FLOWBALANCE: VERIFIER-GROUNDED SELF-IMPROVEMENT FROM ON-POLICY REASONING EXPERIENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

A reasoning model can improve from its own on-policy experience, but this inner loop is fragile: terminal verifiers provide reliable yet sparse supervision, while dense same-model guidance can reinforce false confidence or overconcentrate learning on a narrow solution mode. We introduce **FlowBalance**, a verifier-grounded self-improvement method that learns a normalized distribution over complete responses. For each on-policy trajectory, a frozen training-time view of the same policy uses privileged context to produce token-level log-probability gains, which are aggregated into a trajectory-level self-guidance score. FlowBalance calibrates this score with the verifier-derived group advantage: guidance is retained on positive-advantage trajectories, reversed on negative-advantage trajectories, and disabled when the rollout group provides no outcome preference. The resulting energy exponentially reweights a reference policy, and profiled trajectory balance fits the normalized target with one log-partition estimate per rollout group. This realizes *outcome-calibrated self-guidance via trajectory balance*, without a separate token-level imitation loss. Our analysis establishes within-group contrast preservation, a minimum-change reverse-KL characterization, monotonic verifier control of target reward, and an exact correction against false-positive self-guidance on rejected responses. On mathematical reasoning, FlowBalance improves average performance over FlowRL on both Qwen3-4B and Qwen3-8B, while also improving training speed and stability, avoiding direct OPSD's response-length collapse, and exhibiting higher correct-strategy diversity in a controlled AIME24 diagnostic.

1 INTRODUCTION

A central promise of post-training is that a reasoning model can improve from its own experience: sample solutions, evaluate what worked, and update the policy. We use *self-improvement* in this operational sense—repeated policy improvement from on-policy trajectories under training-time feedback—rather than to claim a fully autonomous system. For long-horizon reasoning, this loop must move probability toward correct reasoning while preserving useful alternative strategies (Guan et al., 2026; Kujawa et al., 2025; Ren & Sutherland, 2025).

Two failure modes make this loop fragile. Reinforcement learning with verifiable rewards (RLVR) grounds updates in reliable but sparse terminal feedback (Shao et al., 2024; Yu et al., 2025c). Distributional methods such as FlowRL translate this evidence into probability over complete responses, but outcome-only energies cannot exploit fine-grained evidence along a reasoning path (Zhu et al., 2025). Conversely, a frozen copy of the current policy can use a reference solution or task feedback to score every sampled token, providing dense privileged-hindsight guidance without an external teacher or additional generated responses (Zhao et al., 2026a; Hübotter et al., 2026). This guidance remains imperfect: it can favor rejected trajectories, shorten reasoning, or concentrate updates around a narrow mode (Kim et al., 2026b; Zhu et al., 2026). Blindly following it creates *self-confirmation*, turning the model's erroneous preference into supervision.

We therefore ask: *given on-policy reasoning experience, sparse verified outcomes, and dense but imperfect self-guidance, what normalized response distribution should the next policy learn?* We

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.

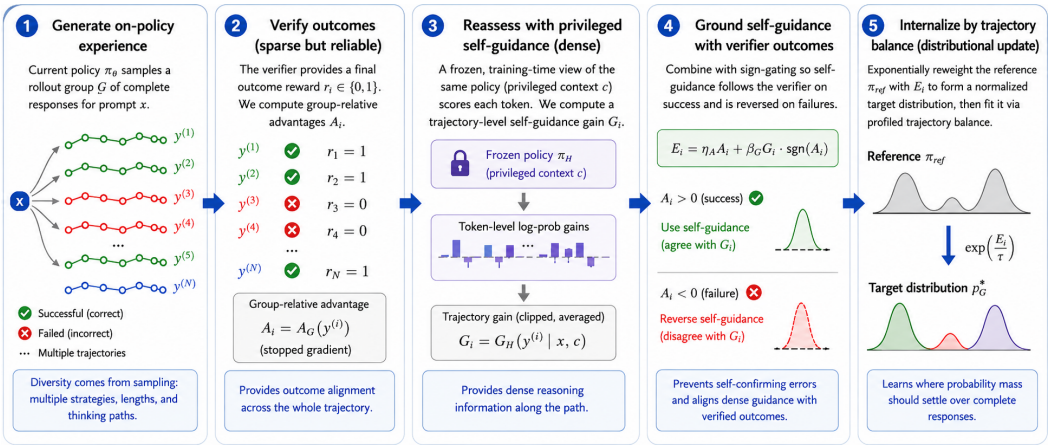


Figure 1: **FlowBalance as a verifier-grounded self-improvement cycle.** The current policy generates on-policy reasoning experience (1); a verifier supplies sparse but reliable outcome feedback (2); a frozen privileged-hindsight view of the same policy supplies dense self-guidance on the sampled tokens (3); sign gating grounds that guidance in verified outcomes (4); and profiled trajectory balance internalizes the resulting normalized response distribution (5). Refreshing the frozen snapshot repeats the inner loop.

introduce **FlowBalance**, using *outcome-calibrated self-guidance via trajectory balance*. A frozen privileged-hindsight view scores on-policy trajectories using training-only context. FlowBalance aggregates these scores into a trajectory-level gain, retains it for positive verifier advantages, reverses it for negative advantages, and disables it at zero advantage. Profiled trajectory balance fits the resulting reference-supported Gibbs target, without a separate token-level imitation loss. The reference controls support and drift; guidance refines the verifier's direction.

Our contributions are:

- A complete-response distribution-matching objective that combines verifier advantages and dense same-model guidance in one reference-supported trajectory energy.
- Outcome calibration that turns false-positive guidance on rejected responses into a probability-ratio correction favoring verified responses.
- Target-level guarantees: profiling preserves within-group probability contrasts; the target minimizes reverse-KL displacement at its attained composite-energy level; and verifier control is monotonic for fixed guidance.
- Evidence of stronger mathematical reasoning across two model scales and broader successful strategies in a controlled diagnostic. Relative to FlowRL, FlowBalance improves the core four-benchmark average by 1.67 points on Qwen3-4B and 1.98 points on Qwen3-8B.

Experiments isolate policy updates on a fixed task distribution. The theory concerns the target and local fitting objective, without global neural-optimizer convergence guarantees; the diversity diagnostic uses one seed and LLM judgments. Appendix B provides the expanded formulation and evidence; Appendix A reviews related work.

2 Preliminaries

2.1 On-Policy Reasoning Experience

Let $x \sim \mathcal{D}$ be a reasoning prompt. A response is a variable-length token sequence $y = (y_1, \ldots, y_T) \in \mathcal{Y}(x)$. At token t, the state is $s_t = (x, y_{<t})$, and the trainable policy factorizes as

$$\pi_\theta(y \mid x) = \prod_{t=1}^{T} \pi_\theta(y_t \mid s_t). \tag{1}$$

At the start of each training iteration, we snapshot the current parameters as $\theta^- \leftarrow \theta$. The frozen rollout policy is π_{θ^-}, while the reference policy π_{ref} is a fixed copy of the initial checkpoint.

For each prompt, we sample a group of responses $\{y^{(i)}\}_{i=1}^{N} \sim \pi_{\theta^-}(\cdot \mid x)$. The verifier assigns the terminal reward $R_i = R(y^{(i)}; x)$ according to final-answer correctness. We compute the stopped group-relative advantage

$$A_i = \frac{R_i - \mu_R(x)}{\sigma_R(x) + \epsilon}, \tag{2}$$

where $\mu_R(x)$ and $\sigma_R(x)$ are the mean and standard deviation of the rewards in the response group. Rewards, group statistics, sampled trajectories, and advantages receive no gradient.

2.2 Privileged-Hindsight Self-Guidance

Each training prompt x is paired with training-only context c, such as a reference solution or task feedback. We evaluate the sampled token y_t through two views of the same frozen snapshot:

$$\pi_{\text{roll}}(\cdot \mid s_t) = \pi_{\theta^-}(\cdot \mid x, y_{<t}), \tag{3}$$

$$\pi_{\text{H}}(\cdot \mid s_t, c) = \pi_{\theta^-}(\cdot \mid x, c, y_{<t}). \tag{4}$$

The rollout view generates the response without observing c. The privileged-hindsight view sees c only after the response has been sampled and scores the same tokens; it generates no replacement trajectory and receives no gradient. At inference, the deployed policy observes neither c nor the hindsight view.

This same-model scoring construction is related to on-policy self-distillation, but its role in FlowBalance is narrower: it supplies a stopped dense feature for defining a trajectory-level target. FlowBalance does not optimize a separate token-level imitation loss.

FlowBalance reweights the reference policy with a stopped trajectory energy at temperature $\tau > 0$. Section 3 defines the target on the realized rollout group; Appendix C gives the response-space formulation.

3 FlowBalance: Verifier-Grounded Self-Improvement

FlowBalance maps a rollout group of self-generated reasoning trajectories into a normalized next-policy target. The verifier supplies sparse but grounded outcome evidence, while a privileged-hindsight view of the same frozen policy supplies dense self-guidance on the sampled tokens. FlowBalance first combines these stopped signals into an outcome-calibrated trajectory energy and then fits the induced complete-response distribution through trajectory balance. The dense branch therefore shapes *which distribution is learned*; it is not optimized as a separate token-level imitation objective.

3.1 Training-Time Self-Guidance from Privileged Hindsight

For a sampled response y, define the clipped token-level hindsight gain

$$\delta_t^{\text{H}}(y; x, c) = \text{clip}\left(\log \pi_{\text{H}}(y_t \mid s_t, c) - \log \pi_{\text{ref}}(y_t \mid s_t), -B, B\right). \tag{5}$$

We aggregate these gains over the complete response:

$$G_{\text{H}}(y \mid x, c) = \frac{1}{T} \sum_{t=1}^{T} \delta_t^{\text{H}}(y; x, c). \tag{6}$$

G_{H} measures how much the privileged-hindsight view raises or lowers the average sampled-token log probability relative to the fixed reference policy. It is a trajectory feature computed on the policy's own on-policy experience. No token is resampled from π_{H}, and no gradient is propagated through π_{H} or G_{H}.

3.2 Outcome-Calibrated Self-Improvement Target

Dense self-guidance can be useful while still being wrong about the verified outcome. FlowBalance therefore uses the group-relative advantage to determine its direction:

$$E_{\mathrm{FlowBalance}}(y \mid x, c) = \eta_A A(y) + \beta_G G_{\mathrm{H}}(y \mid x, c)\,\mathrm{sgn}(A(y)), \tag{7}$$

where $\eta_A, \beta_G \geq 0$. If $A(y) > 0$, positive guidance increases the trajectory energy. If $A(y) < 0$, positive guidance is reversed, preventing a confidently scored failure from becoming self-reinforcing supervision. If $A(y) = 0$, the dense branch is disabled. All quantities on the right-hand side are stopped during the policy update.

Because A is group-relative in implementation, the practical update uses the realized rollout group $\mathcal{G}$:

$$p^{\star}_{\mathrm{FlowBalance},\mathcal{G}}(y^{(i)} \mid x, c) = \frac{\pi_{\mathrm{ref}}(y^{(i)} \mid x)\exp(E_{\mathrm{FlowBalance}}(y^{(i)} \mid x, c)/\tau)}{\sum_{j=1}^{N}\pi_{\mathrm{ref}}(y^{(j)} \mid x)\exp(E_{\mathrm{FlowBalance}}(y^{(j)} \mid x, c)/\tau)}. \tag{8}$$

The objective and profiled partition are evaluated on sampled complete trajectories. Appendix C gives the response-space target and its balance identities.

3.3 Trajectory Balance as a Normalized Self-Update

Let $Z_{\mathrm{FlowBalance}}(x, c)$ denote the response-space target partition (Appendix C); its log is profiled from the rollout group below. The trajectory-balance residual is

$$\Delta_{\mathrm{TB}}(y^{(i)}; x, c) = \tau \log Z_{\mathrm{FlowBalance}}(x, c) + \tau \log \frac{\pi_\theta(y^{(i)} \mid x)}{\pi_{\mathrm{ref}}(y^{(i)} \mid x)} - E_{\mathrm{FlowBalance}}(y^{(i)} \mid x, c). \tag{9}$$

The main loss is

$$\mathcal{L}_{\mathrm{FlowBalance}}(\theta) = \mathbb{E}_{(x,c)\sim\mathcal{D}}\left[\frac{1}{2N}\sum_{i=1}^{N}\Delta_{\mathrm{TB}}(y^{(i)}; x, c)^2\right]. \tag{10}$$

Rewards, advantages, self-guidance scores, partition estimates, and sampled responses are stopped. Gradients are taken only through the trainable-policy log probabilities $\log \pi_\theta(y^{(i)} \mid x)$.

The experiments use complete-response trajectory balance; Appendix D describes the subtrajectory extension.

3.4 Profiled Optimization over Rollout Groups

The partition can be represented by a learned prompt-conditioned estimator or profiled directly from the rollout group. We use the group estimator. Each response implies

$$\widehat{\log Z}_i(x, c) = \frac{E_{\mathrm{FlowBalance}}(y^{(i)} \mid x, c)}{\tau} - \log \frac{\pi_\theta(y^{(i)} \mid x)}{\pi_{\mathrm{ref}}(y^{(i)} \mid x)}, \tag{11}$$

and we set

$$\widehat{\log Z}_{\mathrm{FlowBalance}}(x, c) = \frac{1}{N}\sum_{i=1}^{N}\widehat{\log Z}_i(x, c). \tag{12}$$

Gradients are stopped through this estimate.

