policy prompts but used by the controlled channels, oracle and evaluation. The controlled intervention therefore requires labels. Full model, data, update and decoding settings are retained in Section C.

The reference arms are a frozen policy, an oracle, majority vote, anchored and commit-first self-judges, and a cross-family judge in both forms. Controlled arms vary error placement at matched expected accuracy subject to Equation (3). Only round-0 responses can be identical across on-policy arms. Subsequent policies generate their own responses; prompts and evaluation budgets remain matched, while the comparison includes the effects of different training histories.

The central empirical prediction is that, at comparable verifier accuracy, larger conditional reward gaps tend to produce greater improvement over the frozen control. Table 4 summarizes the hypotheses concerning collapse timing, pass@k, commit-first judging and the a_{eq} ordering. Flat or reversed training outcomes would limit the gap's predictive value.

The drift hypothesis, P6, concerns the *fixed anchored cross-family verifier*, using its commit-first counterpart as a control. We assume that the fixed verifier's accuracy remains approximately stable over the period studied and verify this assumption each round. The designated anchored self-judge is excluded from this drift test: against its own judgment it has $\rho \equiv 1$ in every nondegenerate round. Actual $(a_t, \rho_t, p_{s,t})$ must be reported.

Evaluation compares each arm with matched frozen-control evaluations, using per-problem outcome counts and explicit multiple-testing families (Xu et al., 2026). A problem with no sampled correct answer is recorded as unsolved in that finite evaluation sample. Detailed hypotheses, prompts, sampling budgets, the audit and corrected token accounting are in Sections C and F.

Interpreting the empirical test. The controlled arms isolate verifier-error placement; natural arms test how well the intervention captures real judgments. A natural judge can violate the channel assumptions at coordinates matching a controlled arm. Disagreement in learning outcomes can expose limits of the coordinates or their connection to normalized optimization. Reporting realized parameters, conditional error rates and frozen-control uncertainty helps distinguish these explanations.

Table 4: Empirical training hypotheses and their analytical benchmarks. Numerical slopes and differences refer to conditional reward gaps.

	Empirical training hypothesis	Analytical scope
P1	Final gain decreases with ρ at fixed a.	$-\kappa$ is the reward-gap slope only.
P2	Collapse occurs earlier at higher correlation.	No exact collapse-time prediction.
P3	pass@32 near $\rho_{\max}$ does not exceed the frozen control's.	No pass@k guarantee follows from the gap.
P4a	$(0.90, 0)$ outperforms $(0.90, \rho_{\max})$.	At $p_s = 0.30$, their reward gaps differ by 0.133.
P4b	$(0.97, \rho_{\max})$ outperforms $(0.90, 0)$ when $a_{\text{eq}} > 0.90$.	Gap comparison: $0.900 > 0.800$ at $p_s = 0.30$; tie at 0.15.
P5	Commit-first judging lowers ρ at similar a and improves more.	Natural-judge coordinates and gains require measurement.
P6	Anchored cross-family error correlation rises across rounds.	Fixed-verifier accuracy stability is assumed and checked each round.
P7	Gains follow a_{eq} when its ordering differs from a.	The a_{eq} contours describe conditional reward gaps.

6 Discussion and limitations

The controlled construction shows that error placement can change conditional reward discrimination at fixed expected accuracy. Outside $E \perp c \mid e$, verifier error rates can differ across correctness classes within the same self-judge-error stratum, so the controlled-gap identity need not apply. Stratum-level rates should accompany aggregate coordinates when assessing natural judges.

The construction uses ground-truth labels for a diagnostic intervention. The proposed empirical study is limited to small models and mathematical reasoning. Training experiments are needed to quantify variability from learned self-judgments and changing response populations; longer horizons, other scales and other domains require further study. The stability of a fixed verifier's accuracy is an explicit empirical assumption, to be checked each round.

Finally, the role of group normalization must remain separate from the role of error overlap. A reduced conditional reward gap is not automatically a reduced normalized advantage or a smaller beneficial gradient. Likewise, within-group error correlation determines an effective sample size for a mean error estimator only under specified variance and correlation assumptions; it does not directly determine GRPO reward or gradient variance.

7 Conclusion

Verifier accuracy and error correlation with a specified self-judge capture different properties of feedback. Within a controlled binary channel, we derive an exact conditional reward-gap tax, equivalent independent accuracy, feasible winning region and bounded sensitivity to self-judge error. The copying verifier has zero within-stratum reward gap; other natural judges require measurement. Channel checks support the numerical construction, while the proposed training protocol tests its predictive relevance for self-improvement. We recommend reporting verifier accuracy, measured error correlation and training outcomes together.

AI USE STATEMENT

AI tools were used for language polishing.

REPRODUCIBILITY STATEMENT

The controlled channel is specified in Section 3; proofs, probability assumptions and boundary cases are provided in Section A. Exact sensitivity values and reported Monte Carlo checks appear in Tables 2 and 3, with estimator details in Section B. The unexecuted training protocol, prompts, evaluation audit and compute accounting are retained in Sections C and F. The accompanying LaTeX sources use the unmodified official ICLR 2027 submission style.

REFERENCES

Luke Bailey, Kaiyue Wen, Kefan Dong, Tatsunori Hashimoto, and Tengyu Ma. Scaling self-play with self-guidance, 2026. URL https://arxiv.org/abs/2604.20209.

Xin-Qiang Cai, Wei Wang, Feng Liu, Tongliang Liu, Gang Niu, and Masashi Sugiyama. Reinforcement learning with verifiable yet noisy rewards under imperfect verifiers, 2026. URL https://arxiv.org/abs/2510.00915.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Xinjie Chen, Biao Fu, Jing Wu, Guoxin Chen, Xinggao Liu, Dayiheng Liu, and Minpeng Liao. Jury-rl: Votes propose, proofs dispose for label-free rlvr, 2026. URL https://arxiv.org/abs/2604.25419.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models, 2024. URL https://arxiv.org/abs/2401.01335.

