DISTRIBUTIONAL POSITION SHIFT FOR REASONING: SHARED-FUTURE POLAR SAMPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning systems can solve problems that their base models cannot solve with standard decoding. Reinforcement learning (RL) and multi-step inference have emerged as powerful approaches for improving the reasoning capabilities of large language models by shaping their policy distributions and exploiting information from multi-step trajectories. More recently, power sampling has been proposed as a training-free approach to improve reasoning by sharpening the sequence distribution during generation. However, local sharpening can only amplify the model's existing preferences: it preserves the relative ranking of candidate segments and therefore cannot promote a lower-ranked candidate that may lead to better reasoning when followed by later steps. This motivates a different question: can sampling use evidence from future reasoning to revise the model's local candidate preferences? In this work, we introduce *Shared-Future Polar Sampling* (SFPS), a training-free approach that enables *position shifts* in the candidate distribution by using evidence from subsequent reasoning to re-rank candidate segments. Inspired by RL's use of rollout comparisons, SFPS uses short continuations to estimate how future reasoning supports each candidate segment. These continuations are shared across candidates, allowing their future support to be compared efficiently. Our polar sampling rule uses both the magnitude and direction of this support to update the candidate distribution. SFPS outperforms MCMC Power Sampling on all ten model–task pairs and GRPO on eight. Across four Qwen2.5-Math-7B tasks, it uses 20.13% of Power's generated tokens and 9.31% of its inference time on average.

1 INTRODUCTION

Why can reasoning systems solve problems that ordinary decoding of their base models misses? Two major approaches are *reinforcement learning (RL)* and *multi-step inference*. RL improves a policy using feedback from sampled rollouts (Shao et al., 2024; DeepSeek-AI, 2025), while multi-step inference uses intermediate computation, search, or interaction before answering (Wei et al., 2022; Yao et al., 2023a;b). Despite their differences, both leverage information beyond the model's local prediction to improve subsequent decisions. RL incorporates this information into the learned policy, while multi-step inference obtains it through additional computation at inference time. These methods can also be combined, e.g., DeepSeek-R1-Zero develops long reasoning chains, self-verification, and reflection through RL (DeepSeek-AI, 2025). More related work is discussed in Appendix A.

Can we obtain similar benefits from sampling alone, without training? Power sampling takes a step in this direction by sharpening the base model's sequence distribution toward high-likelihood responses (Karan & Du, 2025). However, **local sharpening amplifies existing preferences without changing the relative ordering of local candidates**. At a given decision point, a lower-ranked candidate cannot overtake a higher-ranked one simply because the local distribution is sharpened. Yet reasoning often requires exactly such a change: a candidate that appears less promising locally may become preferable once its subsequent reasoning is taken into account. We call this a *position shift* in the candidate distribution. Improving sampling therefore requires not making the model more confident in its current preference, but using future reasoning to determine when that preference should change.

We introduce **Shared-Future Polar Sampling** (SFPS), a training-free approach that uses future reasoning to induce such *position shifts*. Inspired by RL's comparison of rollout outcomes, SFPS compares candidate openings through a set of shared short continuations. Because the same contin-

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.

uations are evaluated after each candidate, their future evidence can be compared directly without generating separate complete responses. Within each block, SFPS extends these shared continuations token by token, accumulating evidence about which candidate is better supported by subsequent reasoning. A polar sampling rule then converts this evidence into a distributional update with two complementary effects: a radial component sharpens existing preferences, while a residual component redirects probability mass and can change the candidate ordering. Thus, unlike local sharpening, SFPS allows evidence from future reasoning to shift the model's local preference when warranted.

From RL rollout comparisons to a shared evidence matrix. The key challenge is to extract comparable future evidence for multiple candidate openings without the cost of running independent long rollouts. RL addresses a related problem by comparing the outcomes of different sampled trajectories. SFPS instead uses the base model itself as the source of evidence: each candidate is evaluated against the same set of short continuations, without training or reward labels. This shared structure makes the future support of different candidates directly comparable. We organize these comparisons into a candidate–future matrix, whose entries measure the support provided by each shared continuation for each candidate opening. Our analysis separates local likelihood from future evidence: a challenger can overtake the current preference when its relative future advantage exceeds its local likelihood deficit. The row totals of the resulting matrix determine the preferred opening, which is then committed together with a continuation from the corresponding row.

From shared evidence to polar sampling. The shared matrix provides a way to compare candidates, but the evidence must still be accumulated during generation and translated into a distributional update. This is where multi-step inference motivates our stepwise construction. Within each block, SFPS extends the same shared continuations token by token while keeping the candidate openings fixed. Each newly sampled token contributes additional evidence about the candidates. We decompose this update into a *radial component*, which reinforces the current preference, and a *residual component*, which redistributes support across candidates and can induce a position shift. These two effects can occur simultaneously. Importantly, scalar evidence is insufficient to represent how future support is allocated across candidates. Our polar sampling rule therefore retains the full contribution vector, using both its magnitude and direction to sample the next shared token. The resulting weighted updates form the columns of the candidate–future matrix used for the final block-level comparison.

Contributions and results. Our analysis establishes a condition under which future evidence induces a distributional position shift and connects this condition to a shared evidence matrix and a fixed-reference polar sampler. We further prove consistency of the resulting finite-horizon comparison. Empirically, SFPS outperforms MCMC Power Sampling on all ten evaluated model–task pairs and outperforms GRPO on eight (Table 1). Across the four tasks on the 7B model, SFPS uses 20.13% of Power's generated tokens and 9.31% of its measured inference time on average (Table 2).

2 Distributional position shift: what must change?

We consider a base language model with policy function p. Given a prompt and a sequence of previously generated tokens, together forming the prefix z, we consider m candidate segments $(a_1, \ldots, a_m)$ with positive conditional probabilities $p(a_i \mid z)$. Let $k = \arg\max_{i \in \{1,\ldots,m\}} p(a_i \mid z)$ denote the *leader candidate*, i.e., the candidate with the highest immediate likelihood. Our goal is to understand how to select, among these m candidates, a segment that leads to a better final response. In particular, the candidate with the highest immediate likelihood may not lead to the highest final reward. We refer to this phenomenon as a *position shift in the candidate distribution*: the leader candidate under the model's immediate likelihood is not necessarily the optimal candidate when future evidence is taken into account. Figure 1 contrasts these changes.

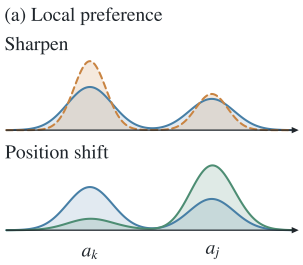


Figure 1: Local sharpening (orange) preserves the blue leader; future evidence (green) changes it.

Research question: what must change beyond sharpening?

When can evidence from rollouts make a locally disfavored continuation the preferred choice?

To address this question, we assume that each candidate a_i is associated with a score V_i that summarizes future evidence as generation continues after a_i. The score V_i can be estimated based on the reward function or model likelihoods. We will explain how to estimate V_i in the next section. Furthermore, we define the score s_i and the normalized candidate probability π_i as follows:

$$s_i = \underbrace{\lambda \log p(a_i \mid z)}_{\text{local preference}} + \underbrace{V_i}_{\text{future evidence}}, \qquad \pi_i = \frac{e^{s_i}}{\sum_{\ell=1}^{m} e^{s_\ell}}, \qquad \lambda > 0. \tag{1}$$

Proposition 2.1 (Position shift is a local–future competition)**.** *If the current likelihood favors a_k over a_j, the future-sensitive rule ranks a_j ahead of a_k exactly if*

$$V_j - V_k > \lambda \log \frac{p(a_k \mid z)}{p(a_j \mid z)}. \tag{2}$$

The original leader ceases to maximize when at least one challenger[a] satisfies this inequality.

[a] Here, "challenger" means a candidate segment that is currently not the top choice but could overtake the current leader once future evidence is taken into account.

The left-hand side captures the challenger's future-evidence advantage, while the right-hand side captures its current likelihood deficit. A common V_i cancels across candidates and therefore leaves the ranking induced by the current likelihood $p(a_i \mid z)$ unchanged. Increasing λ increases the effective likelihood deficit, making it harder for future evidence to reverse this ranking. Therefore, changing the strength of the local preference alone cannot cause a position shift in the candidate distribution. A shift requires candidate-specific future evidence V_i. Note that the condition in Proposition 2.1 characterizes when the preferred choice changes; it does not establish whether the resulting choice is correct. More discussion on this condition can be found in Appendices B.2 and B.9.

This formulation specifies what we need to compare across candidates for token generation, but not how to estimate the future evidence V_i. We next use the rollout-based comparison idea from RL to estimate future evidence directly from the base model, without training.

3 From RL rollouts to a shared comparison matrix

In this section, we develop a decoding strategy for generation inspired by reinforcement learning (RL). In particular, GRPO does not rely solely on the immediate likelihood of tokens; instead, it uses multiple rollouts to compare their rewards and update the policy (Shao et al., 2024). Motivated by this idea, our method also looks beyond immediate token likelihoods by estimating future evidence V_i from a small shared group of short candidate sequences.