Algorithm 1 makes the self-improvement loop explicit. The frozen snapshot first generates experience without privileged context, then provides a training-only hindsight score on those same trajectories. The verifier grounds the direction of the score, and trajectory balance internalizes the resulting normalized distribution into the next policy snapshot.

4 Why FlowBalance Supports Verifier-Grounded Self-Improvement

Section 3 defines FlowBalance as a partition-normalized update over a realized group of the model's own trajectories. We analyze four properties that are useful for a self-improvement inner loop. First,

Algorithm 1 FlowBalance: verifier-grounded self-improvement from on-policy experience

Require: Prompt–context dataset $\mathcal{D}$; policy π_θ; fixed reference π_{ref}; verifier R; coefficients η_A, β_G, τ.

for each training iteration **do**
  Snapshot the current policy: $\theta^- \leftarrow \theta$.
  Sample a minibatch $\{(x_b, c_b)\}_{b=1}^{B} \sim \mathcal{D}$.
  Generate N responses $y_b^{(i)} \sim \pi_{\theta^-}(\cdot \mid x_b)$ for every prompt.
  Evaluate rewards and compute stopped group-relative advantages $A_{b,i}$ using Eq. (2).
  Score sampled tokens under π_θ, π_{ref}, and the frozen hindsight view $\pi_{\text{H}} = \pi_{\theta^-}(\cdot \mid x_b, c_b, y_{<t})$.
  Compute $G_{\text{H}}(y_b^{(i)} \mid x_b, c_b)$ and $E_{\text{FlowBalance}}(y_b^{(i)})$ using Eqs. (6) and (7).
  Profile $\widehat{\log Z}_{\text{FlowBalance}}(x_b, c_b)$ by Eq. (12).
  Compute $\Delta_{\text{TB}}(y_b^{(i)}; x_b, c_b)$ and update θ with Eq. (10).
end for

profiling the group partition should retain the relative evidence contained in all sampled responses. Second, the target should move conservatively from the reference. Third, the verifier should remain an explicit control on target reward even in the presence of dense self-guidance. Fourth, false-positive guidance on a rejected response should not become self-reinforcement. All statements below are conditional on the realized rollout group and treat verifier and guidance quantities as stopped. Proofs are in Appendix F.

4.1 Distributional Structure of the Self-Update

For each sampled response $y^{(i)}$, let

$$E_i = \eta_A A_i + \beta_G G_{\text{H}}(y^{(i)} \mid x, c)\,\text{sgn}(A_i) \tag{13}$$

be its stopped FlowBalance energy. The normalized group target is

$$p_i^\star = \frac{\pi_{\text{ref}}(y^{(i)} \mid x)\exp(E_i/\tau)}{\sum_{j=1}^{N} \pi_{\text{ref}}(y^{(j)} \mid x)\exp(E_j/\tau)}. \tag{14}$$

Let $p^\star = (p_1^\star, \ldots, p_N^\star)$. All results are conditional on the realized rollout group.

Proposition 1 (Profiled balance uses all within-group contrasts)**.** *For any candidate group distribution with positive mass on the realized responses, the profiled trajectory-balance loss is zero if and only if every relative probability contrast matches*

$$\frac{\pi(y^{(i)})}{\pi(y^{(j)})} = \frac{\pi_{\text{ref}}(y^{(i)} \mid x)}{\pi_{\text{ref}}(y^{(j)} \mid x)} \exp\left(\frac{E_i - E_j}{\tau}\right), \qquad i, j \in [N]. \tag{15}$$

In particular, whenever the group target is representable, the global minimum is zero; the unknown prompt-level normalizer removes exactly one common offset and preserves the remaining $N-1$ contrast directions.

The group partition therefore does not collapse the rollout group to one comparison. It absorbs only the common energy shift while retaining all relative evidence supplied by the model's own sampled experiences.

Proposition 2 (Conservative minimum-change self-update)**.** *Let the reference be restricted and renormalized to the realized rollout group when computing KL. For any group distribution $p \in \Delta_N$,*

$$\text{KL}(p\|\pi_{\text{ref}}) = \text{KL}(p^\star\|\pi_{\text{ref}}) + \text{KL}(p\|p^\star) + \frac{1}{\tau}\left(\sum_{i=1}^{N} p_i E_i - \sum_{i=1}^{N} p_i^\star E_i\right). \tag{16}$$

Consequently, among all group distributions that attain at least the expected FlowBalance energy of $p^\star$, the FlowBalance target is the unique minimum-reverse-KL displacement from the reference.

FlowBalance is therefore the smallest reference-supported distributional tilt that reaches its composite self-improvement energy level. This is a property of the target distribution, not a claim that an arbitrary finite neural-network update exactly reaches the target in one optimizer step.

4.2 Verifier Control against Self-Confirmation

Proposition 3 (Verifier weight monotonically improves target reward)**.** *Hold the self-guidance scores and the realized rollout group fixed. Increasing the verifier coefficient η_A monotonically shifts the FlowBalance target toward higher-reward responses. In particular,*

$$\frac{\partial}{\partial \eta_A}\mathbb{E}_{p^\star}[R] = \frac{\mathrm{Var}_{p^\star}(R)}{\tau(\sigma_R(x)+\epsilon)} \geq 0. \tag{17}$$

The verifier remains an explicit control knob even when dense self-guidance is present: for fixed guidance, increasing its weight cannot reduce the target's expected verifier reward.

Proposition 4 (Outcome calibration corrects false-positive self-guidance)**.** *Consider a verified success y_+ and a verifier-rejected response y_- in the same mixed-outcome group. Relative to otherwise identical* ungated *self-guidance, sign gating changes their target probability ratio according to*

$$\frac{p^\star_{\text{gated}}(y_+)}{p^\star_{\text{gated}}(y_-)} = \frac{p^\star_{\text{ungated}}(y_+)}{p^\star_{\text{ungated}}(y_-)} \exp\left(\frac{2\beta_G G_{\mathrm{H}}(y_- \mid x, c)}{\tau}\right). \tag{18}$$

Hence, when $G_{\mathrm{H}}(y_- \mid x, c) > 0$, positive self-guidance on a rejected response is converted from self-reinforcement into a probability-ratio correction favoring the verified response.

This proposition captures the intended useful-but-imperfect regime: privileged hindsight can assign positive local likelihood both to correct reasoning and to plausible failures. FlowBalance preserves the ability of guidance to distinguish successful modes while reversing its false-positive support where the verifier rejects the response. Appendix G.1 provides exactly enumerable diagnostics for the same target, including a reliability sweep over imperfect self-guidance.

5 Experiments

We evaluate whether FlowBalance supports reliable policy self-improvement from on-policy reasoning experience along four axes. First, we compare final mathematical reasoning accuracy against a verifier-only RL baseline, direct on-policy distillation, a verifier–distillation hybrid, and outcome-only trajectory balance. Second, we inspect update efficiency, late-training stability, and response-length behavior. Third, we isolate verifier grounding and self-guidance strength. Fourth, we test whether the learned response distribution retains multiple successful strategies through an LLM-judged AIME24 diagnostic. Appendix H.1 gives the training and decoding details, Appendix H.3 gives the diversity protocol, and Appendix H.2 gives the full ablation tables.

5.1 Self-Improvement Evaluation Setup

We evaluate Qwen3-4B and Qwen3-8B (Yang et al., 2025a). Every deployed policy receives only the problem statement. For FlowBalance, a frozen current-policy snapshot additionally sees the training solution or task feedback only while scoring already sampled tokens; the resulting self-guidance is unavailable at inference. RLSD uses an analogous frozen current-policy scoring path, whereas OPSD uses a fixed privileged teacher according to its original formulation. All methods share the same training prompts, rollout group size, verifier, response-length cap, checkpoint schedule, and evaluation script within each backbone. We compare GRPO (Shao et al., 2024; Yu et al., 2025c), OPSD (Zhao et al., 2026a), RLSD (Yang et al., 2026), FlowRL (Zhu et al., 2025), and FlowBalance. Table 1 reports final benchmark accuracy, and Figure 2 reports the intermediate traces used to diagnose update efficiency, stability, and response-length behavior. Appendix H.1, especially Table 5, lists the concrete training and decoding fields used for reproducibility.

5.2 Verifier-Grounded Self-Improvement on Mathematical Reasoning

Main Results. Table 1 summarizes mathematical reasoning results across two model backbones. All entries report step-180 results over five seeds. AIME24 is evaluated with Pass@16 to reflect competition-style sampling, while HMMT25, Minerva, MATH500, and OlympiadBench use Pass@1. The reported aggregate averages summarize broad performance rather than tuning to a single benchmark.

Table 1: Mathematical reasoning results across Qwen3-4B and Qwen3-8B. AIME24 uses Pass@16; all other benchmarks use Pass@1. Entries are percentages reported as mean ± sample standard deviation over five seeds at step 180. "Avg." averages the five reported benchmark means.

Model	Method	AIME24@16	HMMT25@1	Minerva@1	MATH500@1	Olympiad@1	Avg.
Qwen3-4B	GRPO	78.00 ± 1.83	26.67 ± 2.36	51.18 ± 1.36	92.04 ± 0.98	63.68 ± 0.58	62.31
	OPSD	65.33 ± 3.80	14.67 ± 2.98	47.28 ± 0.56	87.56 ± 1.34	55.76 ± 1.47	54.12
	RLSD	73.33 ± 2.36	21.33 ± 3.80	50.29 ± 0.88	91.44 ± 0.52	61.36 ± 0.66	59.55
	FlowRL	75.33 ± 1.83	30.67 ± 4.94	$\mathbf{51.99 \pm 1.51}$	92.84 ± 0.62	65.25 ± 0.49	63.22
	FlowBalance	$\mathbf{80.00 \pm 0.00}$	$\mathbf{32.00 \pm 2.98}$	50.51 ± 0.56	$\mathbf{93.28 \pm 0.59}$	$\mathbf{65.49 \pm 0.92}$	**64.26**
Qwen3-8B	GRPO	85.33 ± 1.83	31.33 ± 7.67	52.87 ± 1.02	93.16 ± 0.83	64.78 ± 0.62	65.49
	OPSD	48.67 ± 3.80	4.00 ± 3.65	38.46 ± 3.85	74.56 ± 4.81	40.09 ± 3.57	41.16
	RLSD	82.67 ± 3.65	28.00 ± 1.83	52.94 ± 1.38	93.44 ± 0.17	63.56 ± 1.19	64.12
	FlowRL	86.67 ± 0.00	30.67 ± 4.35	52.79 ± 1.37	92.92 ± 0.50	66.20 ± 1.05	65.85
	FlowBalance	$\mathbf{89.33 \pm 1.49}$	$\mathbf{34.67 \pm 9.89}$	$\mathbf{53.68 \pm 0.78}$	$\mathbf{93.52 \pm 0.30}$	$\mathbf{66.85 \pm 0.46}$	**67.61**

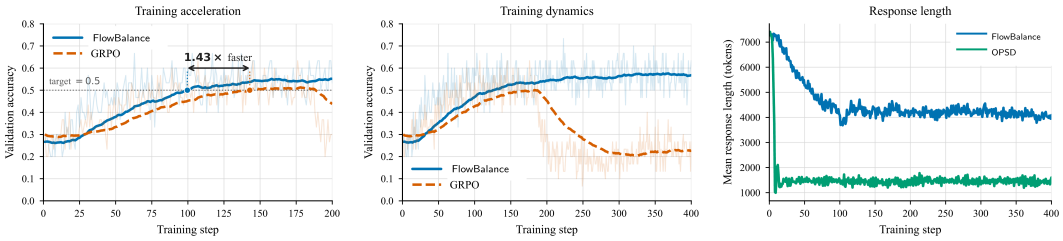
On Qwen3-4B, FlowBalance reaches a five-benchmark average of 64.26, improving over GRPO by 1.95 points, OPSD by 10.14 points, RLSD by 4.71 points, and FlowRL by 1.04 points. It outperforms OPSD and RLSD on all five benchmarks and improves over GRPO on AIME24, HMMT25, MATH500, and OlympiadBench; FlowRL obtains the highest Minerva mean on this backbone. On Qwen3-8B, FlowBalance reaches 67.61 and obtains the best mean on every reported benchmark, improving over GRPO by 2.12 points, OPSD by 26.45 points, RLSD by 3.49 points, and FlowRL by 1.76 points. The comparison to FlowRL isolates the value of self-guidance inside the same trajectory-distribution family, while the comparisons to GRPO, OPSD, and RLSD show that neither verifier-only optimization nor direct local self-imitation explains the aggregate gain. A plausible explanation for OPSD's poor reasoning performance is that imitating a solution-conditioned privileged teacher can suppress epistemic verbalization and shorten reasoning, weakening uncertainty-driven exploration and self-correction (Kim et al., 2026b); this is consistent with Figure 2c. The largest FlowBalance gain over stronger baselines appears on AIME24 Pass@16, where allocating mass across multiple correct trajectories is especially useful. The smaller but consistent gains on MATH500 and OlympiadBench indicate that this distributional update does not trade broad Pass@1 accuracy for a sampling-heavy metric.