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

Yi Duan, Ying Liu, Zirui Tang, Haodong Chen, Jun Zhou, Yumou Liu, Bangrui Xu, Yukai Wu, Sidi Chen, Yuhan Zhou, Haoyu Wang, Xiaoyou Yu, Shaokun Han, Xuzhou Zhu, Le Zhou, Bolin Lu, Wei Zhou, Jiachen Liu, Nuozhou Fang, Jiaxin Tian, Ruoyu Chen, Yuxuan Li, Kai Zuo, Kaiyan Zhang, Jiantao Qiu, Conghui He, Guoliang Li, Bowen Zhou, Zhiyuan Liu, Zhoufutu Wen, Jihua Kang, Xuanhe Zhou, and Fan Wu. The last ai built by humans: Toward genuine recursive self-improvement, 2026. URL https://arxiv.org/abs/2609.11873.

Mingyuan Fan, Weiguang Han, Daixin Wang, Cen Chen, Zhiqiang Zhang, and Jun Zhou. When sharpening becomes collapse: Sampling bias and semantic coupling in rl with verifiable rewards, 2026. URL https://arxiv.org/abs/2601.15609.

Shi Feng, Hanlin Zhang, Fan Nie, Sham Kakade, and Yiling Chen. Peer-predictive self-training for language model reasoning, 2026. URL https://arxiv.org/abs/2604.13356.

Yonatan Gideoni, Sebastian Risi, and Yarin Gal. Simple baselines are competitive with code evolution, 2026. URL https://arxiv.org/abs/2602.16805.

Jiangang Han. Partially correlated verifier cascades in llm harnesses: Concave log-odds, polynomial reliability, and blind-spot ceilings, 2026. URL https://arxiv.org/abs/2607.13918.

Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, Xiusi Chen, Youbang Sun, Xingtai Lv, Xuekai Zhu, Li Sheng, Ran Li, Huan ang Gao, Yuchen Zhang, Bowen Zhou, Zhiyuan Liu, and Ning Ding. How far can unsupervised rlvr scale llm training?, 2026. URL https://arxiv.org/abs/2603.08660.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

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

Audrey Huang, Adam Block, Dylan J. Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T. Ash, and Akshay Krishnamurthy. Self-improvement in language models: The sharpening mechanism, 2024. URL https://arxiv.org/abs/2412.01951.

Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-zero: Self-evolving reasoning llm from zero data, 2026. URL https://arxiv.org/abs/2508.05004.

Jakub Grudzien Kuba, Mengting Gu, Qi Ma, Yuandong Tian, Vijai Mohan, and Jason Chen. Language self-play for data-free training, 2025. URL https://arxiv.org/abs/2509.07414.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step, 2023. URL https://arxiv.org/abs/2305.20050.

Jianzhe Lin. Self-improvement can self-regress: The rise-and-collapse failure mode of llm self-training, 2026. URL https://arxiv.org/abs/2606.21090.

Bo Liu, Chuanyang Jin, Seungone Kim, Weizhe Yuan, Wenting Zhao, Ilia Kulikov, Xian Li, Sainbayar Sukhbaatar, Jack Lanchantin, and Jason Weston. Spice: Self-play in corpus environments improves reasoning, 2025. URL https://arxiv.org/abs/2510.24684.

Bo Liu, Leon Guertler, Simon Yu, Zichen Liu, Penghui Qi, Daniel Balcells, Mickel Liu, Cheston Tan, Weiyan Shi, Min Lin, Wee Sun Lee, and Natasha Jaques. Spiral: Self-play on zero-sum games incentivizes reasoning via multi-agent multi-turn reinforcement learning, 2026a. URL https://arxiv.org/abs/2506.24119.

Chenruo Liu, Yijun Dong, Yiqiu Shen, and Qi Lei. A task-centric theory for iterative self-improvement with easy-to-hard curricula, 2026b. URL https://arxiv.org/abs/2602.10014.

Ambuj Mehrish and Sebastiano Vascon. Core: Consensus rewards via equilibrium for test-time reinforcement learning, 2026. URL https://arxiv.org/abs/2608.09324.

Sarthak Mittal, Leo Gagnon, and Guillaume Lajoie. Beyond distribution sharpening: The importance of task rewards, 2026. URL https://arxiv.org/abs/2604.16259.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing confidence alone improves reasoning, 2025. URL https://arxiv.org/abs/2505.22660.

Zhenting Qi, Susanna Maria Baby, Stefanie Anna Baby, Kan Yuan, Andrew Tomkins, Tu Vu, Da-Cheng Juan, and Cyrus Rashtchian. On the generalization gap in self-evolving language model reasoning, 2026. URL https://arxiv.org/abs/2606.01075.

Shuvendu Roy, Hossein Hajimirsadeghi, Mengyao Zhai, and Golnoosh Samei. You need reasoning to learn reasoning: The limitations of label-free rl in weak base models, 2025. URL https://arxiv.org/abs/2511.04902.

Sheikh Shafayat, Fahim Tajwar, Ruslan Salakhutdinov, Jeff Schneider, and Andrea Zanette. Can large reasoning models self-train?, 2025. URL https://arxiv.org/abs/2505.21444.

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, 2024. URL https://arxiv.org/abs/2402.03300.

Yuda Song, Hanlin Zhang, Carson Eisenach, Sham Kakade, Dean Foster, and Udaya Ghai. Mind the gap: Examining the self-improvement capabilities of large language models, 2025. URL https://arxiv.org/abs/2412.02674.

Yifan Sun, Yushan Liang, Zhen Zhang, Xin Liu, and Jiaye Teng. Theoretical modeling of large language model self-improvement training dynamics through solver-verifier gap, 2026. URL https://arxiv.org/abs/2507.00075.

Shobhita Sundaram, John Quan, Ariel Kwiatkowski, Kartik Ahuja, Yann Ollivier, and Julia Kempe. Teaching models to teach themselves: Reasoning at the edge of learnability, 2026. URL https://arxiv.org/abs/2601.18778.