3.1 What to read from rollouts without training

The RL reference. Let y denote the rollout after prefix za (with a being a candidate segment) and $R(zay)$ denote the reward associated with sequence zay. For an unrestricted policy, RL maximizes expected terminal reward minus $\beta\,\mathrm{KL}(\pi\|p)$ giving optimal policy as $\pi^*(ay \mid z) \propto p(ay \mid z)e^{R(zay)/\beta}$ (Levine, 2018; Rafailov et al., 2023). Summing over all possible future y gives

$$\pi_i^{\mathrm{RL}}(a_i|z) \propto \sum_y p(a_i \mid z)p(y \mid za_i)e^{R(za_iy)/\beta} = p(a_i \mid z)\mathbb{E}_{y\sim p(\cdot|za_i)}\left[e^{R(za_iy)/\beta}\right]. \tag{3}$$

Eq. (3) is a special case of Eq. (1) with $\lambda = 1$ and V_i given by the logarithm of the conditional expectation of the exponentiated reward. This represents an idealized target rather than the exact policy obtained after finite GRPO training, as normalization and regularization in the training objective can affect the resulting policy (Vojnovic & Yun, 2025). Appendix B provides the policy-gradient and variational derivations and clarifies their distinction from a local expected-reward update.

A computable signal from the base model. If we only take into account the probability $p(a_iy \mid z)$ for sampling, then by applying factorization $p(a_iy \mid z) = p(a_i \mid z)p(y \mid za_i)$, we can write the

candidate probability generation π_i and s_i in Eq. (1) as

$$s_i = 1 \cdot \log p(a_i|z) + \log\left(\sum_y p(y \mid za_i)\right), \quad \pi_i \propto p(a_i \mid z)\sum_y p(y \mid za_i).$$

More generally, Eq. (1) gives

$$\pi_i \propto p(a_i \mid z)^\lambda B_i, \qquad B_i := e^{V_i}. \tag{4}$$

Here B_i can represent reward-based evidence from Eq. (3) ($\lambda = 1$), or likelihood-based evidence from sequence-level power sampling (Karan & Du, 2025). For our method, we set $\lambda = \alpha > 1$ and define $B_i = \sum_y p(y \mid za_i)^\alpha$ over a fixed short continuation horizon. The local likelihood weight $p(a_i \mid z)^\alpha$ sharpens the candidate probability while B_i can change their candidate ranking. Eq. (4) forms the core of our proposed sampling method, SFPS. Since our method will work with partial continuation y instead of a complete response for token generation, we are not including the reward function in our sampling procedure. In particular, in each iteration of SFPS, we will have a set of candidate and rely on partial continuations/rollout to compare them before sampling and generation. Unlike an RL reward, B_i represents model support rather than a correctness label. Appendix B gives its relation to conditional Rényi entropy (van Erven & Harremoës, 2014).

3.2 Rollout comparison matrix for sampling

In this section, we show how Eq. (4) can be used for sampling and generation. For SFPS, calculating B_i requires summing over all continuations y, which is computationally expensive. Directly sampling multi-token candidates according to $p(a_i \mid z)^\alpha$ is also expensive. We therefore sample a_i from the language model with temperature, correct the sampling probabilities by importance weighting, and share a few partial rollouts[1] across all candidates. Algorithm 1 summarizes the purposed SFPS decoding algorithm. In this and the next section, we explain the idea behind SFPS.

The SFPS algorithm has three main parts. Given a pre-fix z:

1. Sample K segment candidates $a_1, \ldots, a_K$ of length L. Merging exact duplicates gives $m \le K$ distinct candidates $a_1, \ldots, a_m$.
2. Generate n shared continuation $y_1, \ldots, y_n$, each with at most H tokens. The continuations are generated token by token. We denote the probability of generating y_i given $z, a_1, \ldots, a_m$ by $G(y_i)$.
3. Select a candidate a and a continuation y, update the pre-fix as $z \leftarrow zay$, and return to Step 1.

We discuss Step 2, including the polar construction of G, in Section 4.3. Here, we focus on how candidate segments are sampled and corrected in Step 1 and how shared continuations are used to select the next segment in Step 3.

For Step 1, we first draw K candidate segments from the base language model with sampling probability $q(a \mid z) = \prod_{t=1}^{|a|} \frac{p(a_t|za_{<t})^{1/\tau}}{\sum_x p(x|za_{<t})^{1/\tau}}$, where τ is the temperature. We then keep only the distinct candidate $a_1, \ldots, a_m$ and let c_i denote the frequency of a_i, with $\sum_{i=1}^m c_i = K$. We define $\widetilde{b}_i = c_i \frac{p(a_i|z)^\alpha}{q(a_i|z)}$, $A = \sum_i \widetilde{b}_i$, and $b_i = \widetilde{b}_i / A$, where $b_i > 0$ and $\sum_i b_i = 1$. Lemma D.2 shows that as $K \to \infty$, $b_i \to p(a_i \mid z)^\alpha / \sum_a p(a \mid z)^\alpha$ almost surely, where the sum is over all possible candidate segments at the current prefix. We use b_i in Steps 2 and 3 of our algorithm.

In Step 3, our goal is to select one of the m candidate segments together with one of the n shared continuations. To do so, we define the $m \times n$ contribution matrix $\mathbf{M}$ as follows,

$$\mathbf{M} = \begin{bmatrix} \frac{b_1 p_1(y_1)^\alpha}{G(y_1)} & \cdots & \frac{b_1 p_1(y_n)^\alpha}{G(y_n)} \\ \vdots & \ddots & \vdots \\ \frac{b_m p_m(y_1)^\alpha}{G(y_1)} & \cdots & \frac{b_m p_m(y_n)^\alpha}{G(y_n)} \end{bmatrix}, \qquad \widehat{W}_i = \frac{1}{n}\sum_j M_{ij}, \tag{5}$$

[1] We use partial rollout and continuation interchangeably.

where $p_i(y) = p(y \mid za_i)$. Row i corresponds to candidate a_i and column j corresponds to continuation y_j. Dividing by $G(y_j)$ corrects for how often the sampler generates continuation y_j. For a fixed candidate pool, this importance-sampling correction gives $\mathbb{E}[\widehat{W}_i] = b_i B_i$. After normalization, $b_i B_i$ converges to the comparison distribution in Eq. (4) as $K \to \infty$ (Lemma D.2). Thus, $\widehat{W}_i$ estimates the unnormalized support of candidate a_i.

Within each column, the common probability of generation y_j cancels out, giving a direct comparison:

$$\frac{M_{ij}}{M_{kj}} = \frac{b_i}{b_k} \left[\frac{p_i(y_j)}{p_k(y_j)} \right]^{\alpha}. \tag{6}$$

This ratio accounts for both the candidate probabilities and the probability of the same continuation under each candidate. Thus, a shared continuation can provide different amounts of evidence for different candidate openings. Since $\widehat{W}_i$ estimates $b_i B_i$, decoder picks the best-supported candidate:

$$i^* = \arg\max_i \widehat{W}_i. \tag{7}$$

To reuse the generated text, the decoder then picks the continuation with the largest contribution in the row of candidate a_{i^*},

$$j^* = \arg\max_j M_{i^* j}, \qquad z \leftarrow z\, a_{i^*} y_{j^*}. \tag{8}$$

The matrix readout has the same local–future structure as Eq. (1): its local term is $\log b_i$, while its estimated future term is $\log(\widehat{W}_i / b_i)$. Their sum is $\log \widehat{W}_i$.

4 From multi-step updates to polar sampling

In this section, we generate a continuation y of length H token by token. At each step, we select the next token by jointly considering the probability of each candidate segment, the probability of the prefix generated so far, and the vocabulary distribution. We provide a geometry interpretation of token generation and explain how each column of $\mathbf{M}$ is produced at the end of generation of y.

4.1 Compare the next token across all candidates

To generate continuation y, all the candidate segment is contributing to the generation process. This generation happens token by token. Let h denote the tokens generated so far, and w_i denote the running weight of sequence $a_i h$, initially (when h is empty) set to $w_i = b_i$. We denote D as the vocabulary size, and $x_1, \ldots, x_D$ its tokens. For any token x, $p_i(x \mid h) = p(x \mid za_i h)$ is its next-token probability after opening i and text h. The contributions of tokens form an $m \times D$ matrix $\mathbf{F}(h)$:

$$\mathbf{F}(h) = \begin{bmatrix} w_1 p_1(x_1 \mid h)^{\alpha} & \cdots & w_1 p_1(x_D \mid h)^{\alpha} \\ \vdots & \ddots & \vdots \\ w_m p_m(x_1 \mid h)^{\alpha} & \cdots & w_m p_m(x_D \mid h)^{\alpha} \end{bmatrix}, \tag{9}$$

where each column corresponds to one token. We use $f(x)$ to denote the column associated with token x, and $f_i(x)$ to denote the entry of $f(x)$ associated with candidate i. To quantify the aggregate contribution of token x across all candidates, we use the weighted norm $r(x) = \|f(x)\|_b = \sqrt{\sum_{i=1}^{m} \frac{f_i(x)^2}{b_i}}$, and define the shared token distribution $g(x \mid h) = \frac{r(x)}{\sum_{\ell=1}^{D} r(x_\ell)}$. Thus, $g(x \mid h)$ assigns higher probability to tokens with larger aggregate contributions across the candidate segments. In particular, a token receives more probability when its contribution is large for one or more candidates, while the weighting by $1/b_i$ accounts for the fixed initial weight of each candidate. We will provide a geometrical interpretation of this sampling in Sections 4.2 and 4.3.

We sample one token x from the shared distribution $g(x \mid h)$ and append it to every candidate context. Since x is sampled from $g(x \mid h)$ while $f_i(x)$ is an unnormalized contribution for candidate i, we use an importance-weighting correction when updating the candidate weights: $w_i' = \frac{f_i(x)}{g(x|h)}$. The division by $g(x \mid h)$ compensates for probability with which x was sampled, thus, for each candidate,

$\mathbb{E}_{x\sim g(\cdot|h)}\left[\frac{f_i(x)}{g(x|h)}\right] = \sum_x f_i(x)$. After the token is generated, we set $w_i \leftarrow w_i'$ and use these updated weights in the next generation round. Moreover, because the same factor $g(x \mid h)$ appears in every candidate's update, it cancels when comparing any two candidates:

$$\frac{w_i'}{w_k'} = \frac{w_i}{w_k}\left(\frac{p_i(x \mid h)}{p_k(x \mid h)}\right)^{\alpha}.$$

Thus, $g(x \mid h)$ determines which token is observed, while the relative likelihood of the observed token under different candidates determines how that observation changes their relative weights.