Training Dynamics. Figure 2 summarizes the dynamics of repeated policy improvement on Qwen3-8B. The accuracy panels use matched rollout and evaluation budgets for FlowBalance and GRPO, while the response-length panel compares FlowBalance with direct OPSD. First, FlowBalance reaches 0.5 AIME24 validation accuracy in about 100 steps, compared with roughly 143 for GRPO, a $1.43\times$ reduction in updates to the threshold. Second, FlowBalance remains near its peak accuracy throughout 400 steps, whereas GRPO degrades sharply after approximately step 180. Third, FlowBalance maintains substantially longer reasoning trajectories than direct OPSD, which rapidly collapses to short responses. These observations support a narrow but useful claim: in this fixed-task inner loop, verifier-grounded self-guidance accelerates and stabilizes the update while avoiding the shortcut behavior of ungrounded local imitation.

Ablation Study. We study the two coefficients in the FlowBalance energy:

$$E_{\mathrm{FlowBalance}}(y) = \eta_A A_{\mathcal{G}}(y) + \beta_G G_{\mathrm{H}}(y)\,\mathrm{sgn}\big(A_{\mathcal{G}}(y)\big). \tag{19}$$

We vary η_A to control verifier grounding and β_G to control self-guidance strength. The ablation uses one-dimensional sweeps around the default setting rather than a full grid, so that each table isolates one mechanism while holding the other coefficient fixed. Table 2 sweeps $\eta_A \in \{5, 10, 15\}$, and Table 3 sweeps the self-guidance coefficient $\beta_G \in \{1, 2, 3\}$. The default run corresponds to $\eta_A = 15$ and $\beta_G = 1$; the $\eta_A = 10$ run and the full β_G sweep are now complete. Each ablation row uses the same backbone, training budget, seeds, and five-benchmark average as Table 1; in addition to accuracy, we track training stability to detect settings that improve one benchmark at the cost of unstable optimization.



(a) Training acceleration. (b) Training stability. (c) Response length.

Figure 2: **Training dynamics on mathematical reasoning with Qwen3-8B. (a) Training acceleration:** FlowBalance reaches 0.5 AIME24 validation accuracy in about 100 steps, compared with roughly 143 steps for GRPO (1.43× faster). **(b) Training stability:** over 400 training steps, FlowBalance remains near its peak performance, whereas GRPO degrades sharply after approximately step 180. **(c) Response length:** direct OPSD rapidly collapses to substantially shorter responses, whereas FlowBalance maintains longer reasoning trajectories. In the accuracy panels, solid curves show smoothed trends and lighter curves show the corresponding per-step measurements.

Table 2: Ablation over verifier coefficient η_A.

η_A	5	10	15
Avg.	65.65	65.41	67.61

Table 3: Ablation over self-guidance coefficient β_G.

β_G	1	2	3
Avg.	67.61	66.48	65.95

5.3 Correct-Strategy Diversity under Self-Improvement

A reliable self-improvement update should not obtain accuracy only by sharpening one successful template. FlowBalance also broadens the kinds of correct solutions represented by the policy. We measure this with an LLM-judged semantic strategy diagnostic on AIME24: a GPT-5.5 judge extracts each full trajectory's mathematical representation and tools, clusters trajectories with the same core strategy, and we report Simpson strategy diversity,

$$D_{\text{Simpson}} = 1 - \sum_k p_k^2, \tag{20}$$

where p_k is the fraction of correct trajectories in strategy cluster k. This metric is the probability that two sampled correct trajectories use different semantic strategies; Appendix H.3 gives the full protocol and scope.

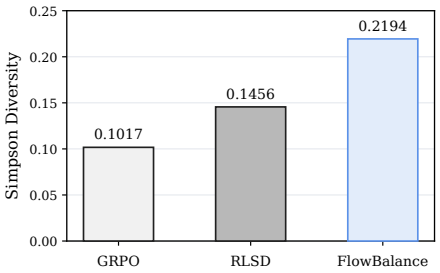


Figure 3: **LLM-judged strategy diversity.** Correct-only Simpson diversity on AIME24; protocol and scope are given in Appendix H.3.

Figure 3 reports correct-only Simpson diversity. FlowBalance achieves 0.2194, compared with 0.1017 for GRPO and 0.1456 for RLSD. Thus, within this diagnostic setting, FlowBalance's successful responses span a broader range of semantic solution strategies.

Case Study. Table 4 illustrates what the LLM-judged strategy clusters are intended to capture: differences in the mathematical objects and representations that make a solution work. We choose this example for its structural contrast rather than for per-problem win rate. GRPO uses a standard and broadly applicable Cayley–Menger determinant solution: it treats the tetrahedron as six pairwise distances and obtains the volume from a distance invariant. FlowBalance follows a different route by recognizing $41 = 4^2 + 5^2$, $80 = 4^2 + 8^2$, and $89 = 5^2 + 8^2$, which reveals a hidden $4 \times 5 \times 8$ rectangular-box embedding and reduces the volume computation to a scalar triple product. This case study therefore grounds the aggregate LLM-judged result: FlowBalance can place probability mass on correct trajectories that are not merely rephrasings of the dominant solution template, but use a different mathematical representation altogether. Appendix H.5 gives three additional

Table 4: **Case study on AIME24 Problem 23.** Boxed spans indicate key reasoning actions in representative correct trajectories; "$\cdots$" denotes omitted intermediate text.

	Content (boxed = reasoning actions; "$\cdots$" = omitted)
Question	Tetrahedron $ABCD$ satisfies $AB = CD = \sqrt{41}$, $AC = BD = \sqrt{80}$, and $BC = AD = \sqrt{89}$. If the common distance from the incenter to the four faces is $m\sqrt{n}/p$, find $m+n+p$.
GRPO	**Common Cayley–Menger route.** "$\cdots$ all faces have sides $\sqrt{41}, \sqrt{80}, \sqrt{89}$ $\cdots$ $\boxed{S = 24\sqrt{21}}$ $\cdots$ $\boxed{\text{build Cayley–Menger matrix } M \text{ from } 41, 80, 89}$ $\cdots$ $\boxed{\text{diagonalize the distance block by symmetry}}$ $\cdots$ $\boxed{\det M = 819200}$ $\cdots$ $\boxed{V = 160/3}$ $\cdots$ $\boxed{r = 3V/S = 20\sqrt{21}/63}$ $\cdots$ $\boxed{20 + 21 + 63 = 104}$ $\cdots$"
FlowBalance	**Hidden box-embedding route.** "$\cdots$ observe $\boxed{41 = 4^2 + 5^2, 80 = 4^2 + 8^2, 89 = 5^2 + 8^2}$ $\cdots$ $\boxed{\text{embed in a } 4 \times 5 \times 8 \text{ box}}$ $\cdots$ $\boxed{A = (0,0,0), B = (4,5,0), C = (4,0,8), D = (0,5,8)}$ $\cdots$ $\boxed{\text{verify all six edge lengths}}$ $\cdots$ $\boxed{\text{scalar triple product gives } V = 160/3}$ $\cdots$ $\boxed{r = 20\sqrt{21}/63}$ $\cdots$ $\boxed{20 + 21 + 63 = 104}$ $\cdots$"

AIME24 examples in the same format, covering envelope versus multiple-root reasoning, planar circle tangency versus implicit-surface normals, and coordinate elimination versus trigonometric hyperbola parameterization.

6 Conclusions and Discussions

Conclusion. FlowBalance treats policy self-improvement as a distribution-learning problem over the model's own on-policy reasoning experience. A frozen privileged-hindsight view of the same policy supplies dense self-guidance, a verifier determines whether that guidance should be retained or reversed, and profiled trajectory balance fits the resulting reference-supported target over complete responses. This separation is central: dense guidance informs the target, but the policy is updated only through the normalized trajectory-balance objective. The theory shows that the profiled update preserves all within-group contrasts, is the minimum reverse-KL displacement at its attained energy level, retains monotonic verifier control of target reward, and converts false-positive guidance on rejected trajectories into an anti-self-confirmation correction. Across Qwen3-4B and Qwen3-8B, FlowBalance achieves the strongest five-benchmark average among GRPO, OPSD, RLSD, FlowRL, and FlowBalance. It also reaches the AIME24 validation threshold in fewer updates than GRPO, remains stable over extended training, avoids direct OPSD's response-length collapse, and exhibits higher correct-only semantic strategy diversity in the controlled AIME24 diagnostic.

Limitations and scope. Four limitations delimit these claims. First, the large-scale experiments focus on mathematical reasoning, so generalization to agentic, multimodal, or other long-horizon domains remains open. Second, the response-length comparison diagnoses a correlated failure mode but does not establish that longer responses cause higher accuracy. Third, the diversity analysis is an LLM-judged AIME24 diagnostic at one checkpoint and seed rather than a multi-seed human study. Fourth, we isolate the experience-to-policy inner loop on a fixed prompt distribution; FlowBalance is not by itself a full self-evolving system that generates or curates new tasks. A natural next step is to combine the distributional update with an outer-loop curriculum or guided task generator, including R-Few-style Challenger–Solver systems, while separately measuring task drift, policy stability, and strategy diversity.

AI USE STATEMENT

We used AI tools for drafting and language editing throughout the manuscript, as well as for assisting with the implementation of parts of the proposed method. All AI-assisted code, analyses, and written content were reviewed, verified, and revised by the authors, who take full responsibility for the final manuscript and results.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024.

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27381–27394, 2021.

Yoshua Bengio, Salem Lahlou, Tristan Deleu, Edward J. Hu, Mo Tiwari, and Emmanuel Bengio. GFlowNet foundations. *Journal of Machine Learning Research*, 24(210):1–55, 2023.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 535–541, 2006. doi: 10.1145/1150402.1150464.

Li Du, Kanglong Li, Yihang Cao, Kaiyan Lin, Yuanqi Ma, Yuxuan Lu, Xi Chen, Jinyang Yang, Zhiyuan Ma, Zhixuan Chu, Xuanjing Huang, Hui Yu, Jie M. Zhang, Furu Wei, Bowen Yu, and Lei Li. GFlowRL: Scaling distribution-matching RL to large language models, 2026.

Jiannan Guan, Qiguang Chen, Libo Qin, Dengyun Peng, Jinhao Liu, Liangyu Huo, Jian Xie, and Wanxiang Che. Beware of reasoning overconfidence: Pitfalls in the reasoning process for multi-solution tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 30843–30851, 2026. doi: 10.1609/aaai.v40i36.40342.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

Zixun Huang, Jiayi Sheng, and Zeyu Zheng. Variance-aware baselines and adaptive learning rates for reinforcement learning with verifiable rewards, 2026.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation, 2026.

Nan Jia, Haojin Yang, Xing Ma, Jiesong Lian, Shuailiang Zhang, Weipeng Zhang, Ke Zeng, Xunliang Cai, and Zequn Sun. Asymmetric on-policy distillation: Bridging exploitation and imitation at the token level, 2026.

Jaehoon Kim and Dongha Lee. OPSD compresses what RLVR teaches: A post-RL compaction stage for reasoning models, 2026.

Jeonghye Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Rebellious student: Reversing teacher signals for reasoning exploration with self-distilled RLVR, 2026a.

Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Why does self-distillation (sometimes) degrade the reasoning capability of LLMs?, 2026b.

Zeno Kujawa, John Poole, Dobrik Georgiev, Danilo Numeroso, Henry Fleischmann, and Pietro Liò. Neural algorithmic reasoning with multiple correct solutions. In *Workshop on Machine Learning on Graphs in the Era of Generative Artificial Intelligence at KDD*, 2025.

Zhenwen Liang, Sidi Lu, Wenhao Yu, Kishan Panaganti, Yujun Zhou, Haitao Mi, and Dong Yu. Can LLMs guide their own exploration? gradient-guided reinforcement learning for LLM reasoning, 2025.

Nikolay Malkin, Moksh Jain, Emmanuel Bengio, Chen Sun, and Yoshua Bengio. Trajectory balance: Improved credit assignment in GFlowNets. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5955–5967, 2022.

Leyi Pan, Shuchang Tao, Yunpeng Zhai, Shun Zhang, Wenzhe Li, Xiao Liu, Jibin Wang, Jian Shen, Zhaopeng Tu, and Kaisheng Zhao. RLCSD: Reinforcement learning with contrastive on-policy self-distillation, 2026.

Kishan Panaganti, Zaiyan Xu, Dileep Kalathil, and Mohammad Ghavamzadeh. Robust reinforcement learning using offline data. In *Advances in Neural Information Processing Systems*, 2022.

Kishan Panaganti, Zhenwen Liang, Wenhao Yu, Haitao Mi, and Dong Yu. Group distributionally robust optimization-driven reinforcement learning for LLM reasoning, 2026.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models, 2026.

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *International Conference on Learning Representations*, 2025.

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.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning, 2026.

Jiangnan Xia, Yucheng Shi, Yu Yang, Kishan Panaganti, Zhenwen Liang, and Ninghao Liu. Reasoning or memorization? direction-aware diversity exploration in LLM reinforcement learning, 2026.

Zhengpeng Xie, Qiang Zhang, Fan Yang, Marco Hutter, and Renjing Xu. Simple policy optimization, 2025.

Haoran Xu, Chufan Shi, Chenyang Yang, Ming Li, Kaiyan Liu, Weinan Zhang, and Jun Wang. β-OPSD: Deriving with Policy Optimization, Training with Self-Distillation, 2026a.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, Zhipeng Wang, and Alborz Geramifard. TIP: Token importance in on-policy distillation, 2026b.