Zihan Tan, Leixin Sun, Zitong Shi, Yitao Liu, Jiajun Wu, Nathaniel Brooks, Jiaru Qian, Xiaoran Shang, Suyuan Huang, Yi Ding, Yangxu Liao, Mukai Li, Qiushi Sun, Shudong Liu, Xuankun Rong, Xiaohang Yu, Zhuo Chen, Hejia Geng, Chenxin Li, Aozhou Wang, Zengji Tu, Robert Tang, Yuxin Zhan, Eric Jiang, Yuxin Wu, Jianqing Zhang, Xiao Liang, Fang Wu, Haochi Zhang, Alexander Marlow, and Guancheng Wan. Metarsi / rsi2: A meta-recursive self-improving system for recursive self-improving systems themselves, 2026. URL https://arxiv.org/abs/2609.06396.

Wenyi Wang, Piotr Piękos, Li Nanbo, Firas Laakom, Yimeng Chen, Mateusz Ostaszewski, Mingchen Zhuge, and Jürgen Schmidhuber. Huxley-gödel machine: Human-level coding agent development by an approximation of the optimal self-improving machine, 2025. URL https://arxiv.org/abs/2510.21614.

Chen Henry Wu and Aditi Raghunathan. Self-trained verification for training- and test-time self-improvement, 2026. URL https://arxiv.org/abs/2605.30290.

Peng Xia, Kaide Zeng, Jiaqi Liu, Can Qin, Fang Wu, Yiyang Zhou, Caiming Xiong, and Huaxiu Yao. Agent0: Unleashing self-evolving agents from zero data via tool-integrated reasoning, 2025. URL https://arxiv.org/abs/2511.16043.

Esther Xin. Are verifier errors independent within a grpo group? evidence from qwen2.5 rollouts, 2026. URL https://arxiv.org/abs/2609.06386.

Cheng Xu, Nan Yan, Liming Chen, and M-Tahar Kechadi. Phantom gains: Auditing self-improvement against a measured null, 2026. URL https://arxiv.org/abs/2608.20290.

Haoyan Yang, Mario Xerri, Solha Park, Huajian Zhang, Yiyang Feng, Sai Akhil Kogilathota, and Jiawei Zhou. Self-improvement of large language models: A technical overview and future outlook, 2026. URL https://arxiv.org/abs/2603.25681.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2025. URL https://arxiv.org/abs/2401.10020.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022. URL https://arxiv.org/abs/2203.14465.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin godel machine: Open-ended evolution of self-improving agents, 2026a. URL https://arxiv.org/abs/2505.22954.

Jenny Zhang, Bingchen Zhao, Wannan Yang, Jakob Foerster, Jeff Clune, Minqi Jiang, Sam Devlin, and Tatiana Shavrina. Hyperagents, 2026b. URL https://arxiv.org/abs/2603.19461.

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, 2025. URL https://arxiv.org/abs/2505.03335.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards, 2026. URL https://arxiv.org/abs/2505.19590.

Chenyu Zhou. More convincing, not more correct: Self-play reward hacking of reference-free llm judges, 2026a. URL https://arxiv.org/abs/2607.05904.

Todd Zhou. When rlvr shrinks the reasoning boundary: Diagnosing pass@k inversion, 2026b. URL https://arxiv.org/abs/2607.20543.

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, 2025. URL https://arxiv.org/abs/2504.16084.

A PROOFS AND ANALYTICAL DETAILS

A.1 SETTING AND FEASIBILITY

Let $c \in \{0, 1\}$ denote correctness, $e = \mathbb{1}[\hat{s} \neq c]$ the designated self-judge error, and $E = \mathbb{1}[v \neq c]$ the verifier error. Thus $\hat{s} = c \oplus e$ and $v = c \oplus E$. The controlled channel draws E from e alone, with $P(E = 1 \mid e = 1) = \pi_1$ and $P(E = 1 \mid e = 0) = \pi_0$. In particular, $E \perp c \mid e$. The reward-gap proof does not assume $e \perp c$; that additional assumption is used only for the worked information calculations below.

Write $q = 1 - a$ and $h = P(E = 1, e = 1)$. The Bernoulli marginals require

$$\max(0, q + p_s - 1) \leq h \leq \min(q, p_s). \tag{10}$$

Substituting $h = qp_s + \rho\sqrt{a(1-a)p_s(1-p_s)}$ gives the upper bound in Eq. (3). The nonnegative-correlation restriction $\rho \geq 0$ describes the intervention region studied here, rather than the full mathematical range of a Bernoulli correlation. Within that region, the conditional rates in Eq. (2) are feasible precisely when $\rho \leq \rho_{\max}(a, p_s)$. Accuracy and correlation are population targets; a finite realization is reported with its sampling uncertainty.

A.2 PROOF OF THE CONDITIONAL REWARD-GAP IDENTITY

Fix a self-judgment stratum $\hat{s} = \sigma$ in which both correctness classes have positive probability. If $\sigma = 1$, then $e = 1 - c$, giving

$$\mathbb{E}[v \mid c = 1, \hat{s} = 1] = 1 - \pi_0, \qquad \mathbb{E}[v \mid c = 0, \hat{s} = 1] = \pi_1. \tag{11}$$

If $\sigma = 0$, then $e = c$, giving

$$\mathbb{E}[v \mid c = 1, \hat{s} = 0] = 1 - \pi_1, \qquad \mathbb{E}[v \mid c = 0, \hat{s} = 0] = \pi_0. \tag{12}$$

In both strata, subtraction yields $\Delta = 1 - \pi_1 - \pi_0$. Substitution of Eq. (2) gives

$$\begin{aligned} \Delta &= 1 - \pi_1 - \frac{q - p_s\pi_1}{1 - p_s} = \frac{1 - p_s - q - (1 - 2p_s)\pi_1}{1 - p_s} \\ &= (2a - 1) - \rho\frac{\sqrt{a(1-a)}(1 - 2p_s)}{\sqrt{p_s(1-p_s)}}. \end{aligned} \tag{13}$$

This is Eq. (5). Its slope is strictly negative for $0 < a < 1$ and $0 < p_s < 1/2$. The channel assumption ensures that the flip rates do not change when correctness is specified within an error stratum.