If $p_i(x \mid h) = p_k(x \mid h)$ for every x and every pair of candidates i, k, the rows of $\mathbf{F}(h)$ are proportional. In this case, every candidate weight is multiplied by the same factor after any sampled token, so their relative weights cannot change. Conversely, when the row weights are positive, rank one is equivalent to the candidates having identical next-token distributions (Appendix E.4). Thus, nonproportional rows are necessary for an observation to create a new relative preference among candidates, although they do not by guarantee the resulting preference is useful after aggregation.

4.2 Candidate Preference: Sharpening and Redirection

The normalized running weights $w_i/\sum_k w_k$ define a distribution over the candidate sequences $a_i h$. Adding one token can strengthen the current preference, redirect it toward other candidates, or do both. To separate these effects, subtract the mean log-weight: $v_i = \log w_i - \frac{1}{m}\sum_k \log w_k$, and define v' from w' in the same way. This standard coordinate removes normalization while preserving every pairwise log-ratio (Aitchison, 1982). The increment is the centered vector of $\alpha \log p_i(x \mid h)$. For $v \neq 0$, let c be its radial scaling coefficient and d the residual vector. Using the dot product $\langle \cdot, \cdot \rangle$ and Euclidean norm $\| \cdot \|_2$, we have,

$$v' = cv + d, \qquad c = \frac{\langle v', v\rangle}{\|v\|_2^2}, \qquad d \perp v. \tag{10}$$

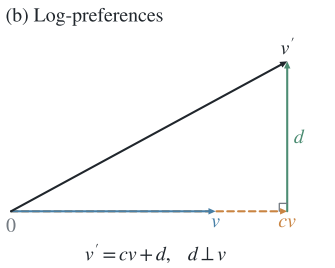


Figure 2: Blue v extends to orange cv; green d redirects the black update v'.

Here $d \perp v$ means $\langle d, v\rangle = 0$. The parallel component follows the existing preference. For $c > 1$ it is *sharpening*; for $0 < c < 1$ it flattens the same ordering. The residual d is the *residual shift component*: it changes the preference pattern and can make another candidate lead. These two components can coexist in one evidence update (Figure 2). A residual is not yet a realized position shift; it must be strong enough in the relevant direction.

Proposition 4.1 (Which component can overturn the current preference?)**.** *Let k lead at v and write $v' = cv + d$. A challenger j overtakes k exactly when*

$$d_j - d_k > c(v_k - v_j). \tag{11}$$

For $c > 0$, a zero residual preserves every ordering. A nonzero residual changes the leader only if it crosses one of these boundaries.

Corollary 4.2 (A minimum residual is needed for a distributional shift)**.** *If k uniquely leads at v, $c > 0$, and the leader changes in $v' = cv + d$, then*

$$\|d\|_2 > \frac{c}{\sqrt{2}} \min_{j\neq k}(v_k - v_j). \tag{12}$$

A large residual is necessary, but not sufficient for a position shift: it must also favor a competing candidate. Appendix C.2 proves this bound and provides a more detailed characterization of the corresponding boundary. Note that the magnitude $\|d\|_2$ alone does not identify which candidates the future evidence favors. We therefore retain the full vector $f(x)$ and use its polar magnitude and direction to sample tokens, as we explain in the next part.

4.3 Token Selection by Contribution Magnitude and Direction

For a token x, $f_i(x)$ is the contribution of x to candidate i, $\mu(x) = \sum_i f_i(x)$ is its total contribution, and $\|f(x)\|_b^2 = \sum_i f_i(x)^2/b_i$ measures the strength of the contribution relative to the initial

candidate weights b_i. Moreover, $\mu(x)b$ is a vector that distributes the total contribution according to the initial candidate weights, and $e(x) = f(x) - \mu(x)b$ is a residual vector that compares the current contribution of x to each candidate with the contribution expected under the initial candidate allocation. In particular, $e_i(x) > 0$ indicates that candidate i receives more contribution from token x than its initial allocation would suggest, whereas $e_i(x) < 0$ indicates that it receives less. Since $\mu(x)b \perp_b e(x)$ under the b-weighted inner product and $\|b\|_b = 1$, the following holds by Pythagorean theorem,

$$r(x)^2 = \mu(x)^2 + \|e(x)\|_b^2, \quad r(x) = \mu(x)\sec\theta(x), \tag{13}$$

where $r(x) = \|f(x)\|_b$ and $\theta(x)$ is the angle between the contribution vector $f(x)$ and the initial candidate allocation b, $\cos\theta(x) = \mu(x)/r(x)$, and $\sec\theta = 1/\cos\theta$.The angle measures departure of the current contribution from the initial allocation (Figure 3); the full vector $f(x)$ retains which candidates receive the contribution. $r(x)$ and $\theta(x)$ are the polar *contribution* coordinates, whereas Eq. (10) describes log-preferences. $(r(x), \theta(x))$ retain evidence size and allocation, while d in Eq. (10) captures redirection that can change the leader.

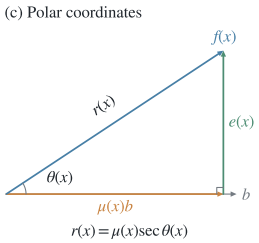


Figure 3: Blue $f(x)$ splits into orange $\mu(x)b$ and green $e(x)$ in the b-weighted geometry; its radius sets the sampling weight.

We sample in proportion to this radius, with u ranging over all vocabulary tokens:

$$g(x \mid h) = \frac{r(x)}{\sum_u r(u)}, \qquad r(x) = \sqrt{\sum_i \frac{w_i^2 p_i(x \mid h)^{2\alpha}}{b_i}} = \mu(x)\sec\theta(x). \tag{14}$$

The rule favors tokens with strong contributions, especially when those contributions differ across candidates. The polar coordinates (r, θ) determine g, which samples x; the resulting weight update w' gives v', whose residual d in $v' = cv + d$ captures the resulting redirection of candidate preferences.

Proposition 4.3 (An exact identity for the polar update)**.** *Fix the token matrix and sample x using Eq.* (14)*. The updated vector $w' = f(x)/g(x \mid h)$ satisfies*

$$\mathbb{E}[w'] = \sum_u f(u), \qquad \|w'\|_b = \sum_u r(u). \tag{15}$$

Its weighted radius is independent of the sampled token; its candidate-support direction is retained.

For this fixed-reference vector measure, the rule also minimizes the one-step mean-squared estimation error among supported token proposals (Appendix E.3). However, this rule need not be globally optimal in terms of complete reasoning trajectories.

4.4 The token updates recover the future matrix

Repeating the update along a sampled future y yields

$$w_i = b_i \frac{p_i(y)^\alpha}{\prod_t g(y_t \mid y_{<t})} = b_i \frac{p_i(y)^\alpha}{G(y)}. \tag{16}$$

For the jth continuation, these running weights are exactly column j of Eq. (5). Thus, the token-generation loop constructs the matrix as it generates the continuation, without requiring a separate scoring step afterward. Once the columns are complete, their row sums enter Eq. (7). This connects the polar sampler to the decision criterion introduced in Section 2.

5 The complete training-free decoder

Given prompt z, Algorithm 1 generates a response by jointly selecting a candidate and continuing iteratively. In each iteration, it proposes several candidates, and initializes the matrix rows (Section 3), extends a small group of shared continuations using the polar token rule (Section 4.3), calculates the summation of the resulting columns, and selects a row (candidate), and commit its opening–continuation pair. This effectively performs the local–future comparison of Section 2. Algorithm implementation details are provided in Appendix H.

Algorithm 1 Shared-Future Polar Sampling (SFPS)

Require: Base model p, prompt z, proposal q; α, K, n, L, H, output cap

1: **while** not terminated and below the output cap **do**
2: $a^{(1)}, \dots, a^{(K)} \overset{\text{iid}}{\sim} q(\cdot \mid z)$ ▷ at most L tokens
3: Merge duplicates; initialize b by Eq. (D.2).
4: $y_j \leftarrow \varnothing, \quad w_{ij} \leftarrow b_i \qquad (\forall i, j)$
5: **for** $t = 1, \dots, H$ **do**
6: For unfinished futures j with live rows, form $\mathbf{F}_j$ and g_j ▷ Eqs. (9), (14)
7: Sample their tokens $x_j \sim g_j$ with fresh randomized stratification ▷ Eq. (G.3)
8: $w_{ij} \leftarrow w_{ij}\, p_i(x_j \mid y_j)^{\alpha} / g_j(x_j)$ for live rows i
9: Append each sampled token x_j to its continuation y_j
10: **end for**
11: $M_{ij} \leftarrow w_{ij}, \quad \widehat{W}_i \leftarrow n^{-1} \sum_j M_{ij}$ ▷ Eq. (5)
12: $i^* \leftarrow \arg\max_i \widehat{W}_i, \quad j^* \leftarrow \arg\max_j M_{i^* j}$ ▷ Eqs. (7)–(8)
13: $z \leftarrow z\, a_{i^*} y_{j^*}$ ▷ empty future for an ended action
14: **end while**
15: **return** generated response

Line 8 builds each column by Eq. (16); lines 11–12 read and select without rescoring. The following result accounts for dependence between shared continuations.