Zaiyan Xu, Sushil Vemuri, Kishan Panaganti, Dileep Kalathil, Rahul Jain, and Deepak Ramachandran. Robust LLM alignment via distributionally robust direct preference optimization. In *Advances in Neural Information Processing Systems*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Kexin Yang, Le Yu, Xinyu Wang, Jing Zhou, and Qwen Team. Qwen3 technical report, 2025a.

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled RLVR, 2026.

Shihui Yang, Chengfeng Dou, Peidong Guo, Kai Lu, Qiang Ju, Fei Deng, and Rihui Xin. DCPO: Dynamic clipping policy optimization, 2025b.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models, 2026.

Dian Yu, Yulai Zhao, Kishan Panaganti, Linfeng Song, Haitao Mi, and Dong Yu. Every question has its own value: Reinforcement learning with explicit human values, 2025a.

Fangxu Yu, Lai Jiang, Haoqiang Kang, Shibo Hao, and Lianhui Qin. Flow of reasoning: Training LLMs for divergent reasoning with minimal examples, 2025b. ICML 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Li, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Yan Lin, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale, 2025c.

Wenhao Yu, Zhenwen Liang, Chengsong Huang, Kishan Panaganti, Tianqing Fang, Haitao Mi, and Dong Yu. Guided self-evolving llms with minimal human supervision, 2025d. URL https://arxiv.org/abs/2512.02472.

Xinlei Yu, Gen Li, Qingyi Si, Guibin Zhang, Yuqi Xu, Congcong Wang, Shuai Dong, Kaiwen Tuo, Xiangyu Zeng, Kaituo Feng, Qunzhong Wang, Yang Shi, Xiaobin Hu, Xiangyu Yue, Jiaqi Wang, and Shuicheng Yan. DOPD: Dual on-policy distillation, 2026.

Hongbin Zhang, Chaozheng Wang, Kehai Chen, Youcheng Pan, Yang Xiang, Jinpeng Wang, and Min Zhang. Tailoring teaching to aptitude: Direction-adaptive self-distillation for llm reasoning, 2026.

Kaichen Zhang, Yuzhong Hong, Junwei Bao, Hongfei Jiang, Yang Song, Dingqian Hong, and Hui Xiong. GVPO: Group variance policy optimization for large language model post-training, 2025.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models, 2026a.

Ziqi Zhao, Xinyu Ma, Liu Yang, Yujie Feng, Daiting Shi, Jingzhou He, Xin Xin, Zhaochun Ren, and Xiao-Ming Wu. Reflective on-policy self-distillation for language model reasoning across domains, 2026b.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025.

Yujun Zhou, Zhenwen Liang, Haolin Liu, Wenhao Yu, Kishan Panaganti, Linfeng Song, Dian Yu, Xiangliang Zhang, Haitao Mi, and Dong Yu. Evolving language models without labels: Majority drives selection, novelty promotes variation, 2025.

Siqi Zhu, Xuyan Ye, Hongyu Lu, Weiye Shi, and Ge Liu. The many faces of on-policy distillation: Pitfalls, mechanisms, and fixes, 2026.

Xuekai Zhu, Daixuan Cheng, Dinghuai Zhang, Hengli Li, Kaiyan Zhang, Che Jiang, Youbang Sun, Ermo Hua, Yuxin Zuo, Xingtai Lv, Qizheng Zhang, Lin Chen, Fanghao Shao, Bo Xue, Yunchong Song, Zhenjie Yang, Ganqu Cui, Ning Ding, Jianfeng Gao, Xiaodong Liu, Bowen Zhou, Hongyuan Mei, and Zhouhan Lin. FlowRL: Matching reward distributions for LLM reasoning, 2025.

A Related Work

Reinforcement learning from verifiable outcomes. RLVR is a major route to improving LLM reasoning with automatically checkable answers (Shao et al., 2024; Yu et al., 2025c; Zhou et al., 2025; Yu et al., 2025a; Liang et al., 2025; Xia et al., 2026). Group-based methods estimate advantages from multiple responses to one prompt without a separately learned critic. Recent work improves this optimization through sequence-level ratios, trust regions, variance-aware weighting, dynamic clipping, and adaptive baselines or learning rates (Zheng et al., 2025; Xie et al., 2025; Zhang et al., 2025; Yang et al., 2025b; Huang et al., 2026). These methods provide reliable outcome grounding, but their supervision remains response-level. FlowBalance addresses the complementary experience-to-policy question: how sparse outcomes and dense training-time evidence should define a normalized distribution over complete responses.

Privileged self-guidance and on-policy distillation. Knowledge distillation transfers behavior from a teacher to a student, while on-policy distillation evaluates student-generated trajectories under a teacher (Buciluǎ et al., 2006; Hinton et al., 2015; Agarwal et al., 2024). Recent methods use a privileged view of the same model, conditioned on reference answers, demonstrations, correct rollouts, or environment feedback (Hübotter et al., 2026; Zhao et al., 2026a; Penaloza et al., 2026; Shenfeld et al., 2026; Yang et al., 2026; Ye et al., 2026). This produces dense token-level information, but it can also shorten reasoning, suppress uncertainty, leak unavailable context into the learning signal, or plateau and degrade (Yang et al., 2026; Kim et al., 2026b; Kim & Lee, 2026; Zhu et al., 2026). Existing work modifies, selects, or contrasts local teacher signals (Pan et al., 2026; Yu et al., 2026; Zhao et al., 2026b; Zhang et al., 2026; Jia et al., 2026; Xu et al., 2026b; Kim et al., 2026a); concurrent β-OPSD derives an efficient KL-regularized target with a tunable reference–teacher tradeoff (Xu et al., 2026a). FlowBalance uses the same-model privileged scorer only as a stopped *self-guidance feature*. The algorithmic object is instead a verifier-grounded complete-response distribution fitted by trajectory balance, with no separate token-level imitation loss.

Guided self-improvement and self-evolving systems. Self-evolving systems add an outer loop that generates, selects, or schedules new learning experiences. R-Few, for example, stabilizes a Challenger–Solver loop with a small pool of human anchors and an online difficulty-based curriculum (Yu et al., 2025d). Its primary object is the evolving task and curriculum distribution. FlowBalance studies a complementary inner-loop object: given prompts and on-policy solution trajectories, how should those experiences update the policy without amplifying false self-confidence or collapsing successful-strategy diversity? We do not combine the two systems experimentally; rather, R-Few illustrates how an outer-loop experience generator could supply tasks to a FlowBalance-style distributional policy update.

Distribution matching and trajectory balance. GFlowNets learn stochastic construction policies that sample objects in proportion to an unnormalized reward rather than concentrating only on maximizers (Bengio et al., 2021; 2023; Xu et al., 2025; Panaganti et al., 2026; 2022). Trajectory balance enforces this condition over complete trajectories and improves long-horizon credit assignment (Malkin et al., 2022). For LLM reasoning, Flow of Reasoning and FlowRL instantiate trajectory balance with a prompt-conditioned partition term (Yu et al., 2025b; Zhu et al., 2025), while GFlowRL replaces the auxiliary partition network with an in-batch rollout-group estimate (Du et al., 2026). FlowBalance builds on this distributional lineage but changes the target energy: verified outcome advantages are combined with outcome-calibrated privileged-hindsight guidance, and the resulting reference-supported distribution is profiled over the rollout group. The comparison to FlowRL in our experiments isolates this additional guidance inside the same broad trajectory-balance family.

B Extended Introduction and Contribution Evidence

B.1 Motivation and self-improvement setup

A central promise of post-training is that a reasoning model can improve from its own experience: sample several solutions, evaluate what worked, and update the policy so that the next round of experience is better. We use *self-improvement* in this operational sense—repeated policy improvement from the model's own on-policy trajectories under training-time feedback—rather than to claim a

fully autonomous or closed-loop system. For long-horizon reasoning, this inner loop must learn more than a higher mean reward. It must move probability toward correct reasoning, preserve useful alternative strategies, and avoid repeatedly sharpening one locally preferred trace (Guan et al., 2026; Kujawa et al., 2025; Ren & Sutherland, 2025).

Two failure modes make this loop fragile. First, reinforcement learning with verifiable rewards (RLVR) provides dependable outcome grounding, but only through sparse terminal feedback (Shao et al., 2024; Yu et al., 2025c). A response may contain hundreds or thousands of tokens while the verifier supplies one final reward; group-based objectives improve comparison across responses but still attach a response-level signal to every decision. Distributional methods such as FlowRL improve how this terminal evidence is translated into probability over complete responses (Zhu et al., 2025), yet outcome-only energies cannot exploit fine-grained evidence along a sampled reasoning path.

Second, the model can reassess its own sampled trajectories using a more informed training-time view. A frozen copy of the current policy can condition on a reference solution or task feedback and score every sampled token (Zhao et al., 2026a; Hübotter et al., 2026). This same-model, privileged-hindsight signal is dense and inexpensive: it requires neither a larger external model nor additional generated responses. However, dense self-guidance is not automatically trustworthy. Because the scoring view observes information unavailable at inference, it may favor a plausible but ultimately rejected trajectory, shorten reasoning, suppress uncertainty-driven exploration, or concentrate updates around a narrow local mode (Kim et al., 2026b; Zhu et al., 2026). Blindly following such confidence creates *self-confirmation*: the model's own erroneous preference becomes the next update's supervision.

We therefore ask a distributional self-improvement question:

> *Given on-policy reasoning experience, sparse verified outcomes, and dense but imperfect self-guidance, what normalized response distribution should the next policy learn?*

We introduce **FlowBalance**, a verifier-grounded self-improvement operator that answers this question through *outcome-calibrated self-guidance via trajectory balance*. The current policy generates a rollout group. A verifier supplies stopped group-relative advantages. A frozen privileged-hindsight view of the same policy then scores the already sampled tokens using training-only context. FlowBalance aggregates these scores into a trajectory-level guidance gain, uses the verifier to determine its direction, and fits the resulting reference-supported Gibbs target with profiled trajectory balance. The policy is optimized only through this complete-response distribution-matching objective; there is no separate token-level imitation loss.

B.2 Target distribution and trajectory-balance objective

For a prompt x, let $\mathcal{G} = (y^{(1)}, \dots, y^{(N)})$ be the on-policy rollout group and let $A_i = A_{\mathcal{G}}(y^{(i)})$ be its stopped group-relative verifier advantage. The privileged-hindsight scoring view π_{H} conditions on training-only context c and yields clipped token-level log-probability gains relative to the reference policy. Their trajectory average is the self-guidance gain $G_{\mathrm{H}}(y^{(i)} \mid x, c)$. FlowBalance defines

$$E_{\mathrm{FlowBalance},\mathcal{G}}(y^{(i)} \mid x, c) = \eta_A A_i + \beta_G G_{\mathrm{H}}(y^{(i)} \mid x, c)\,\mathrm{sgn}(A_i). \tag{21}$$

The verifier term anchors the direction of improvement. When $A_i > 0$, positive self-guidance can reinforce a verified response; when $A_i < 0$, the same positive guidance is reversed rather than allowed to self-confirm a failure; and when $A_i = 0$, the dense branch is disabled. The guidance therefore refines the verifier's direction instead of overriding it.

Conditioned on the realized group, FlowBalance defines the reference-supported target

$$p^{\star}_{\mathrm{FlowBalance},\mathcal{G}}(y \mid x, c) \propto \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{E_{\mathrm{FlowBalance},\mathcal{G}}(y \mid x, c)}{\tau}\right), \tag{22}$$

and fits it through the profiled trajectory-balance residual

$$\Delta_{\mathrm{TB}}(y; x, c, \mathcal{G}) = \tau \log Z_{\mathrm{FlowBalance},\mathcal{G}}(x, c) + \tau \log \frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} - E_{\mathrm{FlowBalance},\mathcal{G}}(y \mid x, c). \tag{23}$$

One log-partition estimate is profiled per rollout group. The reference policy controls support and drift, the verifier determines outcome direction, privileged hindsight provides dense within-trajectory

evidence, and trajectory balance converts their composite energy into a normalized distribution over complete responses.

B.3 Scope and theoretical interpretation

This construction can also be viewed as a *distributional inner-loop update*: a task source supplies prompts, the current policy supplies experience, FlowBalance maps $\{(y_i, R_i, G_{\mathrm{H},i})\}_{i=1}^{N}$ to a target distribution, and policy fitting produces the next policy. Our experiments deliberately hold the task distribution fixed, isolating this experience-to-policy update from outer-loop task generation or curriculum evolution. The latter is complementary rather than assumed by the method.

Our analysis studies the induced target directly. We show that profiling the partition preserves every within-group probability contrast; characterize FlowBalance as the unique minimum reverse-KL displacement from the reference at its attained composite-energy level; establish monotonic verifier control of the target reward statistic; and quantify exactly how outcome calibration converts false-positive self-guidance on a rejected response into a probability-ratio correction favoring a verified response. These are properties of the target and its local fitting objective, not a claim of global convergence for an arbitrary neural optimizer.

B.4 Empirical overview

On mathematical reasoning, FlowBalance obtains the strongest overall averages on both Qwen3-4B and Qwen3-8B among GRPO, OPSD, RLSD, FlowRL, and FlowBalance. Relative to FlowRL, it improves the core four-benchmark average over AIME24, HMMT25, MATH500, and OlympiadBench by 1.67 points on Qwen3-4B and 1.98 points on Qwen3-8B, while also improving the full aggregate average. On Qwen3-8B, FlowBalance reaches 0.5 AIME24 validation accuracy in about 100 steps versus roughly 143 for GRPO, remains stable over 400 steps, and avoids the response-length collapse observed under direct OPSD. A controlled AIME24 diagnostic further finds higher correct-only semantic strategy diversity than GRPO and RLSD.