On the disagreement set $e = 1$, we have $\hat{s} = 1 - c$, and therefore

$$\mathbb{E}[v \mid c, e = 1] = (1 - \pi_1)c + \pi_1(1 - c) = (1 - \pi_1)c + \pi_1\hat{s}. \tag{14}$$

As ρ varies from zero to its feasible maximum, the coefficient π_1 increases from q to $\min\{1, q/p_s\}$. This proves the stated mixture interpretation on the self-judge's error set.

An independent verifier with accuracy a' has $\pi_1 = \pi_0 = 1 - a'$ and gap $2a' - 1$. Equating this gap to Eq. (13) yields $a' = a - \rho\kappa/2$, establishing Eq. (6). At the copying point $q = p_s$, $\rho = 1$, one has $\pi_1 = 1$, $\pi_0 = 0$, and $v = \hat{s}$. Hence $\Delta = 0$ and $a_{\text{eq}} = 0.5$, completing the proof of Proposition 1. This endpoint describes the specified copying judge; it does not locate every same-model, majority-vote, or self-certainty verifier.

What the identity does and does not establish. The reward-gap identity does not use $P(c = 1)$ or independence between e and c. It characterizes expected reward separation within a self-judgment stratum. A shared group baseline cancels in a within-group reward difference, but group normalization and policy-gradient features also affect advantage separation and the update. Thus Δ alone determines neither normalized GRPO advantage differences nor final training gains. The channel identity and the empirical hypotheses in Appendix D have different objects of validation.

A.3 THE BOUNDED MAXIMUM TAX AND THE CROSSOVER

At maximum feasible correlation, there are two cases. If $q \le p_s$, then $\pi_1 = q/p_s$ and $\pi_0 = 0$, so $a_{\mathrm{eq}} = 1 - q/(2p_s)$. If $q \ge p_s$, then $\pi_1 = 1$ and $\pi_0 = (q - p_s)/(1 - p_s)$, giving $a_{\mathrm{eq}} = a/[2(1 - p_s)]$. Consequently,

$$a - a_{\mathrm{eq}}(a, \rho_{\max}) = \begin{cases} \dfrac{(1-a)(1-2p_s)}{2p_s}, & 1 - a \le p_s, \\ \dfrac{a(1-2p_s)}{2(1-p_s)}, & 1 - a \ge p_s. \end{cases} \tag{15}$$

The branches agree at $1 - a = p_s$. For fixed $0 < a < 1$, the maximum tax tends to $a/2$ as $p_s \to 0$: the divergent slope is offset by the shrinking feasible correlation interval. The tax vanishes as $p_s \to 1/2$.

For a maximally correlated 97% verifier and an independent 90% comparator, the first branch gives $a_{\mathrm{eq}} = 1 - 0.03/(2p_s)$. The reward gaps tie when $p_s = 0.15$. At $p_s = 0.30$, their gaps are 0.900 and 0.800, respectively; at $p_s = 0.10$, the maximally correlated 97% channel has $a_{\mathrm{eq}} = 0.85$. These exact comparisons motivate training hypotheses without determining their outcomes. A weaker independent verifier has a larger conditional gap precisely in the band $a_{\mathrm{eq}}(a, \rho_{\max}) < a_{\mathrm{weak}} < a$.

B INFORMATION DIAGNOSTICS AND CHANNEL VALIDATION

B.1 CONDITIONAL MUTUAL INFORMATION IS CONVEX, NOT MONOTONE

Let $w = e \oplus E$. Within $\hat{s} = \sigma$, bijective relabelling gives

$$I(v; c \mid \hat{s} = \sigma) = I(w; e \mid \hat{s} = \sigma). \tag{16}$$

The channel satisfies $P(w = 1 \mid e = 0) = \pi_0$ and $P(w = 1 \mid e = 1) = 1 - \pi_1$. For a fixed input law, mutual information is convex in the channel matrix. Along an iso-accuracy segment, the matrix varies affinely with ρ, so conditional mutual information is convex in ρ and can have an interior minimum.

The worked eight-cell enumeration uses $P(c = 1) = 0.55$, $p_s = 0.30$, and $e \perp c$. At $a = 0.90$, $I(v; c \mid \hat{s})$ is approximately 0.452 bits at $\rho = 0$, 0.409 near the interior minimum $\rho = 0.33$, and 0.443 at $\rho_{\max} \simeq 0.51$. As $\pi_0 \to 0$, the verdict carries additional information about correctness. This behavior does not establish whether or how GRPO exploits that information. Conditional mutual information and conditional reward separation are distinct diagnostics.

Under the worked-example independence assumption and fixed $P(c = 1)$, marginalizing over e produces a correctness-independent flip probability $1 - a$, making $I(v; c)$ constant in ρ at fixed a. Accuracy alone does not imply this constancy for arbitrary joint populations. A verdict that is a deterministic function of the specified $\hat{s}$ has $I(v; c \mid \hat{s}) = 0$; other same-model or majority-vote judgments need not be such functions.

Information-figure interpretation. Figure 6 shows three diagnostics at the same worked population: conditional information with an interior minimum, constant unconditional information under the stated independence assumption, and declining reward–correctness covariance among responses with $\hat{s} = 1$. Its Monte-Carlo crosses use 4×10^5 responses and agree with the exact information curves to less than 4×10^{-3} bits.

B.2 MONTE-CARLO CHANNEL CHECK

Table 3 reports eight numerical channel checks. Each check uses 4×10^5 synthetic responses with $P(c = 1) = 0.55$ and $p_s = 0.30$. The reported errors in realized accuracy and correlation are at sampling-noise scale. These checks validate the channel construction and information estimator; they are not language-model training outcomes.