Theorem 5.1 (Consistent future comparison)**.** *Fix a finite candidate pool, weights b, and finite horizon H. Let $W = (b_1 B_1, \dots, b_m B_m)$ be the target row weights. Under exact probabilities, full proposal support, and fresh randomized stratification, as the number of shared futures $n \to \infty$,*

$$\mathbb{E}\left[\|\widehat{W} - W\|_2^2\right] \to 0, \qquad \mathbb{P}\left(\arg\max_i \widehat{W}_i = \arg\max_i W_i\right) \to 1,$$

where the second limit assumes that W has a unique largest entry.

Appendix G proves consistency and extends the result to finite-cap output convergence under unique row and continuation choices. The limiting object is this decoder itself, rather than a correctness guarantee, an RL policy, or a global power law.

6 Experiments

Models, tasks, and baselines. We evaluate three models: the math-specialized Qwen2.5-Math-7B and Qwen2.5-Math-1.5B (Yang et al., 2024), and the general-purpose Llama-3.2-3B-Instruct (Meta, 2024). Our benchmarks include MATH500 (Lightman et al., 2023; HuggingFaceH4, 2024) and AI-MO AMC (AI-MO, 2024), both consisting of competition-mathematics problems; GPQA Diamond, a graduate-level science benchmark (Rein et al., 2023); and HumanEval, a Python programming benchmark (Chen et al., 2021). We score one final response per problem using accuracy for mathematics and science and pass@1 for code. We compare against Base sampling, low-temperature decoding, MCMC Power (Karan & Du, 2025), and publicly released GRPO checkpoints post-trained on mathematical data (Shao et al., 2024). Checkpoints and settings are in Appendix H.3.

Table 1: **Reasoning results (%).** Bold marks the best score in each column.

	Qwen2.5-Math-7B				Qwen2.5-Math-1.5B			Llama-3.2-3B-Instruct		
Method	MATH500	AI-MO	GPQA	HumanEval	MATH500	AI-MO	HumanEval	MATH500	AI-MO	HumanEval
Base	51.80	28.92	29.29	28.66	47.60	21.69	20.73	41.40	19.28	32.93
Low Temp.	66.80	44.58	32.32	53.66	61.20	34.94	36.59	44.40	15.66	43.29
MCMC Power	71.60	45.78	35.35	55.49	63.00	37.35	38.41	49.00	25.30	49.39
SFPS	**77.60**	53.01	**42.42**	**60.98**	**70.40**	**44.58**	**43.90**	51.60	**27.71**	**50.61**
GRPO	77.20	**57.83**	39.39	56.10	69.20	43.37	40.85	**52.00**	26.51	48.17

Reasoning performance and cost. Table 1 shows that SFPS outperforms MCMC Power on all ten evaluated model–task pairs and exceeds GRPO on eight. For example, on 7B MATH500, it reaches

Table 2: **Inference cost per problem on Qwen2.5-Math-7B.** T is the mean Base token count.

Method	Cost	MATH500	AI-MO	GPQA	HumanEval
Base	Tokens	672.6 (1.00T)	767.0 (1.00T)	776.2 (1.00T)	453.3 (1.00T)
	Time (s)	8.25 (1.00×)	9.40 (1.00×)	9.68 (1.00×)	5.61 (1.00×)
Low Temp.	Tokens	635.3 (0.94T)	802.6 (1.05T)	873.2 (1.12T)	519.0 (1.14T)
	Time (s)	7.73 (0.94×)	9.77 (1.04×)	10.88 (1.12×)	6.54 (1.17×)
MCMC Power	Tokens	14,544.6 (21.63T)	21,037.5 (27.43T)	22,715.8 (29.27T)	13,445.8 (29.66T)
	Time (s)	179.45 (21.75×)	259.79 (27.63×)	285.99 (29.56×)	167.75 (29.93×)
SFPS	Tokens	3,381.1 (5.03T)	4,715.9 (6.15T)	3,793.4 (4.89T)	2,442.8 (5.39T)
	Time (s)	16.55 (2.01×)	25.19 (2.68×)	29.40 (3.04×)	13.49 (2.41×)

77.60% accuracy, compared with 71.60% for MCMC Power and 77.20% for GRPO. These gains are obtained with substantially lower inference cost: across the four 7B tasks, SFPS uses only **20.13%** of MCMC Power's generated tokens and **9.31%** of its measured inference time on average (Table 2).

Future evidence changes the preferred candidate. Figure 4(a) plots each selected candidate's initial support b_i (horizontal) against its final support $\widehat{\pi}_i$ (vertical). Points above the diagonal indicate candidates that gain support. In 418 of 1,880 blocks (**22.23%**), a candidate initially ranked below first is promoted to first place. This demonstrates that future evidence can overcome an initial preference deficit and induce a position shift, as characterized by Proposition 2.1.

Polar sampling improves the local estimate. Figure 4(b) compares token proposals $g_{\mathrm{Polar}}(x \mid h) \propto \|f(x)\|_b$, $g_{\mathrm{Mass}}(x \mid h) \propto \sum_i f_i(x)$, and $g_{\mathrm{Euclidean}}(x \mid h) \propto \|f(x)\|_2$ at fixed history h. Each uses one-token estimate $f(x)/g(x \mid h)$ of the full contribution $F_0 = \sum_u f(u)$. We measure their relative one-step MSE as defined in Appendix Eq. (I.1). Across 4,266 probes, averaging first within questions and then across questions gives 0.195 for Polar, 0.909 for Mass, and 0.903 for Euclidean. Polar reduces error by **78.6%** relative to Mass, consistent with its local estimation advantage in Proposition 4.3.

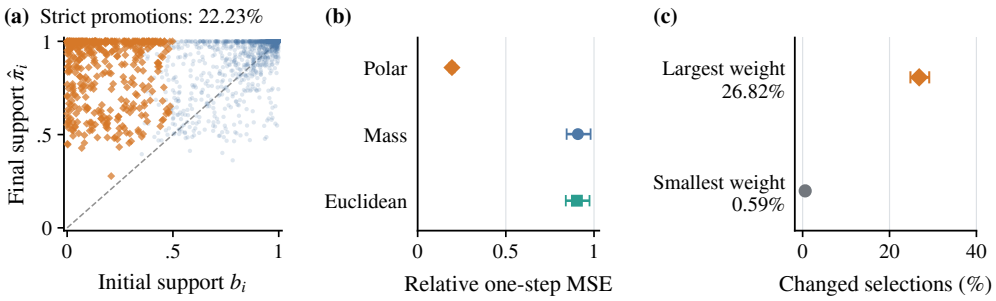


Figure 4: **Mechanism analysis.** (a) Selected-candidate support: orange diamonds indicate strict promotions, where a lower-ranked candidate becomes the first-ranked candidate; blue circles denote other competitive blocks. (b) Relative one-step MSE of Polar (orange), Mass (blue), and Euclidean (green). (c) Fraction of selections changed after deleting the largest- (orange) or smallest-weight (gray) column from 1,864 matrices. Bars show 95% confidence intervals from resampling questions.

The matrix readout requires both aggregate evidence and column weights. See Appendix I.2 for details of the Initial-b and Equal-column ablations. The Initial-b ablation ignores aggregate future evidence, whereas Equal-column removes the contribution of column magnitude. Their accuracies are 3.6 and 3.4 points below SFPS, respectively (Table 3). Figure 4(c) provides a complementary sensitivity analysis: deleting the largest-weight column changes 26.82% of selections, compared with only 0.59% when deleting the smallest-weight column. Both aggregate evidence and column weights contribute meaningfully to the matrix in Eq. (5).

Table 3: **Readout ablations.** MATH500 accuracy and SFPS gains with 95% paired bootstrap confidence intervals.

Decoder	Acc. (%)	Difference [95% CI]
SFPS	**77.6**	—
Initial-b	74.0	+3.6 [1.2, 6.0]
Equal-column	74.2	+3.4 [1.2, 5.6]

AI USE STATEMENT

Generative AI tools were used for language editing, including grammar correction, wording refinement, and improvements to readability.

REFERENCES

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The unreasonable effectiveness of entropy minimization in LLM reasoning. *arXiv preprint arXiv:2505.15134*, 2025. URL https://arxiv.org/abs/2505.15134.

AI-MO. AIMO validation AMC dataset. Dataset card, 2024. URL https://huggingface.co/datasets/AI-MO/aimo-validation-amc.

John Aitchison. The statistical analysis of compositional data. *Journal of the Royal Statistical Society: Series B*, 44(2):139–160, 1982. doi: 10.1111/j.2517-6161.1982.tb01195.x.

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998. doi: 10.1162/089976698300017746.

Seyedarmin Azizi, Erfan Baghaei Potraghloo, Minoo Ahmadi, Souvik Kundu, and Massoud Pedram. Power-SMC: Low-latency sequence-level power sampling for training-free LLM reasoning. *arXiv preprint arXiv:2602.10273*, 2026. URL https://arxiv.org/abs/2602.10273.

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

Nicolas Chopin. Central limit theorem for sequential monte carlo methods and its application to bayesian inference. *The Annals of Statistics*, 32(6):2385–2411, 2004. doi: 10.1214/009053604000000698.

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

Imre Csiszár. Information-type measures of difference of probability distributions and indirect observations. *Studia Scientiarum Mathematicarum Hungarica*, 2:299–318, 1967.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. URL https://arxiv.org/abs/2501.12948.

Randal Douc, Olivier Cappé, and Eric Moulines. Comparison of resampling schemes for particle filtering. In *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*, pp. 64–69, 2005. doi: 10.1109/ISPA.2005.195385. URL https://arxiv.org/abs/cs/0507025.