B.5 Expanded contributions and evidence

Our contributions are:

- **A distributional self-improvement objective that improves over both verifier-only RL and RL–self-guidance hybrids.** GRPO improves reasoning from sparse verifier outcomes, while RLSD augments verifier-based policy optimization with dense same-model guidance. FlowBalance addresses a different question: *what normalized complete-response distribution should the next policy learn from these signals?* It combines verifier advantages and privileged-hindsight guidance into one reference-supported trajectory energy, then fits the induced distribution through profiled trajectory balance. This makes the comparison to GRPO and RLSD especially informative: FlowBalance is not merely adding more supervision to GRPO or inserting another guidance term into an RL objective; it changes the policy-update object from a local optimization signal into an explicitly normalized distribution over the model's own reasoning trajectories.

  Concrete evidence. On the core four-benchmark average over AIME24, HMMT25, MATH500, and OlympiadBench, FlowBalance improves over GRPO by 2.60 points on Qwen3-4B (67.69 versus 65.10) and 2.44 points on Qwen3-8B (71.09 versus 68.65). Relative to RLSD, the corresponding gains are 5.83 points on Qwen3-4B (67.69 versus 61.87) and 4.18 points on Qwen3-8B (71.09 versus 66.92). The same ordering holds on the full five-benchmark aggregate: FlowBalance exceeds GRPO by 1.95 and 2.12 points and RLSD by 4.71 and 3.49 points on the 4B and 8B backbones, respectively. On Qwen3-8B, it also reaches 0.5 AIME24 validation accuracy in about 100 steps rather than roughly 143 for GRPO and remains near its peak over 400 steps while GRPO degrades after about step 180. These results isolate the practical contribution: learning a verifier-grounded trajectory distribution improves over both outcome-only RL and a direct RL–dense-guidance combination across two model scales.

- **Outcome-calibrated self-guidance that converts false confidence into a verifier-grounded correction.** The privileged-hindsight view is deliberately treated as *useful but imperfect*: it can recognize promising reasoning structure, yet it can also assign positive confidence to a trajectory

that fails the verifier. FlowBalance therefore uses $+G_{\mathrm{H}}$ when the group-relative advantage is positive, $-G_{\mathrm{H}}$ when it is negative, and zero guidance when the group provides no outcome preference. This is more than a heuristic filter. Proposition 4 shows that, for a verified success and a rejected response, sign gating multiplies their target probability ratio by the exact factor $\exp(2\beta_G G_{\mathrm{H}}(y_-)/\tau)$ relative to ungated guidance. Dense same-model evidence can therefore refine verified successes without allowing a confidently scored failure to become self-reinforcing supervision.

Concrete evidence. In the exact four-mode diagnostic, reward-only shaping reaches success mass 0.818, ungated self-guidance reaches 0.832, and FlowBalance reaches 0.900 while also increasing the robust-success mode from 0.327 under reward-only shaping to 0.440. In the binary false-confidence sweep at $G_- = 0.5$, FlowBalance attains target success probability 0.894, compared with 0.817 for reward-only shaping and 0.807 for ungated shaping. The language-model results show the same qualitative distinction: direct OPSD rapidly collapses to short responses, whereas FlowBalance maintains long reasoning traces; moreover, increasing the guidance coefficient from $\beta_G = 1$ to 3 lowers AIME24 from 89.33 to 86.00 and HMMT25 from 34.67 to 30.00. The benefit therefore comes from *calibrated* self-guidance, not from simply making the dense signal stronger.

- **A conservative and information-efficient trajectory-balance update with exact target-level guarantees.** FlowBalance profiles one scalar log-partition value per rollout group, but this nuisance parameter removes only the common offset: all $N - 1$ independent within-group probability contrasts remain available to train the policy. The induced target is also the unique minimum reverse-KL displacement from the reference among group distributions reaching its expected composite-energy level, and increasing the verifier coefficient monotonically increases the target's expected verifier reward for fixed guidance. Together, these results identify the update as a principled minimum-change self-improvement step that uses the full relative structure of the rollout group while keeping verified reward as an explicit control variable.

  Concrete evidence. The exact synthetic diagnostics quantify both effects. A matched-energy full-support alternative requires reverse KL 0.973, whereas the FlowBalance exponential tilt requires only 0.273; the alternative is therefore $3.6\times$ farther from the reference. At rollout-group size $N = 32$, profiling all contrasts yields 2.92% of the local Gaussian parameter risk of a one-contrast-per-group estimator—about $1/34$ as much risk. Consistent with this conservative, all-contrast update—without claiming that the target-level results alone prove optimizer convergence—FlowBalance reaches 0.5 AIME24 validation accuracy in about 100 steps rather than roughly 143 for GRPO and remains near its peak over 400 steps, while GRPO degrades sharply after approximately step 180.

- **Performance and semantic-diversity evidence that self-improvement need not collapse to one reasoning mode.** FlowBalance improves mathematical reasoning while preserving a broader successful response distribution. On Qwen3-4B, it improves over FlowRL on all four core reported benchmarks: AIME24, HMMT25, MATH500, and OlympiadBench. On Qwen3-8B, it obtains the best mean on every benchmark in the main table and improves the core four-benchmark average by 1.98 points over FlowRL. These results matter for the paper's central claim because the largest gains are not confined to a single metric: they span both multi-sample competition reasoning and single-sample accuracy, while the training traces show that the policy does not obtain them by collapsing to the short-response behavior seen under direct OPSD.

  Concrete evidence. In the controlled AIME24 diagnostic, correct-only Simpson strategy diversity is 0.2194 for FlowBalance, compared with 0.1017 for GRPO and 0.1456 for RLSD. The associated traces differ in their mathematical representation, not merely their wording: FlowBalance discovers a hidden $4\times5\times8$ box embedding where GRPO uses Cayley–Menger, and the appendix shows envelope versus multiple-root reasoning, meridian-section versus implicit-normal geometry, and coordinate elimination versus secant–tangent hyperbola parameterization. Although this is a one-seed LLM-judged diagnostic rather than a population-level diversity guarantee, it supplies concrete qualitative evidence that the learned distribution places mass on genuinely different correct derivations instead of stylistic rewrites of one dominant template.

C Response-Space Targets and Balance Identities

C.1 Reference-Supported Target Distribution

We consider a normalized target over complete responses,

$$p^\star(y \mid x, c) = \frac{1}{Z(x,c)} \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{E(y \mid x, c)}{\tau}\right), \tag{24}$$

where $E(y \mid x, c)$ is a stopped trajectory energy, $\tau > 0$ is its temperature, and $Z(x, c)$ is the partition function. The reference policy fixes support and controls drift; the energy specifies relative preference among complete responses. Section 3 defines FlowBalance's outcome-calibrated self-guidance energy and fits the target through trajectory balance.

C.2 FlowBalance target and normalization

FlowBalance defines the unnormalized and normalized targets

$$\widetilde{p}_{\mathrm{FlowBalance}}(y \mid x, c) = \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{E_{\mathrm{FlowBalance}}(y \mid x, c)}{\tau}\right), \tag{25}$$

$$p^\star_{\mathrm{FlowBalance}}(y \mid x, c) = \frac{\widetilde{p}_{\mathrm{FlowBalance}}(y \mid x, c)}{Z_{\mathrm{FlowBalance}}(x, c)}, Z_{\mathrm{FlowBalance}}(x, c) = \sum_{y \in \mathcal{Y}(x)} \widetilde{p}_{\mathrm{FlowBalance}}(y \mid x, c). \tag{26}$$

The three factors have distinct roles: π_{ref} retains reference support, A supplies verified outcome direction, and G_{H} provides dense within-trajectory evidence.

The response-space notation describes the corresponding ideal Gibbs distribution; the objective and profiled partition are evaluated on sampled complete trajectories.

Why a partition-normalized target? A local dense score specifies how individual sampled tokens look under privileged hindsight, but not where total probability mass should settle across complete responses. The partition term converts relative trajectory energies into a probability-conserving target. Consequently, self-guidance can refine the distribution within the verifier's direction without becoming an independent local loss. Appendix E gives the corresponding equilibrium interpretation.

C.3 Complete balance and pairwise probability ratios

The complete-trajectory balance equation is

$$\tau \log Z_{\mathrm{FlowBalance}}(x, c) + \tau \log \frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} - E_{\mathrm{FlowBalance}}(y \mid x, c) = 0. \tag{27}$$

At zero residual, the partition cancels in pairwise probability ratios:

$$\log \frac{\pi_\theta(y^{(a)} \mid x)}{\pi_\theta(y^{(b)} \mid x)} = \log \frac{\pi_{\mathrm{ref}}(y^{(a)} \mid x)}{\pi_{\mathrm{ref}}(y^{(b)} \mid x)} + \frac{E_{\mathrm{FlowBalance}}(y^{(a)} \mid x, c) - E_{\mathrm{FlowBalance}}(y^{(b)} \mid x, c)}{\tau}. \tag{28}$$

Thus the energy controls relative preference among the model's own sampled experiences, while $\log Z_{\mathrm{FlowBalance}}$ absorbs only the common prompt-level offset.

D Subtrajectory Balance

The same principle can be applied between intermediate states. Let $Z_{\mathrm{FlowBalance}}(s)$ denote the continuation partition from state s, with $Z_{\mathrm{FlowBalance}}(s_{\mathrm{term}}) = 1$. Define the per-token shaped increment

$$r_t^{\mathrm{FlowBalance}} = \tau \log \pi_{\mathrm{ref}}(y_t \mid s_t) + \frac{\beta_G}{T} \delta_t^{\mathrm{H}} \operatorname{sgn}(A) + \eta_A A \, \mathbf{1}\{t = T\}. \tag{29}$$

For an interval $i{:}j$, the subtrajectory residual is

$$\Delta_{i:j} = \tau \log Z_{\mathrm{FlowBalance}}(s_i) + \tau \sum_{t=i}^{j-1} \log \pi_\theta(y_t \mid s_t) - \tau \log Z_{\mathrm{FlowBalance}}(s_j) - \sum_{t=i}^{j-1} r_t^{\mathrm{FlowBalance}}. \tag{30}$$

The endpoint partitions account for different continuation distributions. Sampling intervals and minimizing $\mathbb{E}[\Delta_{i:j}^2]$ provides a denser fitting objective along long responses; the experiments in this paper use the complete-response implementation unless otherwise stated.

E Detailed-Balance View of FlowBalance

Detailed balance. Let q be a probability distribution on the response space $\mathcal{Y}(x)$, and let $K(y \to y')$ be a Markov transition kernel. The kernel is reversible with respect to q if, for every pair $y, y' \in \mathcal{Y}(x)$,

$$q(y)K(y \to y') = q(y')K(y' \to y). \tag{31}$$

This is the detailed-balance condition: at equilibrium, the probability flow from y to y' equals the reverse flow from y' to y. Summing Eq. (31) over y shows that q is a stationary distribution of K. Whenever both transition probabilities are positive, detailed balance also gives

$$\frac{K(y \to y')}{K(y' \to y)} = \frac{q(y')}{q(y)}. \tag{32}$$

FlowBalance as an equilibrium distribution. For a fixed prompt–context pair (x, c), FlowBalance defines the Gibbs target

$$p^\star(y \mid x, c) = \frac{1}{Z_{\mathrm{FlowBalance}}(x, c)} \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{E_{\mathrm{FlowBalance}}(y \mid x, c)}{\tau} \right). \tag{33}$$

If a reversible Markov kernel were constructed with $p^\star$ as its equilibrium distribution, its forward–reverse transition ratio would satisfy

$$\frac{K(y \to y')}{K(y' \to y)} = \frac{\pi_{\mathrm{ref}}(y' \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} \exp\left(\frac{E_{\mathrm{FlowBalance}}(y' \mid x, c) - E_{\mathrm{FlowBalance}}(y \mid x, c)}{\tau} \right). \tag{34}$$

Thus the reference policy supplies the baseline relative mass, while the FlowBalance energy exponentially reweights transitions toward responses preferred by the verifier-grounded self-guidance energy. The partition function does not appear in this pairwise ratio, but it is required to turn these relative preferences into a normalized distribution.

Trajectory-balance interpretation. FlowBalance does not construct or simulate the Markov kernel K. Instead, trajectory balance directly imposes the global identity

$$Z_{\mathrm{FlowBalance}}(x, c)\pi_\theta(y \mid x) = \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{E_{\mathrm{FlowBalance}}(y \mid x, c)}{\tau} \right) \tag{35}$$

for each complete response. Taking logarithms and multiplying by τ gives the trajectory-balance residual

$$\tau \log Z_{\mathrm{FlowBalance}}(x, c) + \tau \log \frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} - E_{\mathrm{FlowBalance}}(y \mid x, c). \tag{36}$$

When this residual is zero for all responses, π_θ equals the normalized target in Eq. (33) and therefore has the same pairwise probability ratios as the equilibrium distribution associated with Eq. (34). In this sense, detailed balance provides an equilibrium interpretation, whereas trajectory balance is the learning constraint that fits the desired distribution without explicitly defining pairwise response transitions.