The implementation derives the channel's random stream from an independent seed sequence. This avoids a development-time seed collision in which a caller and the channel reused the same random stream, placing verifier errors only on correct responses. A regression test covers this property. The

implementation also includes 19 pure-NumPy tests spanning answer extraction, channel targets on 4×10^4 responses, information estimation, feasibility flags, reward assignment, GRPO advantages, and a synthetic audit tree with oracle and frozen arms. These implementation checks are separate from the proposed five-round training study.

C Detailed language-model training protocol

C.1 Policies, data, and updates

The protocol studies Qwen2.5-1.5B-Instruct as the primary policy and Qwen3.5-2B-Base as the replicate. Each policy uses a rank-32 LoRA adapter on attention and MLP projections, with scaling $\alpha = 64$ and no dropout; the base weights remain frozen (Hu et al., 2021). No larger policy is included. The protocol specifies five rounds, with $P = 256$ prompts per round, $K = 8$ responses per prompt, temperature 1.0, and a maximum of 768 generated tokens per response. Each round performs one clipped GRPO epoch with learning rate 10^{-5}, micro-batch size 8, clipping coefficient 0.2, length-normalized completions, and no reference-KL term (Shao et al., 2024). Groups with zero reward variance contribute no gradient.

The prompt pool is the GSM8K training split (Cobbe et al., 2021). Ground-truth answers are withheld from policy prompts, but are used to construct the oracle and controlled rewards, measure verifier accuracy and correlation, and evaluate responses. The controlled intervention is therefore a ground-truth-based diagnostic. Each round selects prompts with a deterministic per-seed, per-round shuffle so every arm receives identical prompt identities. The three seeds are 0, 1, and 2; they control the shuffle, generation, and channel randomization.

Round-0 responses are generated once from the base policy and shared across arms. Starting in round 1, each arm samples from its own updated adapter. The later-round comparison therefore includes the effects of different training histories and response distributions, while prompt identities, decoding settings, update settings, and evaluation budgets remain matched. It is not a reward-only intervention on identical later-round responses.

Held-out evaluation covers 500 GSM8K test problems, the MATH-500 evaluation with numeric-answer extraction, and 30 AIME-2024 problems as an out-of-distribution probe (Hendrycks et al., 2021; Lightman et al., 2023). The nominal compute budget uses $1{,}030$ evaluation problems. Evaluation records greedy accuracy and $n = 32$ sampled responses per problem at temperature 0.8, using the unbiased pass@k estimator for $k \leq 32$ (Chen et al., 2021). It also records the mean number of distinct final answers per problem and the number of problems unobserved as solved in the 32 sampled responses.

C.2 Verifier arms and target measurement

The specified design has 15 arms per seed: a frozen control, an oracle, majority vote, anchored and commit-first self-judges, anchored and commit-first cross-family judges, and eight nominal controlled channels. The cross-family judge is Gemma-4-E2B-it. The nominal controlled grid is $\{0.80, 0.90, 0.97\} \times \{0, 0.3, 0.6\}$ with the $(0.97, 0.6)$ corner omitted. Conditional rates are recomputed from the realized self-judge error rate each round. Nominal targets must be adjusted or rejected when infeasible; plotted nominal targets and feasible realized coordinates are distinguished. The empirical record reports the realized values rather than assuming that a nominal target was attained.

Controlled channels are defined against the designated anchored self-judge. Error correlation against commit-first self-judgment is recorded as a second operationalization. Only the arm whose verdict identically copies the designated $\hat{s}$ has $E = e$ and $\rho = 1$ by construction in the nondegenerate case. Majority vote, other self-judge prompts, and cross-family judgments are placed on the plane by measurement. The oracle has $a = 1$ and undefined error correlation because its error indicator has zero variance.

Per arm and round, measurements include realized accuracy with a Wilson interval, correlation with a bootstrap interval, plug-in $I(v; c \mid \hat{s})$, reward rate, and pseudo-label accuracy. Natural-verifier outcomes are compared with a response surface fitted to the controlled grid. Agreement is

an empirical hypothesis. Correctness-, difficulty-, format-, or length-dependent error structure can violate the controlled-channel assumptions even when two natural verifiers have the same measured coordinates.

C.3 Prompts and verdict extraction

The policy prompt asks for step-by-step reasoning and a final numeric answer in `\boxed{ANSWER}`. The anchored self-judge sees the problem, the last 1,500 characters of the candidate solution, and its extracted final answer, and is asked to return exactly one word, *yes* or *no* (Yuan et al., 2025). The commit-first judge sees only the problem and solves it before the candidate is revealed; the verdict is the programmatic agreement between its extracted committed answer and the candidate's answer (Zhou, 2026a). The cross-family judge uses the same two prompt forms. Majority vote compares a candidate answer with the modal answer among the K responses (Zuo et al., 2025). The exact prompt strings are retained in the accompanying implementation.

C.4 Audit against a measured frozen-control null

Following the measured-null protocol of Xu et al. (2026), the frozen control is evaluated once per seed and round with the same sampler and decoding configuration. Replicates are pooled per problem. For each arm, seed, round, and evaluation set, the audit compares the arm's success and failure counts with pooled frozen-control counts using Fisher's exact test in both directions and the specified Benjamini–Hochberg procedure at $q = 0.05$. It reports the number of problems identified as gained and lost rather than comparing an observed transition with an assumed zero.

The held-out set is partitioned according to whether the pooled frozen control solves each problem at least once. Operational sharpening is the accuracy gain on problems observed as solved by the control. Operational expansion is the fraction of problems unobserved as solved by the pooled control that an arm solves at least once in $n = 32$ samples. Finite-sample non-observation is not proof of intrinsic unreachability. Because a frozen model can appear to expand relative to a finite sample of itself, the protocol also computes the expansion statistic between disjoint halves of the frozen replicates as a measured null.

The protocol records the collapse step as the first round in which greedy accuracy falls at least 10% below its running maximum, together with per-round correlation drift, pass@k relative to the frozen control, and answer diversity. These measurements distinguish hypothesized learning trends from the exact channel calculations.