Gonçalo R. A. Faria, Sweta Agrawal, António Farinhas, Ricardo Rei, José G. C. de Souza, and André F. T. Martins. QUEST: Quality-aware metropolis–hastings sampling for machine translation. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.00049.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://arxiv.org/abs/2305.14992.

W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. doi: 10.1093/biomet/57.1.97.

Tim C. Hesterberg. Weighted average importance sampling and defensive mixture distributions. *Technometrics*, 37(2):185–194, 1995. doi: 10.1080/00401706.1995.10484303.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

HuggingFaceH4. MATH-500 dataset. Dataset card, 2024. URL https://huggingface.co/datasets/HuggingFaceH4/MATH-500.

Xiaotong Ji, Rasul Tutunov, Matthieu Zimmer, and Haitham Bou Ammar. Scalable power sampling: Unlocking efficient, training-free reasoning for LLMs via distribution sharpening. *arXiv preprint arXiv:2601.21590*, 2026. URL https://arxiv.org/abs/2601.21590.

Aayush Karan and Yilun Du. Reasoning with sampling: Your base model is smarter than you think. *arXiv preprint arXiv:2510.14901*, 2025. URL https://arxiv.org/abs/2510.14901.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22199–22213, 2022. URL https://arxiv.org/abs/2205.11916.

kong3125. Qwen2.5-MATH-1.5B-BASE-grpo model card. Hugging Face model card, 2025. URL https://huggingface.co/kong3125/Qwen2.5-MATH-1.5B-BASE-grpo.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M. Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024. URL https://arxiv.org/abs/2409.12917.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018. URL https://arxiv.org/abs/1805.00909.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2023. URL https://arxiv.org/abs/2210.15097.

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

Zicheng Lin, Tian Liang, Jiahao Xu, Qiuzhi Lin, Xing Wang, Ruilin Luo, Chufan Shi, Siheng Li, Yujiu Yang, and Zhaopeng Tu. Critical tokens matter: Token-level contrastive estimation enhances LLM's reasoning capability. *arXiv preprint arXiv:2411.19943*, 2024. URL https://arxiv.org/abs/2411.19943.

Fernando Llorente and Luca Martino. Optimality in importance sampling: A gentle survey. *arXiv preprint arXiv:2502.07396*, 2025. URL https://arxiv.org/abs/2502.07396.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2303.17651.

Meta. Llama-3.2-3B-Instruct model card. Hugging Face model card, 2024. URL https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. doi: 10.1063/1.1699114.

Tu Nguyen, Rasul Tutunov, Xiaotong Ji, and Matthieu Zimmer. The model knows, the decoder finds: Future value guided particle power sampling. *arXiv preprint arXiv:2605.02427*, 2026. URL https://arxiv.org/abs/2605.02427v1.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021. URL https://arxiv.org/abs/2112.00114.

Sean O'Brien and Mike Lewis. Contrastive decoding improves reasoning in large language models. *arXiv preprint arXiv:2309.09117*, 2023. URL https://arxiv.org/abs/2309.09117.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022. URL https://arxiv.org/abs/2203.02155.

Art B. Owen. *Monte Carlo Theory, Methods and Examples*. Online manuscript, 2013. URL https://artowen.su.domains/mc/.

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

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.18290.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. *arXiv preprint arXiv:2311.12022*, 2023. URL https://arxiv.org/abs/2311.12022.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Rulin Shao, Shuyue Stella Li, Rui Xin, Scott Geng, Yiping Wang, Sewoong Oh, Simon Shaolei Du, Nathan Lambert, Sewon Min, Ranjay Krishna, Yulia Tsvetkov, Hannaneh Hajishirzi, Pang Wei Koh, and Luke Zettlemoyer. Spurious rewards: Rethinking training signals in RLVR. *arXiv preprint arXiv:2506.10947*, 2025. URL https://arxiv.org/abs/2506.10947.

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

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. URL https://arxiv.org/abs/2408.03314.

stellalisy. Qwen2.5-Math-7B ground-truth GRPO checkpoint. Hugging Face model release, 2025. URL https://huggingface.co/stellalisy/rethink_rlvr_reproduce-ground_truth-qwen2.5_math_7b-lr5e-7-kl0.00-step150.

TMLR-Group-HF. GT-Llama-3.2-3B-Instruct-MATH model card. Hugging Face model card, 2025. URL https://huggingface.co/TMLR-Group-HF/GT-Llama-3.2-3B-Instruct-MATH.

Tim van Erven and Peter Harremoës. Rényi divergence and Kullback–Leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014. doi: 10.1109/TIT.2014.2320500. URL https://arxiv.org/abs/1206.2459.

Milan Vojnovic and Se-Young Yun. What is the alignment objective of GRPO? *arXiv preprint arXiv:2502.18548*, 2025. URL https://arxiv.org/abs/2502.18548.

Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2402.10200.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. URL https://arxiv.org/abs/2201.11903.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. doi: 10.1007/BF00992696.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv preprint arXiv:2408.00724*, 2024. URL https://arxiv.org/abs/2408.00724.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-Math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024. URL https://arxiv.org/abs/2409.12122.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023a. URL https://arxiv.org/abs/2305.10601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023b. URL https://arxiv.org/abs/2210.03629.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2504.13837.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. SimpleRL-Zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv preprint arXiv:2503.18892*, 2025. URL https://arxiv.org/abs/2503.18892.

Stephen Zhao, Rob Brekelmans, Alireza Makhzani, and Roger Grosse. Probabilistic inference in language models via twisted sequential monte carlo. *arXiv preprint arXiv:2404.17546*, 2024. URL https://arxiv.org/abs/2404.17546.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning, acting, and planning in language models. In *Proceedings of the International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2310.04406.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2205.10625.

Felix Zhou, Anay Mehrotra, and Quanquan C. Liu. Reasoning with sampling: Cutting at decision points. *arXiv preprint arXiv:2605.30327*, 2026. URL https://arxiv.org/abs/2605.30327.

A Related work

Reinforcement learning for reasoning. Policy-gradient methods learn from sampled trajectories and their returns (Williams, 1992; Schulman et al., 2017). RLHF uses preference feedback, while GRPO, DeepSeek-R1, and SimpleRL-Zoo study reasoning-oriented post-training with verifiable rewards (Ouyang et al., 2022; Shao et al., 2024; DeepSeek-AI, 2025; Zeng et al., 2025). GRPO compares rewards within rollout groups to construct relative advantages. Comparisons with base models investigate how RL changes access to existing reasoning behavior (Yue et al., 2025). We borrow the idea of comparing rollouts, but use base-model likelihoods rather than rewards, without training.

Multi-step reasoning, search, and agents. Multi-step inference is not restricted to models without RL training. DeepSeek-R1-Zero develops long reasoning chains, reflection, and self-verification through RL (DeepSeek-AI, 2025); SCoRe explicitly uses multi-turn RL to train self-correction (Kumar et al., 2024). They connect the two routes: training can teach a model how to use additional reasoning steps. Scratchpads, CoT, and least-to-most prompting also organize intermediate computation (Nye et al., 2021; Wei et al., 2022; Kojima et al., 2022; Zhou et al., 2023). Tree of Thoughts and reasoning-as-planning compare paths, while ReAct, Reflexion, and language-agent tree search combine reasoning with actions or feedback (Yao et al., 2023a; Hao et al., 2023; Yao et al., 2023b; Shinn et al., 2023; Zhou et al., 2024). Verifiers, process supervision, and critical-token analyses emphasize consequential decisions (Cobbe et al., 2021; Lightman et al., 2023; Lin et al., 2024). Our use of multi-step inference is blockwise construction of one response: each block compares a small shared set of continuations without training.

Distribution sharpening and training-free decoding. Power sampling targets a sharpened sequence distribution and reports reasoning gains without training (Karan & Du, 2025); Scalable Power Sampling and Power-SMC develop more efficient power-based samplers (Ji et al., 2026; Azizi et al., 2026). Entropy minimization and confidence-based objectives offer related concentration-based approaches (Agarwal et al., 2025; Prabhudesai et al., 2025). CoT-decoding, contrastive decoding, and Self-Refine provide other ways to change outputs without updating the base model (Wang & Zhou, 2024; Li et al., 2023; O'Brien & Lewis, 2023; Madaan et al., 2023). Sharpening is a distributional design principle; position shift identifies the change needed to replace a leading choice. Sequence-level sharpening can also change local rankings. We make the candidate comparison explicit and construct a shared matrix and polar rule to carry it out.

B Position shift and the information in rollouts

B.1 Setting and notation

A prefix z contains the question and the text already committed. An action a_i is a candidate token segment, and y is a future after it. All probability sums use finite token trees: a path ends at its first EOS token or the specified horizon. Its probability includes EOS if EOS was generated. Action leaves are disjoint events; future leaves form a normalized conditional distribution. Horizon leaves need not contain answers.

Unless stated otherwise, expectations condition on the candidate pool and its weights. Assume exact arithmetic and supported proposals; remove zero-mass rows. Geometric results use positive weights, and unique-winner statements exclude ties. Table 4 collects the main symbols.

Table 4: Notation. Lower-case g is a token proposal; G is its path probability.