F Proofs for Theoretical Analyses

Throughout this appendix we condition on the same fixed prompt–context pair and realized rollout group as in Section 4. We use the main-text notation

$$E_i = \eta_A A_i + \beta_G G_{\mathrm{H}}(y^{(i)} \mid x, c) \operatorname{sgn}(A_i), \qquad p_i^\star = \frac{\pi_{\mathrm{ref}}(y^{(i)} \mid x) \exp(E_i/\tau)}{\sum_{j=1}^N \pi_{\mathrm{ref}}(y^{(j)} \mid x) \exp(E_j/\tau)}. \tag{37}$$

When KL is computed over the realized rollout group, $\pi_{\text{ref}}(\cdot \mid x)$ denotes its restriction and renormalization to that group.

F.1 Distributional Properties

Proposition F.1 (Profiled balance matches all within-group contrasts)**.** *For a candidate policy π and scalar partition/intercept z, consider the realized-group trajectory-balance loss*

$$\mathcal{L}_N(\pi, z) = \frac{1}{2N} \sum_{i=1}^{N} \left[\tau z + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i \right]^2. \tag{38}$$

After profiling out z, the trajectory-balance loss is zero if and only if every within-group relative probability contrast matches

$$\frac{\pi(y^{(i)})}{\pi(y^{(j)})} = \frac{\pi_{\text{ref}}(y^{(i)} \mid x)}{\pi_{\text{ref}}(y^{(j)} \mid x)} \exp\left(\frac{E_i - E_j}{\tau} \right), \qquad i, j \in [N]. \tag{39}$$

Whenever the group target is representable, the global minimum is therefore zero. Profiling removes only one common group-level offset and preserves the remaining $N - 1$ contrast directions.

Proof. For a candidate policy π, the trajectory-balance residual for response $y^{(i)}$ and scalar partition/intercept z is

$$\tau z + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i. \tag{40}$$

The empirical trajectory-balance loss over the realized group is therefore

$$\mathcal{L}_N(\pi, z) = \frac{1}{2N} \sum_{i=1}^{N} \left[\tau z + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i \right]^2. \tag{41}$$

For fixed π, this is a convex quadratic function of the single scalar z. Differentiating Eq. (41) with respect to z gives

$$\frac{\partial \mathcal{L}_N(\pi, z)}{\partial z} = \frac{\tau}{N} \sum_{i=1}^{N} \left[\tau z + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i \right]. \tag{42}$$

Thus the profiled intercept $\widehat{z}$ is the unique solution of

$$\sum_{i=1}^{N} \left[\tau \widehat{z} + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i \right] = 0. \tag{43}$$

After profiling, the loss is zero if and only if every squared residual in Eq. (41) is zero at $z = \widehat{z}$. Hence, for every pair i, j,

$$\tau \widehat{z} + \tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i = 0, \tag{44}$$

$$\tau \widehat{z} + \tau \log \frac{\pi(y^{(j)})}{\pi_{\text{ref}}(y^{(j)} \mid x)} - E_j = 0. \tag{45}$$

Subtracting the second equality from the first cancels the profiled scalar $\tau \widehat{z}$ and yields

$$\tau \log \frac{\pi(y^{(i)})}{\pi(y^{(j)})} - \tau \log \frac{\pi_{\text{ref}}(y^{(i)} \mid x)}{\pi_{\text{ref}}(y^{(j)} \mid x)} = E_i - E_j. \tag{46}$$

Dividing by τ, exponentiating, and rearranging the reference-policy ratio gives Eq. (39).

Conversely, if Eq. (39) holds for all pairs, then the quantities

$$\tau \log \frac{\pi(y^{(i)})}{\pi_{\text{ref}}(y^{(i)} \mid x)} - E_i \tag{47}$$

are all equal to the same constant. Choosing z to be minus that constant divided by τ makes every trajectory-balance residual zero. Therefore zero profiled loss is equivalent to matching all within-group target log probability-ratio contrasts. The scalar z removes exactly one common offset and cannot change any pairwise contrast. □

Proposition F.2 (Minimum reference displacement)**.** *Let π_{ref} denote the reference distribution restricted and renormalized to the realized rollout group. Then, for any group distribution $p \in \Delta_N$,*

$$\mathrm{KL}(p\|\pi_{\text{ref}}) = \mathrm{KL}(p^\star\|\pi_{\text{ref}}) + \mathrm{KL}(p\|p^\star) + \frac{1}{\tau}\left(\sum_{i=1}^N p_i E_i - \sum_{i=1}^N p_i^\star E_i\right). \tag{48}$$

Consequently, among all group distributions satisfying $\sum_i p_i E_i \geq \sum_i p_i^\star E_i$, the FlowBalance target $p^\star$ uniquely minimizes the reverse-KL displacement $\mathrm{KL}(p\|\pi_{\text{ref}})$.

Proof. From Eq. (14), for each group element,

$$\log \frac{p_i^\star}{\pi_{\text{ref}}(y^{(i)} \mid x)} = \frac{E_i}{\tau} - \log \sum_{j=1}^N \pi_{\text{ref}}(y^{(j)} \mid x) \exp(E_j/\tau). \tag{49}$$

Let $p \in \Delta_N$ be any group distribution. Then

$$\mathrm{KL}(p\|\pi_{\text{ref}}) = \sum_{i=1}^N p_i \log \frac{p_i}{p_i^\star} + \sum_{i=1}^N p_i \log \frac{p_i^\star}{\pi_{\text{ref}}(y^{(i)} \mid x)} \tag{50}$$

$$= \mathrm{KL}(p\|p^\star) + \frac{1}{\tau}\sum_{i=1}^N p_i E_i - \log \sum_{j=1}^N \pi_{\text{ref}}(y^{(j)} \mid x) \exp(E_j/\tau). \tag{51}$$

Applying the same identity with $p = p^\star$ gives

$$\mathrm{KL}(p^\star\|\pi_{\text{ref}}) = \frac{1}{\tau}\sum_{i=1}^N p_i^\star E_i - \log \sum_{j=1}^N \pi_{\text{ref}}(y^{(j)} \mid x) \exp(E_j/\tau). \tag{52}$$

Subtracting this expression from Eq. (51) yields Eq. (48):

$$\mathrm{KL}(p\|\pi_{\text{ref}}) = \mathrm{KL}(p^\star\|\pi_{\text{ref}}) + \mathrm{KL}(p\|p^\star) + \frac{1}{\tau}\left(\sum_{i=1}^N p_i E_i - \sum_{i=1}^N p_i^\star E_i\right). \tag{53}$$

If p attains at least the expected FlowBalance energy of $p^\star$, the last term is nonnegative; $\mathrm{KL}(p\|p^\star) \geq 0$ as well, with equality only when $p = p^\star$. Hence $p^\star$ is the unique minimum-reverse-KL displacement from the reference among all such distributions. □

F.2 Effects of Verifier Weighting and Sign Gating

Proposition F.3 (Verifier weight monotonically improves reward statistics)**.** *Hold the self-guidance scores and the realized rollout group fixed. For the GRPO-normalized advantage*

$$A_i = \frac{R_i - \bar{R}}{\sigma_R(x) + \epsilon}, \tag{54}$$

increasing the verifier coefficient η_A monotonically increases the expected reward under the FlowBalance target:

$$\frac{\partial}{\partial \eta_A}\mathbb{E}_{p^\star}[R] = \frac{\mathrm{Var}_{p^\star}(R)}{\tau(\sigma_R(x) + \epsilon)} \geq 0. \tag{55}$$

Proof. Hold the self-guidance scores fixed. The only dependence of $p_i^\star$ on η_A is through the term $\eta_A A_i$. Differentiating the normalized exponential-family form gives

$$\frac{\partial p_i^\star}{\partial \eta_A} = \frac{p_i^\star}{\tau}\left(A_i - \sum_{j=1}^N p_j^\star A_j\right). \tag{56}$$

Therefore

$$\frac{\partial}{\partial \eta_A}\mathbb{E}_{p^\star}[R] = \frac{1}{\tau}\mathrm{Cov}_{p^\star}(R, A). \tag{57}$$

For the GRPO-normalized advantage used in the main text,

$$A_i = \frac{R_i - \bar{R}}{\sigma_R(x) + \epsilon}, \tag{58}$$

where $\bar{R}$ and $\sigma_R(x) + \epsilon$ are fixed for the realized group. Hence

$$\mathrm{Cov}_{p^\star}(R, A) = \frac{\mathrm{Var}_{p^\star}(R)}{\sigma_R(x) + \epsilon}, \tag{59}$$

and thus

$$\frac{\partial}{\partial \eta_A} \mathbb{E}_{p^\star}[R] = \frac{\mathrm{Var}_{p^\star}(R)}{\tau(\sigma_R(x) + \epsilon)} \geq 0, \tag{60}$$

which proves Eq. (55). □

Proposition F.4 (Sign gating corrects self-guidance support on failures)**.** *Consider a verified success y_+ with positive advantage and a verifier-rejected response y_- with negative advantage in the same rollout group. Relative to otherwise identical ungated self-guidance shaping, sign gating changes their target probability ratio according to*

$$\frac{p^\star_{\mathrm{gated}}(y_+)}{p^\star_{\mathrm{gated}}(y_-)} = \frac{p^\star_{\mathrm{ungated}}(y_+)}{p^\star_{\mathrm{ungated}}(y_-)} \exp\left(\frac{2\beta_G G_{\mathrm{H}}(y_- \mid x, c)}{\tau}\right). \tag{61}$$

Hence positive privileged support on a verifier-rejected response is converted from self-reinforcement pressure into a probability-ratio correction favoring the verified response.

Proof. Consider a verified success y_+ and a verifier-rejected response y_- in the same group. Their probability ratio under the gated target is

$$\frac{p^\star_{\mathrm{gated}}(y_+)}{p^\star_{\mathrm{gated}}(y_-)} = \frac{\pi_{\mathrm{ref}}(y_+ \mid x)}{\pi_{\mathrm{ref}}(y_- \mid x)} \exp\left(\frac{E^{\mathrm{gated}}_+ - E^{\mathrm{gated}}_-}{\tau}\right), \tag{62}$$

where the group normalizer cancels. The corresponding ungated target has

$$\frac{p^\star_{\mathrm{ungated}}(y_+)}{p^\star_{\mathrm{ungated}}(y_-)} = \frac{\pi_{\mathrm{ref}}(y_+ \mid x)}{\pi_{\mathrm{ref}}(y_- \mid x)} \exp\left(\frac{E^{\mathrm{ungated}}_+ - E^{\mathrm{ungated}}_-}{\tau}\right). \tag{63}$$

The reference ratio cancels when comparing the gated and ungated probability ratios. Since y_+ has positive advantage and y_- has negative advantage,

$$E^{\mathrm{gated}}_+ - E^{\mathrm{gated}}_- = \eta_A \left(A_{\mathcal{G}}(y_+) - A_{\mathcal{G}}(y_-)\right) + \beta_G G_{\mathrm{H}}(y_+ \mid x, c) + \beta_G G_{\mathrm{H}}(y_- \mid x, c), \tag{64}$$

$$E^{\mathrm{ungated}}_+ - E^{\mathrm{ungated}}_- = \eta_A \left(A_{\mathcal{G}}(y_+) - A_{\mathcal{G}}(y_-)\right) + \beta_G G_{\mathrm{H}}(y_+ \mid x, c) - \beta_G G_{\mathrm{H}}(y_- \mid x, c). \tag{65}$$

The verifier difference $\eta_A \left(A_{\mathcal{G}}(y_+) - A_{\mathcal{G}}(y_-)\right)$ is identical in the two expressions, and the self-guidance contribution differs by $2\beta_G G_{\mathrm{H}}(y_- \mid x, c)$. Therefore

$$\frac{p^\star_{\mathrm{gated}}(y_+)}{p^\star_{\mathrm{gated}}(y_-)} = \frac{p^\star_{\mathrm{ungated}}(y_+)}{p^\star_{\mathrm{ungated}}(y_-)} \exp\left(\frac{2\beta_G G_{\mathrm{H}}(y_- \mid x, c)}{\tau}\right), \tag{66}$$

which proves Eq. (61). □

G Numerical Analysis Details

G.1 Exact Numerical Diagnostics for Verifier-Grounded Self-Improvement

We instantiate the unchanged FlowBalance target on exactly enumerable response spaces. Every distribution below is obtained by normalizing

$$q(y) \propto \rho(y) \exp\left(\frac{\eta A(y) + \beta G(y)\,\mathrm{sgn}(A(y))}{\tau}\right), \tag{67}$$

with no surrogate objective or alternative gate. These are qualitative diagnostics of the target, not additional language-model training runs. We organize them as a short narrative. The first plot gives the basic geometric intuition for verifier-grounded self-guidance. The second isolates the exact anti-self-confirmation effect in the simplest mixed-outcome group. The third maps the reliability boundary of self-guidance. The fourth returns to the two structural properties emphasized by the theory: conservative reference change and efficient use of all within-group contrasts.

G.1.1 Mechanism View: Grounded Self-Guidance across Multiple Modes

Figure 4 embeds four exact response modes as one-dimensional Gaussians: two failures and two successes. The verifier cannot distinguish the two successful modes, while privileged hindsight assigns its largest gain to the more robust success. It also assigns positive local likelihood to plausible failures, which is exactly the false-positive regime addressed by Proposition 4. We use

$$\rho = (0.30, 0.20, 0.30, 0.20), \quad A = (-1, -1, +1, +1), \quad G = (0.45, 0.25, 0.30, 0.75), \tag{68}$$

with $(\eta, \beta, \tau) = (0.75, 0.80, 1)$. Reward-only shaping raises total success mass to 0.818 but cannot prefer the robust successful mode. Ungated self-guidance identifies that mode, yet also reinforces positive guidance on failures, reaching success mass 0.832. FlowBalance preserves the useful ranking among successes while reversing the failed-response contribution, reaching success mass 0.900 and robust-success mass 0.440. The Gaussian curves visualize exact categorical masses; they do not approximate the target.