D Empirical hypotheses and their analytical benchmarks

The project hypothesis log records the original predictions before training outcomes were inspected. P4 was split after the channel calculation showed that the initial folk ordering, an independent 90% verifier outperforming a maximally correlated 97% verifier, is not implied at the expected self-judge error rates. The seven hypotheses, with P4 split into two comparisons, are stated below. No final-gain slope, collapse step, pass@k ranking, or natural-verifier learning outcome is proved by Proposition 1.

- **P1.** At fixed verifier accuracy, final-round gain over the frozen control decreases as error correlation increases. The exact slope $-\kappa$ applies only to conditional reward gap, not to final-round gain.
- **P2.** The collapse step occurs earlier at higher correlation. This is a dynamical hypothesis, with no exact collapse-time prediction.
- **P3.** Near maximum feasible correlation, pass@32 does not exceed the frozen control's. An observed improvement would challenge this hypothesis (Zhou, 2026b).
- **P4a.** The $(0.90, 0)$ channel outperforms $(0.90, \rho_{\max})$. At $p_s = 0.30$, their conditional reward gaps differ by 0.133; this is not an accuracy-gain effect size.
- **P4b.** The $(0.97, \rho_{\max})$ channel outperforms $(0.90, 0)$ when its equivalent accuracy exceeds 0.90. The exact gap comparison at $p_s = 0.30$ is $0.900 > 0.800$, tied at $p_s = 0.15$ and reversed at $p_s = 0.10$. Training outcomes need not inherit the algebraic ordering.

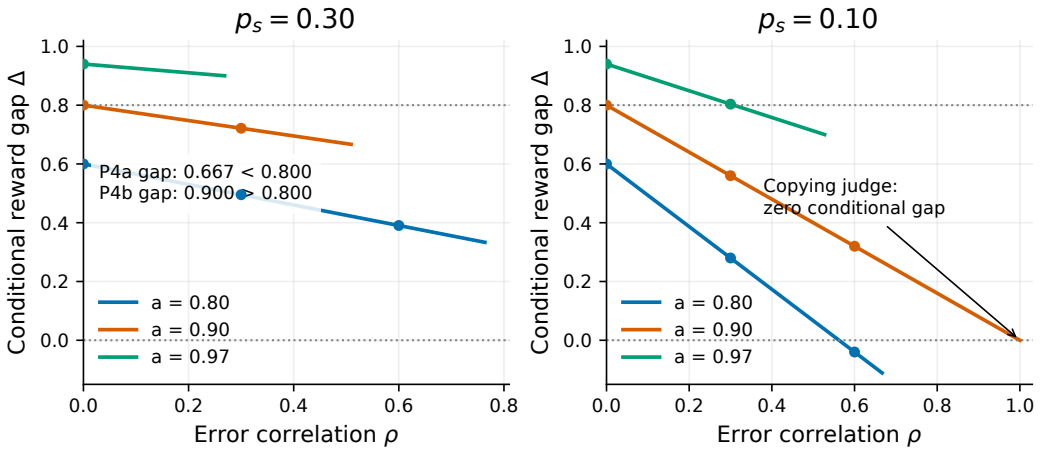

Figure 7: Exact conditional reward-gap curves for the controlled family at $p_s = 0.30$ and $p_s = 0.10$. Curve endpoints mark feasibility limits. The P4 annotations refer to reward-gap benchmarks; they are not guarantees about the corresponding training hypotheses.

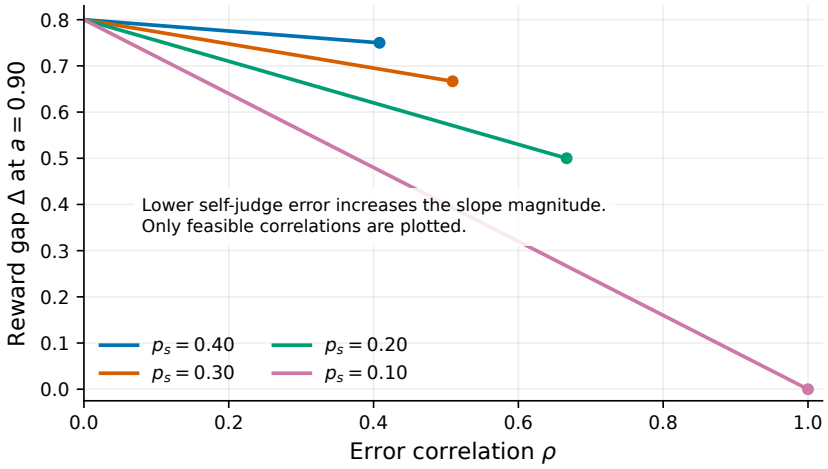

Figure 8: Conditional reward-gap sensitivity to self-judge error rate at fixed verifier accuracy $a = 0.90$. Endpoints mark $\rho_{\max}(0.90, p_s)$. P6 tests the anchored cross-family judge against its commit-first counterpart, with verifier accuracy assumed approximately stable and checked each round; the copying self-judge is excluded.

P5. Commit-first self-judging has lower measured correlation at comparable accuracy and yields greater training gains than anchored self-judging (Zhou, 2026a). The direction and its relation to gain require measurement.

P6. The fixed anchored cross-family judge's measured correlation rises across rounds, with its commit-first counterpart as a control. The designated anchored self-judge is excluded: its correlation with its own error indicator remains one in every nondegenerate round. We assume that the fixed cross-family verifier's accuracy remains approximately stable over the period studied and verify this assumption each round. Fixing the judge alone does not guarantee accuracy stability under a changing response distribution.

P7. When the accuracy and a_{eq} orderings disagree, training outcomes follow a_{eq}. Its contours are exact reward-gap contours for the controlled family, while their predictive value for learned-policy outcomes is empirical.