Symbol	Meaning
z, a_i, y_j	Written prefix, candidate opening, shared future
$p_i(y)$, $p_i(x \mid h)$	Base-model probabilities after a_i; h is a future prefix
b_i	Initial proposal-corrected candidate weight, normalized over the pool
B_i, W_i	Future sum $\sum_y p_i(y)^\alpha$ and supported row weight $b_i B_i$
$\mathbf{M}$, $\widehat{W}$	Candidate–future matrix and its mean row support
$\mathbf{F}(h)$, $f(x)$	Candidate–token contribution matrix and its token column
w_i	Running importance weight within one future
μ, η, r, θ	Contribution mass, normalized allocation, radius, and angle from b
v, ρ, u	Centered log-score, its Euclidean radius, and unit direction
$\|e\|_b^2$	Weighted square $\sum_i e_i^2/b_i$

B.2 Distributional change and a shift of the leading candidate

Position shift throughout the paper refers to the leading candidate of an output distribution, not its mean or a token's location. Fix a prefix and a finite set of positive-probability alternatives. For the original distribution p and an updated distribution π, define $\Delta_i = \log \pi_i - \log p_i$. The exact identity

$$\pi_i = p_i e^{\Delta_i}, \qquad \log \frac{\pi_j}{\pi_k} = \log \frac{p_j}{p_k} + \Delta_j - \Delta_k \tag{B.1}$$

holds whether π comes from learning or decoding. Normalizing both distributions on a common candidate subset adds only a constant to the Δ_i, leaving every pairwise difference unchanged. These quantities are descriptive: an arbitrary change in log-likelihood does not by itself identify a reward, a useful future, or a causal learning mechanism.

Proposition B.1 (Position shift is a relative-gain condition)**.** *Let k be the unique original leader. Candidate j overtakes it exactly when*

$$\Delta_j - \Delta_k > \log \frac{p_k}{p_j}. \tag{B.2}$$

The original leader ceases to maximize precisely when this holds for at least one challenger. Positive power reweighting of the same local distribution preserves its ordering.

Proof of Proposition B.1. Candidate j outranks k if and only if the final log-ratio in Eq. (B.1) is positive, which gives Eq. (B.2). The original leader is no longer a maximizer if and only if at least one challenger outranks it. If $\pi_i = p_i^c / \sum_\ell p_\ell^c$ for $c > 0$, its pairwise log-ratio is $c \log(p_j/p_k)$, so the ordering cannot reverse. A tie in the updated distribution is treated as a boundary event rather than assigned a unique new leader. □

Proposition B.2 (First divergence of greedy completions)**.** *Consider two autoregressive policies using greedy next-token decisions, with unique maximizers on their visited prefixes and a common EOS rule and finite output cap. If their completed token strings differ, there is a prefix visited by both at which their selected local maximizers differ.*

Proof. Take the first unequal token, counting EOS as a token. The earlier tokens and prompt coincide, so both policies reach the same prefix. Their next selected tokens differ by construction. Uniqueness turns that change into a strict change of preferred token. If no such prefix existed, induction would give identical outputs up to EOS or the common cap. □

This does not apply to random samples or imply that a local reversal changes the final answer. Because SFPS samples its candidate set, the proposition is motivation, not a causal explanation of every changed output.

B.3 How a rollout changes policy probabilities

For a fixed prompt and a differentiable policy π_ϑ, let $J(\vartheta) = \mathbb{E}_{u\sim\pi_\vartheta} R(u)$, where u is a complete rollout and the reward does not depend directly on ϑ. On the finite tree, differentiating the sum gives the score-function identity (Williams, 1992)

$$\nabla_\vartheta J = \mathbb{E}[R(u)\nabla_\vartheta \log \pi_\vartheta(u)] = \mathbb{E}\left[R(u)\sum_t \nabla_\vartheta \log \pi_\vartheta(u_t \mid u_{<t})\right]. \tag{B.3}$$

Thus the later return multiplies gradients of earlier local decisions. A baseline depending on the prefix, but not on the sampled current action, can be subtracted inside the corresponding expectation because the conditional score has mean zero. PPO and GRPO add their own policy ratios, clipping, reward normalization, and regularization (Schulman et al., 2017; Shao et al., 2024); Eq. (B.3) describes the basic information pathway, not their complete objectives.

Shared parameters and interacting gradients mean that a parameter update need not raise every rewarded token's probability or change a leader. We compare policies at the *same prefix*, not likelihoods at different histories.

B.4 Power marginalization and the reward-regularized optimum

For $\pi_\alpha(ay \mid z) \propto p(ay \mid z)^\alpha$, the chain rule gives

$$\pi_\alpha(a_i \mid z) = \frac{p(a_i \mid z)^\alpha B_i}{\sum_k p(a_k \mid z)^\alpha B_k}, \qquad B_i = \sum_y p_i(y)^\alpha. \tag{B.4}$$

The factor B_i is omitted if one powers the local marginal instead. This is the established sequence-versus-token distinction in power sampling (Karan & Du, 2025). For $\alpha > 1$, the definition of Rényi entropy gives $\log B_i = (1-\alpha)H_\alpha(p_i)$ (van Erven & Harremoës, 2014). At $\alpha = 1$, normalization of the future leaf tree gives $B_i = 1$. Refining a fixed finite block into an opening and a future does not create a new target distribution: Eq. (B.4) is exactly its action marginal. Repeatedly truncating and recommitting blocks is a separate approximation.

For the RL comparison, consider an unrestricted policy on complete continuations and the objective

$$J_R(\pi) = \mathbb{E}_{u\sim\pi} R(zu) - \beta\,\mathrm{KL}(\pi\|p(\cdot \mid z)), \qquad \beta > 0. \tag{B.5}$$

Let $Z_R = \sum_u p(u \mid z)e^{R(zu)/\beta}$ and $\pi^*(u) = p(u \mid z)e^{R(zu)/\beta}/Z_R$. Substitution gives the standard variational identity (Levine, 2018; Rafailov et al., 2023)

$$J_R(\pi) = \beta \log Z_R - \beta\,\mathrm{KL}(\pi\|\pi^*). \tag{B.6}$$

Nonnegativity of KL proves the optimum. Summing $u = a_i y$ over future leaves yields

$$\pi^*(a_i \mid z) \propto p(a_i \mid z)\mathbb{E}_{y\sim p_i} e^{R(za_iy)/\beta}. \tag{B.7}$$

This is Eq. (1) with $\lambda = 1$ and the stated V_i. For bounded reward and large β, a cumulant expansion makes its connection to a rollout value explicit:

$$V_i = \frac{\mathbb{E}_{p_i} R}{\beta} + \frac{\mathrm{Var}_{p_i}(R)}{2\beta^2} + O(\beta^{-3}). \tag{B.8}$$

The leading term is expected future reward, but the exact exponentially tilted value also depends on higher moments. It would be incorrect to replace the exact expression with mean reward in the general identity.

Proof of Proposition 2.1. Softmax preserves score order. Subtracting the two scores in Eq. (1) gives $s_j - s_k = V_j - V_k - \lambda \log[p(a_k \mid z)/p(a_j \mid z)]$. Its strict positivity is exactly Eq. (2); equality gives a tie. □

For the fixed candidate pool, the identical argument uses current score $\log b_i$ and future score $\log B_i$. The exact population row reversal is $\log(B_j/B_k) > \log(b_k/b_j)$. It is a conditional comparison within the sampled pool, not the full action marginal before sampling.

B.5 THE CONDITIONAL FUTURE IDENTITY BEHIND LOCAL POLICY CHANGES

Let a finite complete continuation u have base probability $p(u) > 0$, and attach a finite terminal log-weight $U(u)$. The sequence tilt is $\pi_U(u) = p(u)e^{U(u)}/Z_U$, where $Z_U = \mathbb{E}_p e^U$. For a prefix z, define the remaining conditional normalizer $Z_U(z) = \mathbb{E}_p[e^{U(u)} \mid z]$. Summing over descendants of za gives

$$\pi_U(a \mid z) = p(a \mid z)\frac{Z_U(za)}{Z_U(z)}, \qquad Z_U(z) = \sum_a p(a \mid z)Z_U(za). \tag{B.9}$$

The boundary value at a completed sequence is $Z_U(u) = e^{U(u)}$. This is the finite-tree conditional-tilting identity underlying the inference view of reward-regularized control (Levine, 2018). It displays the information that a future-aware local policy must contain: the relative continuation factors after each possible action.

Writing $V(z) = \log Z_U(z)$ turns the ratio into a log-likelihood update,

$$\log \frac{\pi_U(a \mid z)}{p(a \mid z)} = V(za) - V(z). \tag{B.10}$$

The common $V(z)$ cancels between alternatives; only the difference of their reachable future values can change their comparison. Along a complete path, these increments telescope to $U(u) - \log Z_U(z_0)$. Conversely, every positive policy on the same finite tree admits such a representation by choosing $U(u) = \log[\pi(u)/p(u)]$. That converse is an algebraic representation, not evidence that the actual training process used this terminal score or that the score is recoverable from the base model alone.

For reward tilting, take $U(u) = R(u)/\beta$. For sequence-level power, take $U(u) = (\alpha - 1)\log p(u)$. In the latter case the conditional factor separates into the powered probability of the prefix and the remaining power sum, recovering Eq. (B.4). Thus the familiar power target already carries future-dependent local changes; SFPS changes how the finite-block comparison is estimated, not this underlying identity.

B.6 HOW INCREASING A FUTURE TILT CHANGES A RELATIVE PREFERENCE

Fix the terminal score U and consider $\pi_t(u) \propto p(u)e^{tU(u)}$ for $t \geq 0$. The parameter t is an analysis variable, not a training iteration or a decoder hyperparameter. For candidate a_i, let $\pi_t(\cdot \mid a_i)$ be its conditional future distribution under the tilted measure.