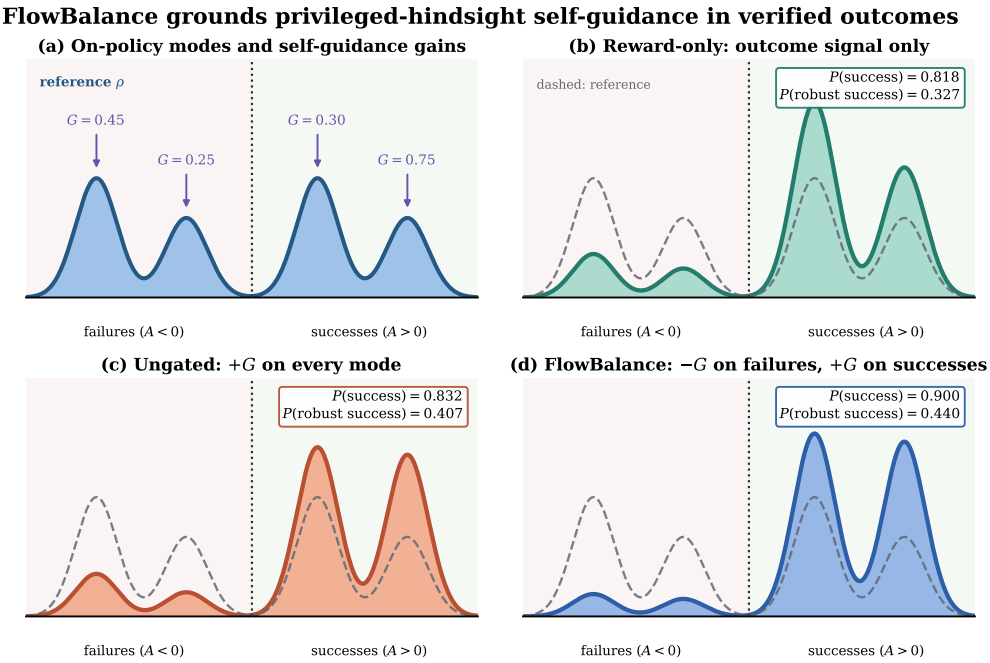


Figure 4: **Mechanism view of verifier-grounded self-guidance.** FlowBalance uses $+G$ on positive-advantage responses and $-G$ on negative-advantage responses. It retains useful guidance among successful modes without converting positive local confidence on failed responses into self-reinforcement.

G.1.2 Pairwise View: Exact Anti-Self-Confirmation in a Mixed Group

Figure 5 fixes positive self-guidance $G_+ > 0$ on a successful response and sweeps positive false-support $G_- > 0$ on a plausible failure. Proposition 4 gives the exact log-odds advantages

$$\begin{aligned} \operatorname{logit} p_{\text{FlowBalance}} - \operatorname{logit} p_{\text{reward}} &= \frac{\beta(G_+ + G_-)}{\tau} > 0, \\ \operatorname{logit} p_{\text{FlowBalance}} - \operatorname{logit} p_{\text{ungated}} &= \frac{2\beta G_-}{\tau} > 0. \end{aligned} \tag{69}$$

Thus FlowBalance dominates both baselines throughout this useful-but-imperfect quadrant. As false-positive support grows, ungated shaping becomes less reward-aligned, whereas outcome calibration becomes more selective. At $G_- = 0.5$, the exact target success probabilities are 0.894 for FlowBalance, 0.817 for reward-only shaping, and 0.807 for ungated shaping.

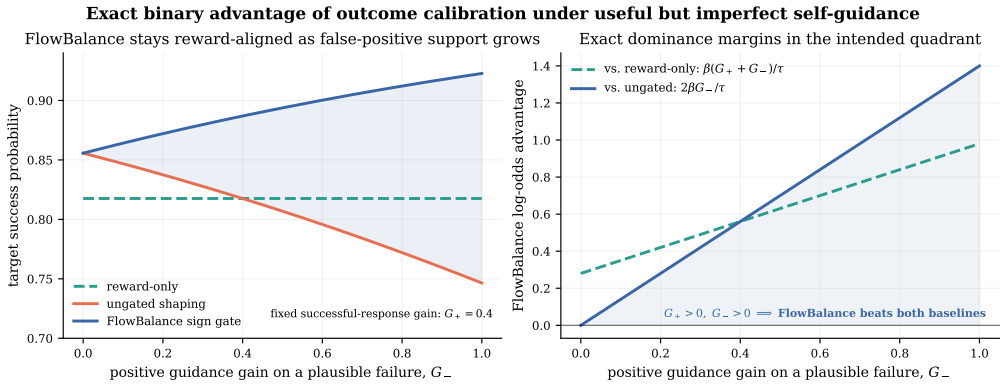


Figure 5: **Exact anti-self-confirmation advantage.** The left panel shows target success probability as positive self-guidance on a failed response increases. The right panel plots the closed-form probability-ratio margins in Eq. (69); no fitted decision boundary is used.

G.1.3 Boundary View: When Self-Guidance Helps and When It Stops Helping

Outcome calibration does not make arbitrary guidance safe. Figure 6 makes this boundary explicit on a five-mode response space. We interpolate the guidance vector as $G_q = qG_{\text{useful}} + (1-q)G_{\text{adv}}$, where $q = 1$ is useful but still false-positive on failures and $q = 0$ is adversarial to the verifier. We then sweep both q and β_G/τ. Above a reliability threshold, FlowBalance improves verified-success mass over reward-only shaping while retaining several successful modes. Below that threshold, stronger guidance can reduce verified success. The reverse-KL panel quantifies the accompanying movement from the reference. This diagnostic therefore supports a bounded claim: sign gating corrects false-positive confidence on failures, but it does not justify arbitrarily strong guidance that is systematically wrong on successful responses.

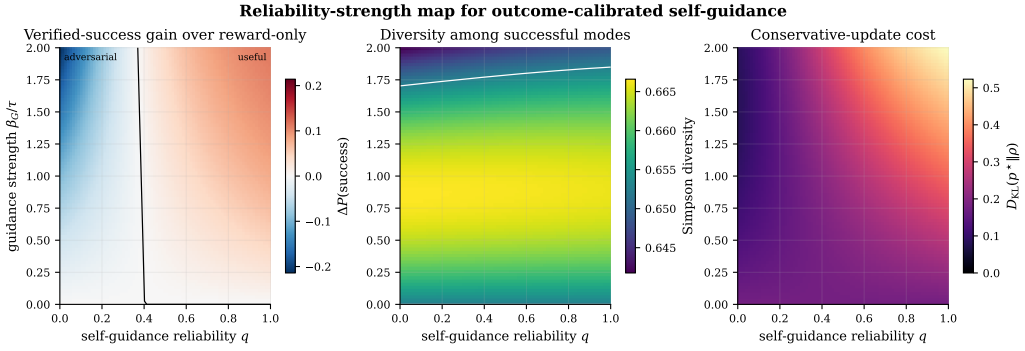


Figure 6: **Reliability–strength map for self-guidance.** Left: gain in verified-success mass over reward-only shaping. Middle: conditional Simpson diversity among successful modes; the white contour marks the reward-only level. Right: reverse KL to the reference. The reliability parameter is a synthetic interpolation, not an empirical calibration estimate.

G.1.4 Structural View: Conservative Change and All-Contrast Efficiency

Figure 7 verifies two distributional consequences of the FlowBalance formulation. First, the exponential-tilt path traces the minimum reverse-KL frontier as expected sign-gated energy increases. A constructed full-support fixed-data target is matched to the exact FlowBalance energy, but requires reverse KL 0.973 instead of 0.273, a $3.6\times$ larger displacement from the reference. Second, a local Gaussian calculation shows that profiling the group partition intercept retains all $N-1$ within-group contrast directions. At group size $N = 32$, its exact parameter risk is 2.92% of a one-contrast-per-group estimator, approximately $34\times$ lower risk. Together, these diagnostics

Table 5: Shared reporting fields for the reasoning experiments. Entries marked "matched" are held identical across methods within a backbone.

Field	Setting
Backbones	Qwen3-4B, Qwen3-8B
Training supervision	final-answer verifier; OPSD, RLSD, and FlowBalance also use privileged training solution/feedback
Compared methods	GRPO, OPSD, RLSD, FlowRL, FlowBalance
Rollout group size	matched across methods; pending release
Training horizon	step-180 final table; up to 400 steps for dynamics and response-length curves
Seeds	five seeds for every main-table entry
Maximum response length	matched across methods; pending release
Reference policy	fixed copy of the initial checkpoint
Training-time scorer	OPSD: fixed privileged teacher; RLSD/FlowBalance: frozen current-policy copy conditioned on c
FlowBalance default coefficients	$\eta_A = 15$, $\beta_G = 1$ for the completed default run
Evaluation metrics	AIME24 Pass@16; HMMT25/Minerva/MATH500/OlympiadBench Pass@1
Aggregate score	unweighted mean of the five benchmark means

explain why the self-improvement update is both conservative in where it moves probability mass and statistically efficient in how it uses each rollout group.

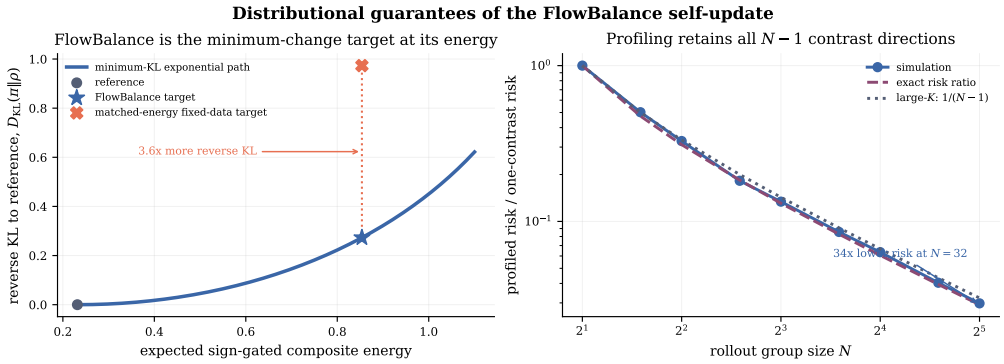


Figure 7: **Conservative change and all-contrast fitting.** Left: FlowBalance is the minimum reverse-KL target at its attained composite-energy level. Right: profiling one nuisance intercept per rollout group preserves all $N - 1$ contrast directions and yields the exact local Gaussian risk reduction.

H LLM Experiment Details

H.1 Reasoning Self-Improvement Setup and Protocol

Scope and fairness controls. The large-scale experiments are designed to compare policy-update objectives rather than infrastructure choices. Within each backbone, all methods use the same training prompts, rollout group size, maximum response length, verifier, optimizer schedule, checkpoint cadence, and evaluation script. For methods that use privileged training information (OPSD, RLSD, and FlowBalance), that information is available only to a frozen scoring path during training. Rollout generation and evaluation never include the solution or feedback field c.

Training signal construction. For every prompt in a minibatch, the frozen rollout snapshot samples a group of responses without seeing c. A rule-based verifier assigns a terminal correctness reward, from which we compute the stopped group-relative advantage in Eq. (2). FlowBalance then reuses the same frozen snapshot as the privileged-hindsight view by conditioning it on c and scoring the already sampled tokens. The self-guidance contribution is the clipped, length-averaged log-probability gain in

Table 6: Full ablation over verifier coefficient η_A on Qwen3-8B at step 180. The self-guidance coefficient is held fixed at the default value $\beta_G = 1$. Entries are percentages reported as mean $\pm$ sample standard deviation over five seeds. "Avg." averages the five benchmark means.

η_A	AIME24@16	HMMT25@1	Minerva@1	MATH500@1	Olympiad@1	Avg.
5	86.00 ± 1.49	30.00 ± 8.50	53.46 ± 1.24	93.32 ± 0.41	65.46 ± 0.69	65.65
10	85.33 ± 1.83	30.00 ± 3.33	52.43 ± 0.81	93.20 ± 0.32	66.11 ± 0.51	65.41
15	89.33 ± 1.49	34.67 ± 9.89	53.68 ± 0.78	93.52 ± 0.30	66.85 ± 0.46	67.61

Table 7: Full ablation over self-guidance coefficient β_G on Qwen3-8B at step 180. The verifier coefficient is held fixed at the default value $\eta_A = 15$. Entries are percentages reported as mean $\pm$ sample standard deviation over five seeds. "Avg." averages the five benchmark means.

β_G	AIME24@16	HMMT25@1	Minerva@1	MATH500@1	Olympiad@1	Avg.
1	89.33 ± 1.49	34.67 ± 9.89	53.68 ± 0.78	93.52 ± 0.30	66.85 ± 0.46	67.61
2	87.33 ± 1.49	32.00 ± 5.06	53.68 ± 1.40	93.64 ± 0.33	65.73 ± 1.16	66.48
3	86.00 ± 1.49	30.00 ± 4.08	53.46 ± 1.39	93.56 ± 0.55	66.74 ± 0.66	65.95

Eq. (6); no response is resampled from the guidance view and no guidance gradient is taken. The sign-gated trajectory energy is optimized through the trajectory-balance residual with the rollout-group partition estimate in Eq. (12).