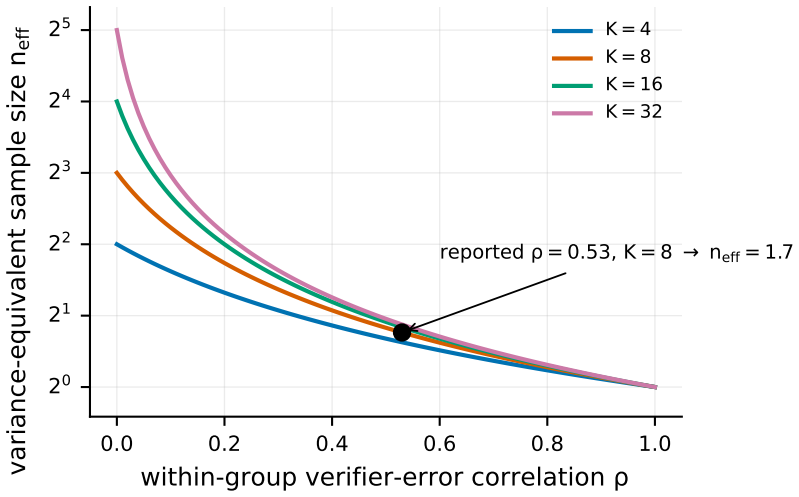

Figure 9: Variance-equivalent sample size for the mean verifier error under equal variances and common pairwise error correlation. The calculation does not establish GRPO reward or gradient variance.

At fixed $a = 0.90$, κ grows from approximately 0.122 at $p_s = 0.40$ to 0.800 at $p_s = 0.10$. At the common feasible correlation $\rho = 0.3$, the corresponding gaps are 0.763 and 0.560. This is a comparison of channel quantities, not an identical predicted change in policy accuracy. Subject to the approximately stable-accuracy assumption, increased error overlap motivates a candidate explanation to test alongside reported rise-and-collapse behavior (Lin, 2026; Fan et al., 2026; Qi et al., 2026).

Flat or reversed gain curves challenge P1; non-monotone collapse steps challenge P2; pass@32 above the frozen control near the endpoint challenges P3; reversed training orderings challenge P4 or P7; absence of the predicted relative correlation/gain pattern challenges P5; and absent or reversed cross-family drift challenges P6. Such outcomes limit the empirical account without contradicting the reward-gap identity under its assumptions. The same grid on the replicate policy, the alternative self-judge operationalization, and transfer evaluation on MATH-500 and AIME-2024 test the scope of these hypotheses.

E Effective sample size for mean verifier error

For verifier-error indicators with equal variance σ_E^2 and common pairwise correlation ρ_w, the mean error rate satisfies

$$\mathrm{Var}\left(\frac{1}{K}\sum_{i=1}^{K} E_i\right) = \frac{\sigma_E^2}{K}\big[1 + (K-1)\rho_w\big], \qquad n_{\mathrm{eff}} = \frac{K}{1 + (K-1)\rho_w}. \tag{17}$$

At the reported within-group error correlation $\rho_w = 0.53$ for Qwen2.5-1.5B responses (Xin, 2026), this model gives $n_{\mathrm{eff}} \simeq 1.7$ for $K = 8$. The quantity is a variance-equivalent sample size for estimating the mean verifier error. It does not directly describe the variance or effective sample size of GRPO rewards or gradients, because $v_i = c_i \oplus E_i$ also depends on correctness. The paper's ρ, which relates a verifier error to a self-judge error on the same response, is distinct from the within-group ρ_w.

F Compute budget and reproducibility details

Table 5 retains the generated-token caps implied by the protocol. The constants are $P = 256$ prompts, $K = 8$ responses, a 768-token response and evaluation cap, a 16-token anchored verdict cap, a 512-token commit-first solution cap, and a nominal evaluation budget of 1,030 problems with

Table 5: Generated-token caps implied by the protocol constants, per arm–round–seed and over the full $15 \times 5 \times 3$ configuration. Sharing initial responses reduces the realized total.

Stage	Per arm–round–seed	Full run ($\times 225$)
Responses ($256 \times 8 \times 768$)	1,572,864	353,894,400
Anchored judge (2048×16)	32,768	7,372,800
Commit-first judge (256×512)	131,072	29,491,200
Cross-family judge, both forms	163,840	36,864,000
Greedy evaluation (1030×768)	791,040	177,984,000
Sampled evaluation ($1030 \times 32 \times 768$)	25,313,280	5,695,488,000
Total	28,004,864	6,301,094,400

32 sampled responses each. The full-run factor is $15 \times 5 \times 3 = 225$ arm–round–seed combinations. These are upper bounds attained only if every generation reaches its cap. Sharing round-0 responses and judgments lowers the realized total.

The specified server configuration uses one NVIDIA H200 with 141 GB, vLLM for response generation, and PyTorch for updates. The setup lists vLLM ≥ 0.8, PyTorch ≥ 2.5, transformers ≥ 4.51, and peft ≥ 0.15. The harness records per-round wall-clock time and generated-token counts and supports checkpointed, resumable runs. These software and compute settings describe the protocol rather than measured training throughput for the complete study.

For scale, a local feasibility probe used Qwen2.5-3B-Instruct in bfloat16 on an M4 Max through MPS: 316 generated tokens per second at batch size 32, 66 at batch size 8, and a peak allocation of 15.3 GB. At the reported batch-32 rate, the tabulated cap corresponds to more than seven months of continuous generation. This local probe is distinct from the proposed primary and replicate training policies and is not a measurement of the five-round experiment.

G Extended background on self-improvement and verification

G.1 Self-improvement, sharpening, and ceilings

Self-generated supervision covers several learning mechanisms. STaR bootstraps reasoning from filtered and rationalized model outputs, Self-Rewarding Language Models uses model-based judgments to construct preferences, and SPIN develops a self-play fine-tuning objective (Zelikman et al., 2022; Yuan et al., 2025; Chen et al., 2024). These methods differ in their starting supervision, selection procedures, and optimization targets.