Proposition B.3 (Integrated future advantage)**.** *On the finite supported tree,*

$$\frac{d}{dt}\log\frac{\pi_t(a_j)}{\pi_t(a_k)} = \mathbb{E}_{\pi_t(\cdot|a_j)}U - \mathbb{E}_{\pi_t(\cdot|a_k)}U, \tag{B.11}$$

$$\log\frac{\pi_t(a_j)}{\pi_t(a_k)} = \log\frac{p(a_j)}{p(a_k)} + \int_0^t \left(\mathbb{E}_{\pi_s(\cdot|a_j)}U - \mathbb{E}_{\pi_s(\cdot|a_k)}U\right) ds. \tag{B.12}$$

Consequently, a challenger overtakes the original leader precisely when this integrated conditional advantage exceeds its original log-likelihood deficit.

Proof. Let $C_i(t) = \sum_y p(y \mid a_i)e^{tU(a_i y)}$. The local log-ratio is $\log[p(a_j)/p(a_k)] + \log C_j(t) - \log C_k(t)$. Finite summation permits differentiation, and $C_i'(t)/C_i(t)$ is exactly the conditional tilted mean of U. Integrate from zero, using $C_i(0) = 1$. □

A constant positive difference of conditional means would move the log-odds monotonically, but the means themselves change with t. The formula therefore does not assert that stronger reward tilting or a larger power exponent monotonically improves reasoning accuracy. For each candidate, $d^2 \log C_i(t)/dt^2 = \mathrm{Var}_{\pi_t(\cdot|a_i)} U \geq 0$; the difference of two such convex log-normalizers need not be monotone.

Specializing to power, use $U(a_i y) = (\alpha - 1)\log p(a_i y)$ and $\lambda(t) = 1 + t(\alpha - 1)$. Equation (B.11) becomes

$$(\alpha - 1)\left[\log\frac{p(a_j)}{p(a_k)} + \mathbb{E}_{p_j^{\lambda(t)}/B_j(t)}\log p_j(y) - \mathbb{E}_{p_k^{\lambda(t)}/B_k(t)}\log p_k(y)\right], \tag{B.13}$$

where $B_i(t) = \sum_y p_i(y)^{\lambda(t)}$. The current disadvantage and the difference between future conditional log-likelihoods act together. This is more informative than treating an increased exponent as local sharpening alone.

B.7 Expected rollout reward and a local policy improvement step

A second useful connection fixes the continuation policy while improving only the current action. Let $Q_i = \mathbb{E}[R \mid a_i]$ under that fixed rollout policy, and let $\eta > 0$. The KL-regularized local improvement problem is

$$\max_{\pi} \left\{ \sum_i \pi_i Q_i - \eta^{-1} \operatorname{KL}(\pi \| p) \right\}, \qquad \pi_i^+ = \frac{p_i e^{\eta Q_i}}{\sum_k p_k e^{\eta Q_k}}. \tag{B.14}$$

The proof is the variational identity in Eq. (B.6), applied to a finite action distribution. In this ideal local step, a_j overtakes a_k exactly when $\eta(Q_j - Q_k) > \log(p_k/p_j)$. Rollouts estimate the required relative future value before the policy stores it.

Equation (B.14) and Eq. (B.7) must not be conflated. The first exponentiates an expected return while holding the continuation policy fixed. The second marginalizes a sequence-wide optimum and therefore takes the expectation of an exponential return. Neither is an exact expression for arbitrary finite neural-network updates, and GRPO's group normalization can alter its effective objective (Vojnovic & Yun, 2025). These distinctions are why we use RL as motivation and a performance reference, not as an asserted policy identity for SFPS.

B.8 What rollout reuse can replace

The power and reward targets share a form, not supervision. Hold p fixed and choose two rewards favoring different continuations. Their optimal policies differ; a decoder given only p has the same inputs in both cases. Universal equivalence to RL therefore requires additional reward assumptions. Training also amortizes information across questions, whereas SFPS recomputes it per question without changing weights. Similar task accuracies do not establish policy equivalence.

B.9 From local decisions to final outcomes

For analysis only, let $\psi(u) = c$ group complete continuations by final outcome. For any terminal log-weight $U(u)$, define

$$A_c = \sum_{u:\psi(u)=c} p(u \mid z) e^{U(u)}, \qquad P_c = \sum_{u:\psi(u)=c} p(u \mid z). \tag{B.15}$$

If $P_c > 0$, then $A_c = P_c \mathbb{E}[e^U \mid \psi(u) = c]$. The outcome log-ratio therefore separates into prior mass and relative future support, just like the action comparison. At a fixed action the joint support further factors as

$$A(a, c) = p(a \mid z)\, P(c \mid za)\, \mathbb{E}[e^{U(ay)} \mid a, \psi(ay) = c]. \tag{B.16}$$

The factors distinguish the opening, reachable-outcome probability, and within-outcome weighting. Different openings may reach the same answer. SFPS compares candidate strings, not outcome clusters; no answer parser is used for selection.

B.10 Multi-step inference and a fixed comparison

RL changes a policy's parameters; ordinary CoT changes the context on which later tokens are conditioned. They therefore cannot both be described as a parameter update at the same prefix. Nor can a change from the next token at one position to the next token at another be treated automatically as a position shift: the choices being made differ. The comparison must retain the same alternatives.

For an exact description along a generated trace, fix a finite joint law for a complete continuation Y and its final-answer label $A = \psi(Y)$. Let H_t be the observed prefix and let $P_t(c) = \mathbb{P}(A = c \mid H_t)$. After observing the next token x, Bayes' rule gives, on positive support,

$$\log \frac{P_{t+1}(c)}{P_{t+1}(k)} = \log \frac{P_t(c)}{P_t(k)} + \log \frac{p(x \mid H_t, A = c)}{p(x \mid H_t, A = k)}. \tag{B.17}$$

The alternatives c and k remain fixed while the reasoning context changes. If k currently leads, c overtakes it exactly when the last log-ratio exceeds $\log[P_t(k)/P_t(c)]$. The centered log-vector of P_t also admits the radial–residual decomposition in Appendix C. Thus reinforcement and redirection are well-defined for changes in answer support, not for a raw comparison of different token decisions. For agent observations the same conditional identity applies only after specifying a joint law that includes the environment.

Equation (B.17) is Bayes conditioning, not a claim that CoT explicitly computes these answer probabilities. Its terms may be intractable, and generated text need not move support toward the correct answer. The position-shift analysis describes a possible decision change; it does not by itself establish the cause of every reported CoT or RL improvement. Our algorithm avoids estimating these answer-conditioned quantities: it fixes short candidate openings at a common prefix and compares their shared future likelihoods. No answer labels or new observation model are needed by SFPS.

C Geometry of reinforcement and position shift

C.1 Centering removes normalization, not relative preference

For $\pi_i = e^{s_i} / \sum_k e^{s_k}$, subtracting the mean log-probability gives $v_i = s_i - \frac{1}{m}\sum_k s_k$. Hence

$$v_i - v_k = \log \frac{\pi_i}{\pi_k} = s_i - s_k, \qquad \arg\max_i v_i = \arg\max_i \pi_i. \tag{C.1}$$

This is the standard centered-log representation (Aitchison, 1982). For $c > 0$, replacing v by cv gives probabilities proportional to π_i^c. Every ordering is preserved, while the log-gaps are scaled by c. Increasing c strengthens the preference pattern; it does not preserve the numerical pairwise probability ratios.

Proposition C.1 (Exact boundary condition)**.** *Let k lead at v. For any update $v' = v + \delta$, candidate j overtakes k precisely when $\delta_j - \delta_k > v_k - v_j$. If $v' = cv + d$, the same condition is $d_j - d_k > c(v_k - v_j)$. In particular, with $c > 0$ and $d = 0$ there is no position shift.*

Proof. Both statements follow by writing $v'_j - v'_k$ in the corresponding decomposition. Positivity of c is required only for the order-preserving interpretation, not for the algebraic identity. □

The Euclidean distance from v to the tie hyperplane $v_j = v_k$ is $(v_k - v_j)/\sqrt{2}$. The nearest tie for the lead is attained by a largest challenger. Averaging its score with the leader's leaves the other scores no larger, so the closest hyperplane point is indeed a leading tie. A strict reversal has greater distance, with infimum

$$\min_{j \neq k} \frac{v_k - v_j}{\sqrt{2}}. \tag{C.2}$$

This is a geometric boundary distance, not a threshold implemented by the decoder.

C.2 Exact residual size in the positive radial regime

Proof of Corollary 4.2. A changed leader means that some challenger j satisfies $d_j - d_k > c(v_k - v_j)$ by Proposition 4.1. Let e_i be a coordinate basis vector. Cauchy–Schwarz gives

$$c \min_{\ell \neq k}(v_k - v_\ell) \leq c(v_k - v_j) < d_j - d_k = \langle e_j - e_k, d \rangle \leq \sqrt{2}\,\|d\|_2. \tag{C.3}$$

Rearranging proves Eq. (12). The $\sqrt{2}$ is the Euclidean norm of a pairwise comparison direction, not a fitted constant. The result is necessary, not sufficient, because a large residual can point toward the wrong comparison. The positive-c condition matters: negative radial scaling can reverse an ordering even when $d = 0$. The sharper result below also enforces $d \perp v$. □

The general Euclidean boundary distance does not enforce the decomposition's constraint $d \perp v$. Enforcing it yields a sharper statement. Fix a nonzero centered vector v, a unique leader k, and a

positive radial coefficient c. For challenger j, write $\gamma_j = v_k - v_j > 0$. Let e_i denote coordinate basis vectors and define

$$n_j = e_j - e_k + \frac{\gamma_j}{\|v\|_2^2} v, \qquad \|n_j\|_2^2 = 2 - \frac{\gamma_j^2}{\|v\|_2^2}. \tag{C.4}$$

This is the projection of the comparison direction onto the centered subspace perpendicular to v.