Baseline implementations. GRPO uses the same verifier rewards and group-relative normalization as FlowBalance but optimizes the standard reward-policy objective rather than a normalized trajectory target. OPSD applies clipped forward-KL self-distillation from a fixed privileged teacher to the current policy's on-policy trajectories (Zhao et al., 2026a). RLSD combines verifier and teacher signals in its policy-optimization objective. FlowRL follows the original outcome-only trajectory-balance algorithm (Zhu et al., 2025). All baselines use the same sampled responses, answer verifier, and response-length cap whenever their objectives permit this sharing.

Evaluation protocol. For AIME24, each problem is decoded with 16 samples and is counted correct if any sample matches the final answer. For HMMT25, Minerva, MATH500, and OlympiadBench, we report Pass@1. The evaluator applies the same benchmark-specific final-answer extraction and normalization to every method. All main-table entries report the mean and sample standard deviation over five seeds at step 180. The accuracy-dynamics plots use matched evaluation checkpoints for FlowBalance and GRPO; the response-length panel compares the logged training trajectories of FlowBalance and OPSD. Solid curves denote smoothed trends and lighter curves denote the corresponding per-step measurements.

H.2 Grounding and Guidance Ablations

Coefficient sweeps. The ablations in Tables 2–3 are one-dimensional sweeps around the default setting. In the $\eta_A \in \{5, 10, 15\}$ sweep, β_G and all other training settings are held fixed. In the $\beta_G \in \{1, 2, 3\}$ sweep, η_A and all other settings are held fixed. Each sweep point is evaluated with the same five-benchmark average used in Table 1; all reported sweep points contain completed five-seed runs. Tables 6–7 provide the corresponding per-benchmark breakdown.

Coefficient ablations. Tables 6–7 expand the coefficient ablations in Section 5.2 to the same five benchmark columns used in the main result table. All rows use the Qwen3-8B backbone and report step-180 evaluation. AIME24 is evaluated with Pass@16, while HMMT25, Minerva, MATH500, and OlympiadBench are evaluated with Pass@1. The average is the unweighted mean of the five benchmark means.

H.3 Semantic Diversity Diagnostic

The diversity study in Section 5.3 evaluates semantic strategy variation on AIME24 using the step-180 checkpoints of GRPO, RLSD, and FlowBalance. We use seed 0 only, decode 16 complete responses for each of the 30 problems and each method, and submit all $30 \times 3 \times 16 = 1440$ full trajectories to a GPT-5.5 judge. No response is truncated or heuristically compressed before judging. These results are therefore a controlled diagnostic of strategy variation at one checkpoint and one seed, not a multi-seed stability estimate.

Two-stage LLM-judge protocol. The judge first performs strategy extraction independently for each full trajectory. The extracted fields include the primary solution strategy, central mathematical representation, main tools or theorems, a short reasoning outline, a compact strategy signature, and, for incorrect responses, the attempted failure mode. The instruction is to describe the strategy actually used by the trajectory, not to repair incorrect reasoning. Differences in variable names, wording, notation, response length, arithmetic detail, or the order of equivalent calculations are not counted as different strategies.

In the second stage, for each problem and method, the 16 extracted strategy summaries are anonymized and randomly ordered. The judge assigns every trajectory to exactly one semantic strategy cluster. During clustering, the judge does not see the source algorithm, checkpoint name, correctness label, or original response order. Correctness labels are applied only after clustering, and the reported diversity metric restricts the resulting cluster assignments to correct trajectories.

Metric. For a fixed problem and method, restrict the judged cluster assignments to correct trajectories and let $p_1, \ldots, p_K$ denote the proportions of those trajectories in the resulting semantic strategy clusters. We report correct-only Simpson strategy diversity,

$$D_{\text{Simpson}} = 1 - \sum_{k=1}^{K} p_k^2, \tag{70}$$

which is the probability that two randomly sampled correct trajectories use different semantic solution strategies. A problem is included in the aggregate only when it has at least two correct trajectories.

Scope. LLM-based clustering is inherently partly subjective, even with anonymization and a two-stage protocol. The main-text conclusion is therefore restricted to this AIME24 seed-0 diagnostic: FlowBalance exhibits higher judged semantic strategy diversity among correct trajectories at this checkpoint.

H.4 Prompt for Semantic Strategy Clustering

The LLM-judged diagnostic uses two prompts: one extracts a strategy summary from each trajectory, and the other clusters anonymized summaries within the same problem and method.

Prompt 1: Per-trajectory strategy extraction

Given the problem statement and one complete reasoning trajectory, describe the mathematical strategy actually attempted by the trajectory. Do not repair incorrect reasoning, replace the attempted method with a cleaner solution, or infer a method not present in the text.
Extract: (i) the primary solution strategy, (ii) the central mathematical representation, (iii) the main tools or theorems, (iv) a short reasoning outline, (v) a compact strategy signature, and (vi) for incorrect responses, the attempted failure mode.
Treat changes in notation, variable names, wording, response length, arithmetic detail, or the order of equivalent calculations as the same strategy. Treat substantively different mathematical objects or tools—for example, rectangular-box embedding versus Cayley–Menger determinant, coordinate versus synthetic geometry, generating functions versus direct counting, or roots-of-unity methods versus real polynomial methods—as different strategies.

Prompt 2: Within-problem strategy clustering

Given 16 anonymized strategy summaries for the same problem and method, assign every trajectory to exactly one semantic strategy cluster. The summaries are randomly ordered and do not reveal the source algorithm, checkpoint name, correctness label, or original response order.
Cluster by the core mathematical representation and main tools, not by surface wording, formatting, or response length. Incorrect but coherent attempts should still be clustered by their attempted strategy; correctness labels are applied only after clustering when computing the correct-only diversity metric.
Return the cluster assignment for each anonymous trajectory, a short cluster name, and a brief rationale for each cluster.

H.5 Why the FlowBalance Traces Are Interesting

The main text shows AIME24 Problem 23 as a compact illustration of the semantic distinctions used by the LLM-judged diversity diagnostic. Here we provide three additional FlowBalance cases from the same seed-0, step-180, 16-sample AIME24 run. Each case contains two representative correct trajectories whose final answers agree but whose central mathematical objects differ; the boxes mark the strategy-defining reasoning actions, following the format of Table 4.

Together with the AIME24 Problem 23 example in the main text, these cases show that FlowBalance's additional correct trajectories are not merely longer, more verbose, or differently worded variants of a standard template. We regard a trace as *interesting* when the boxed steps change the mathematical object or criterion that drives the solution: for example, replacing a local algebraic test with a global envelope argument, replacing a three-dimensional geometry problem with a two-dimensional cross-section reduction, or replacing direct elimination with a parameterized conic argument. In other words, the diversity signal is intended to capture a semantic change in *how* the problem is solved, not a superficial change in phrasing.

On Problem 5, FlowBalance finds both a global envelope view of the unit-intercept line family and a local multiplicity view. The first identifies the special point as the tangency point on the astroid $x^{2/3} + y^{2/3} = 1$, while the second detects uniqueness by forcing the known line AB to be a double root of a one-variable trigonometric equation. On Problem 15, one FlowBalance trajectory reduces the three-dimensional torus-sphere contact to two circle tangencies in a meridian plane. Another keeps the problem as an implicit surface calculation and derives the contact radii from collinear normals in cylindrical coordinates. On Problem 27, FlowBalance again covers two different representations of the same constraint: direct coordinate elimination in squared variables, and a secant-tangent hyperbola parameterization that turns the rhombus condition into the trigonometric product $\sin\theta \sin\phi = -5/6$. These examples mirror the main-text box-embedding case and help explain why the FlowBalance traces are meaningful: the model is spreading probability mass across genuinely different correct derivations, not just stylistic rewrites of one template.

How to read the traces. The boxed lines mark only the reasoning actions that define the route; standard algebra, routine simplifications, and repeated bookkeeping are suppressed with "$\cdots$". The goal is not to claim that every token sequence differs globally, but to show that the high-level derivational backbone differs in a way that would matter to a human reader choosing between strategies.

Table 8: **Additional case study on AIME24 Problem 5.** Two correct FlowBalance trajectories solve the same unit-intercept-segment problem using different criteria for uniqueness.

	Content (boxed = reasoning actions; "· · · " = omitted)
Question	Let $O = (0, 0)$, $A = (1/2, 0)$, and $B = (0, \sqrt{3}/2)$. Among all unit-length segments with one endpoint on the positive x-axis and one endpoint on the positive y-axis, there is a unique interior point C of AB that lies on no other such segment. If $OC^2 = p/q$ in lowest terms, find $p + q$.
Route 1	**Envelope / astroid route.** "· · · "
	$F(x, y, u) = x/u + y/\sqrt{1 - u^2} - 1$ for the unit-intercept line family.
	Solve $F = 0$ and $F_u = 0$ to obtain the envelope.
	$x = u^3$, $y = (1 - u^2)^{3/2}$, the first-quadrant astroid.
	AB corresponds to $u = 1/2$, so $C = (1/8, 3\sqrt{3}/8)$.
	$OC^2 = 7/16$, hence $p + q = 23$.
Route 2	**Multiple-root uniqueness route.** "· · · "
	Write an interior point as $C(t) = ((1 - t)/2, \sqrt{3}t/2)$.
	A unit-intercept line through $C(t)$ satisfies $(1 - t)/(2\cos\theta) + \sqrt{3}t/(2\sin\theta) = 1$.
	The segment AB is the known solution $\theta = \pi/3$.
	Uniqueness at the boundary forces this solution to be a double root.
	$f'(\pi/3) = \sqrt{3}(1 - t) - \sqrt{3}t/3 = 0$, so $t = 3/4$.
	$C = (1/8, 3\sqrt{3}/8)$ and $p + q = 23$.

Table 9: **Additional case study on AIME24 Problem 15.** Two correct FlowBalance trajectories use either a two-dimensional meridian section or a three-dimensional implicit-normal condition.

	Content (boxed = reasoning actions; "· · · " = omitted)
Question	A torus is formed by rotating a circle of radius 3 about an axis in its plane at distance 6 from the circle's center. A sphere of radius 11 is tangent to the torus in two rotationally symmetric positions, with corresponding contact-circle radii r_i and r_o. If $r_i - r_o = m/n$ in lowest terms, find $m + n$.
Route 1	**Meridian-section circle-tangency route.** "· · · "
	Take a plane through the rotation axis; the torus cross-section is a circle of radius 3 centered at $(6, 0)$.
	The sphere cross-section is a circle of radius 11 centered at $(0, h)$.
	The two tangencies have center distance $11 - 3 = 8$ or $11 + 3 = 14$.
	The contact-circle radius is the horizontal coordinate of the tangency point.
	$r_i = 11 \cdot 6/8 = 33/4$ and $r_o = 11 \cdot 6/14 = 33/7$.
	$r_i - r_o = 99/28$, so $m + n = 127$.
Route 2	**Implicit-surface normal route.** "· · · "
	Use cylindrical coordinates (ρ, z).
	Torus: $(\rho - 6)^2 + z^2 = 9$; sphere: $\rho^2 + (z - h)^2 = 121$.
	Tangency means normals are collinear: $(\rho - 6, z) = \lambda(\rho, z - h)$.
	Eliminate z, h to get $\rho/\lvert\rho - 6\rvert = 11/3$.
	$\rho > 6$ gives $\rho = 33/4$; $\rho < 6$ gives $\rho = 33/7$.
	$m + n = 127$.

Table 10: **Additional case study on AIME24 Problem 27.** Two correct FlowBalance trajectories optimize the same diagonal length using algebraic elimination or a trigonometric hyperbola parameterization.

	Content (boxed = reasoning actions; "· · · " = omitted)
Question	Points A, B, C, D lie on the hyperbola $x^2/20 - y^2/24 = 1$ and form a rhombus whose diagonals intersect at the origin. Find the greatest real number that is strictly less than BD^2 for every such rhombus.
Route 1	**Coordinate-elimination route.** "· · · "
	Write opposite vertices as $(a, b), (-a, -b)$ and $(c, d), (-c, -d)$.
	A centered parallelogram is a rhombus iff its diagonals are perpendicular, so $ac + bd = 0$.
	Set $x = a^2$ and use $b^2 = 6x/5 - 24$.
	From perpendicularity, $c^2 = (b^2/a^2)d^2$.
	The hyperbola equation for B gives $d^2 = 600x/(11x - 720)$.
	$BD^2 = 4(c^2 + d^2) = 480 + 288000/(11x - 720) > 480$, with limit 480 from above.
Route 2	$\sec$–$\tan$ **parameterization route.** "· · · "
	Parameterize the hyperbola as $(2\sqrt{5}\sec\theta, 2\sqrt{6}\tan\theta)$.
	Take A with parameter θ and B with parameter ϕ.
	Perpendicularity gives $20\sec\theta\sec\phi + 24\tan\theta\tan\phi = 0$.
	Equivalently, $\sin\theta\sin\phi = -5/6$.
	Since $\lvert\sin\theta\rvert < 1$, we have $\lvert\sin\phi\rvert > 5/6$, hence $\tan^2\phi > 25/11$.
	$BD^2 = 80 + 176\tan^2\phi > 480$, with limit 480 attainable only at infinity.