Sharpening theory studies how a model can concentrate probability on responses favored by its own reward signal (Huang et al., 2024). Generation–verification-gap studies examine the difference between producing and recognizing useful answers, while solver–verifier models analyze how coupled capabilities affect training trajectories (Song et al., 2025; Sun et al., 2026). Task-centric analyses further connect iterative improvement to initial competence and curriculum difficulty (Liu et al., 2026b). These theories address distinct mathematical objects under different assumptions.

Empirical work also separates observable score changes from stronger claims about capability. Studies of the generalization gap and task rewards examine what self-generated supervision contributes beyond concentrating existing behavior (Qi et al., 2026; Mittal et al., 2026). Phantom Gains motivates measuring an untrained control under the same evaluation procedure before attributing transitions to learning (Xu et al., 2026). Our channel analysis complements these questions by varying where verifier mistakes occur on a fixed population.

G.2 Self-play and self-generated curricula

Self-play systems jointly determine what to practice and how to score it. Absolute Zero uses executable verification in a proposer–solver loop, whereas R-Zero studies self-evolving reasoning with model-generated tasks and pseudo-labels (Zhao et al., 2025; Huang et al., 2026). Agent0 adds tool-integrated reasoning to self-evolving agents (Xia et al., 2025). These systems motivate distin-

guishing the curriculum source from the source of evidence used to evaluate an answer. A task can be generated internally while its evaluation depends on an external execution environment.

Other approaches change the interaction structure. SPIRAL and Language Self-Play investigate game-based training; SPICE uses corpus environments; Scaling Self-Play with Self-Guidance introduces an additional guidance role; and SOAR studies teaching near the learner's competence boundary (Liu et al., 2026a; Kuba et al., 2025; Liu et al., 2025; Bailey et al., 2026; Sundaram et al., 2026). These designs can change problem sampling, candidate-answer distributions, available evidence, and proposer incentives alongside verifier accuracy.

Executors, corpora, formal checkers, and measured student progress provide different forms of grounding. None should be assigned independence merely because it is external to the generator. Incomplete tests, restricted task coverage, or selection effects can still produce systematic mistakes. Conversely, a model-based verifier may obtain useful evidence through a different prompt, context, or inference procedure. Comparing these alternatives requires measuring their errors on an explicitly defined population.

G.3 Intrinsic rewards, degradation, and evaluation

Test-time reinforcement learning, self-certainty, and confidence maximization explore feedback signals that do not require a supplied correctness label for every training response (Zuo et al., 2025; Zhao et al., 2026; Prabhudesai et al., 2025). Their empirical gains motivate studying when internal preferences align with task success. A continuous confidence score, a plurality answer, and a binary judgment are nevertheless distinct signals. Reducing all three to one designated self-judgment would remove information used by their respective training procedures.

Several studies investigate degradation during extended self-training. Work on unsupervised RLVR and self-training describes limitations of internal rewards, while CoRE examines consensus-based reward construction (He et al., 2026; Shafayat et al., 2025; Mehrish & Vascon, 2026). The weak-model setting is particularly relevant when sampled candidates rarely include a correct solution (Roy et al., 2025). Analyses of oversharpening and pass@k inversion further distinguish an increase in preferred responses from preservation of a broader set of successful trajectories (Fan et al., 2026; Zhou, 2026b). These findings concern the models, tasks, and training conditions studied.

Rise-and-collapse behavior has also been examined with an external grader (Lin, 2026). This motivates an audit of reward quality that considers generator–verifier correlation alongside changing task difficulty, reward exploitation, loss of diversity, and optimization dynamics. A useful audit therefore follows accuracy, sampled success, diversity, and verifier errors over rounds, with matched evaluation budgets and a frozen control. Our static channel provides one diagnostic for this audit.

G.4 Correlated errors and interventions on judgment

Dependence between mistakes can arise at several levels. Within-group analyses examine whether verifier errors on different completions co-vary, and verifier-cascade analyses study the reliability consequences of shared blind spots (Xin, 2026; Han, 2026). Neither quantity is automatically the correlation defined here between a verifier's error and a designated self-judgment error on the same item. A variance-based effective sample size for an exchangeable mean also does not directly determine the number of useful GRPO comparisons or training groups.

Reference-free judge studies motivate changing the information available when a judgment is formed. In particular, committing to an answer before viewing a candidate separates an initial solution from subsequent agreement checking (Zhou, 2026a). Peer-predictive self-training uses cross-model aggregates, and JURY-RL combines proposed answers with formal verification (Feng et al., 2026; Chen et al., 2026). Such interventions may change accuracy, calibration, or dependence simultaneously. Any decorrelation effect must be established through measurement.

Noisy-verifier analyses and learned verification provide further context for imperfect feedback (Cai et al., 2026; Cobbe et al., 2021; Lightman et al., 2023; Wu & Raghunathan, 2026). Our construction specifies conditional flip probabilities under explicit assumptions. Only the identity channel $v = \hat{s}$ is the self-judgment endpoint used in our derivation; using the same model does not imply that

identity or error correlation one. Under natural error structures, accuracy and correlation alone need not determine the conditional reward gap.

G.5 Scaffold-level and meta-recursive improvement

A separate line of work improves the machinery around a model. The Darwin Gödel Machine, Huxley-Gödel Machine, and Hyperagents study changes to agent implementations or their improvement procedures (Zhang et al., 2026a; Wang et al., 2025; Zhang et al., 2026b). MetaRSI extends this perspective to systems that modify the improvement process itself (Tan et al., 2026). These settings can change search strategies, tool interfaces, and evaluation procedures without requiring the same weight-update mechanism considered in a policy-training loop.

Evaluation remains central because a modification must be selected before it can accumulate. Audits comparing code evolution with simpler budget-matched baselines caution against assigning gains to the proposed search mechanism without appropriate controls (Gideoni et al., 2026). Broader discussions of recursive self-improvement likewise identify verification as an open challenge (Duan et al., 2026; Yang et al., 2026). Extending our analysis to scaffold search requires a separate study that defines the candidate population, correctness criterion, self-judgment, and selection rule.