Proposition C.2 (Smallest orthogonal residual that can overturn the leader)**.** *Among centered residuals $d \perp v$, the infimum norm needed to make j strictly outrank k in $cv + d$ is*

$$\frac{c\gamma_j}{\sqrt{2 - \gamma_j^2/\|v\|_2^2}}, \tag{C.5}$$

with value infinity when the denominator is zero. The infimum needed for any challenger to overtake is the minimum of Eq. (C.5) *over $j \neq k$.*

Proof. For $d \perp v$, its comparison gain satisfies $d_j - d_k = \langle n_j, d \rangle$. Cauchy–Schwarz bounds this by $\|n_j\|_2 \|d\|_2$. The required strict gain is $c\gamma_j$ by Proposition 4.1. If $n_j \neq 0$, the centered vector $d = c\gamma_j n_j / \|n_j\|_2^2$ attains equality at a tie, and an arbitrarily small positive rescaling gives a strict reversal. If $n_j = 0$, all admissible residuals have zero comparison gain. Taking the union over challengers gives the final statement. □

The bound scales linearly with the positive radial coefficient: stronger order-preserving sharpening demands more directional evidence when everything else is held fixed. In the two-candidate centered space every n_j is zero, recovering the fact that an orthogonal residual cannot reverse the one-dimensional direction; a sign change of the radial coefficient is needed instead. This is a geometric statement, not a sampling threshold or an error bound.

Proof of Proposition 4.1. Subtract the k coordinate from the j coordinate in $v' = cv + d$. The difference is $-c(v_k - v_j) + (d_j - d_k)$, and strict positivity is exactly Eq. (11). For $c > 0, d = 0$, the sign of every pairwise difference is unchanged. □

C.3 The finite polar identity and its exceptional cases

For $v \neq 0$, let $\rho = \|v\|_2$, choose $c = \langle v', v \rangle / \|v\|_2^2$, and set $d = v' - cv$. Then $d \perp v$. If $v' \neq 0$, let $\rho' = \|v'\|_2$ and $\varphi = \arccos(\langle v', v \rangle / (\|v'\|_2 \|v\|_2))$. Pythagoras gives

$$\rho'^2 = c^2\rho^2 + \|d\|_2^2, \quad \cos\varphi = \frac{c\rho}{\rho'}, \quad \sin\varphi = \frac{\|d\|_2}{\rho'}. \tag{C.6}$$

For $d \neq 0$, define $u = v/\rho$ and $u_\perp = d/\|d\|_2$. These identities give the complete finite polar representation

$$v' = \rho'\big(\cos\varphi\, u + \sin\varphi\, u_\perp\big), \qquad c\rho = \rho'\cos\varphi, \quad \|d\|_2 = \rho'\sin\varphi. \tag{C.7}$$

The residual direction remains a vector; for multiple candidates, its norm and angle do not say which coordinates it increases. The finite polar representation is equivalent to the full radial–residual decomposition, not a source of extra statistical information.

If $d = 0$ and $c > 0$, the angle is zero; if $d = 0$ and $c < 0$, it is π. The latter can reverse the leader without an orthogonal residual. With two candidates, the centered subspace is one-dimensional, so this sign change is the only way to change direction. A claim that every reversal requires a nonzero orthogonal residual would therefore be false. At $v = 0$, or $v' = 0$, the corresponding direction is undefined; the pairwise score comparison in Proposition C.1 remains valid.

C.4 Evidence accumulation and the first crossing

Let $v(t)$ be differentiable and nonzero, with $v(t) = \rho(t)u(t)$. Differentiating $\rho^2 = \langle v, v \rangle$ and then $u = v/\rho$ gives

$$\dot\rho = \langle u, \dot v \rangle, \qquad \dot u = \frac{\dot v - u\langle u, \dot v \rangle}{\rho}, \qquad \|\dot v\|_2^2 = \dot\rho^2 + \rho^2 \|\dot u\|_2^2. \tag{C.8}$$

The orthogonal component changes direction at a speed that depends on the current radius. Thus parallel and perpendicular motion are locally separable but not independent coordinates for a finite sequence of updates. These derivatives describe a score path, not learned training phases and not a numerical integrator used by SFPS.

A particularly transparent path gradually incorporates a *fixed* future-score vector: $\pi_i(t) \propto b_i B_i^t$, for $t \in [0,1]$. Its centered score is $v(t) = v(0) + t\delta$, with $\delta_i = \log B_i - \frac{1}{m}\sum_k \log B_k$. If k uniquely leads initially, define

$$t_* = \min_{j:\delta_j > \delta_k} \frac{v_k(0) - v_j(0)}{\delta_j - \delta_k}, \tag{C.9}$$

with the minimum of the empty set equal to infinity. Before t_*, k leads every challenger; at t_* at least one ties it. A strict change occurs by $t = 1$ exactly when $t_* < 1$. This follows because every leader–challenger gap is affine in t. The variable t is an analysis interpolation, not an additional algorithm parameter or a schedule that SFPS estimates.

The first crossing can occur after both the radius and direction have changed. Increasing a fixed sharpening coefficient raises the initial gap, but in a true power target the future factor B_i also depends on α. Therefore the fixed-vector crossing formula does not imply a monotone relation between the power exponent and reasoning accuracy.

C.5 Why lower entropy is not a reversal certificate

For fixed positive b, let $\pi_\lambda(i) \propto b_i^\lambda$, $\lambda > 0$. Differentiation gives, for any fixed candidate function c_i,

$$\frac{d}{d\lambda}\mathbb{E}_{\pi_\lambda} c = \mathrm{Cov}_{\pi_\lambda}(c, \log b), \qquad \frac{d}{d\lambda} H(\pi_\lambda) = -\lambda \,\mathrm{Var}_{\pi_\lambda}(\log b) \le 0. \tag{C.10}$$

Indeed $\partial_\lambda \pi_\lambda(i) = \pi_\lambda(i)(\log b_i - \mathbb{E}\log b)$, and $H = \log \sum_i b_i^\lambda - \lambda \mathbb{E}\log b$. If c indicates correctness, accuracy can increase without a mode change whenever the covariance is positive. This is why we target correction of a *leading error*, not all sources of higher sampling accuracy.

As a finite illustration, $(0.7, 0.2, 0.1)$ and $(0.2, 0.7, 0.1)$ have equal entropy and different leaders. A positive power of either distribution preserves its leader while changing entropy.

D The matrix estimates a specific future-supported measure

D.1 Candidate generation and exact duplicate weights

Draw K candidate segments independently with replacement from an action proposal q. For token temperature τ, the recorded proposal probability is

$$q(a \mid z) = \prod_t \frac{p(a_t \mid z a_{<t})^{1/\tau}}{\sum_x p(x \mid z a_{<t})^{1/\tau}}. \tag{D.1}$$

The intermediate normalizers depend on the history. For a multi-token segment this is not generally a normalized power of $p(a \mid z)$.

Merge exactly identical token strings. If c_i draws equal a_i, set

$$\widetilde{b}_i = c_i \frac{p(a_i \mid z)^\alpha}{q(a_i \mid z)}, \qquad A = \sum_i \widetilde{b}_i, \qquad b_i = \widetilde{b}_i / A. \tag{D.2}$$

The implementation sums the recorded per-copy weights, allowing harmless arithmetic differences across copies. No semantic grouping, answer clustering, or task reward is involved. Conditioning on this pool leaves the finite row target $W_i = b_i B_i$. The normalization changes the common scale, not the ranking.

D.2 Conditional and outer importance identities

Proposition D.1 (Conditional matrix identity)**.** *If a shared future has marginal probability $G(y) > 0$ on every nonzero row contribution, then the matrix in Eq.* (5) *satisfies $\mathbb{E}[\widehat{W}_i] = b_i B_i$. Dependence between columns does not affect this identity.*

Proof. For each column,

$$\mathbb{E}M_{ij} = \sum_y G(y)\frac{b_i p_i(y)^\alpha}{G(y)} = b_i B_i. \tag{D.3}$$

Average by linearity. An action already ending in EOS has the known factor $B_i = 1$ and is assigned $M_{ij} = b_i$ directly. Its row is exact rather than estimated. □

This is ordinary importance weighting applied to a shared vector (Owen, 2013). The denominator is the probability under the actual shared proposal, not the probability under an arbitrarily designated owner candidate. Normalizing $\widehat{W}$, taking its logarithm, or selecting its argmax does not preserve unbiasedness; consistency is established separately below.

For any bounded action function ϕ, restoring the common scale before taking expectation over candidate draws gives

$$\mathbb{E}\left[\frac{A}{K}\sum_i \widehat{W}_i \phi(a_i)\right] = \sum_a p(a \mid z)^\alpha B(a)\phi(a). \tag{D.4}$$

To prove it, first replace $\widehat{W}_i$ by $b_i B_i$ conditionally using Eq. (D.3). Expand the grouped draws in Ab_i, then take the expectation of each $p(a)^\alpha B(a)\phi(a)/q(a)$ term under q. The factor $1/K$ is needed for the identity even though it cancels in every row decision.

Correcting temperature-based candidate sampling. Candidates are generated token by token at temperature $\tau > 0$. The proposal $q(a \mid z)$ in Eq. (D.1) is this actual sampling probability, including its per-token normalizers. The factor $p(a \mid z)^\alpha$ is the local likelihood weight used in our comparison. Dividing by $q(a \mid z)$ and summing repeated draws corrects the sampling frequencies before future evidence is incorporated.

Lemma D.2 (Consistent candidate comparison under temperature sampling)**.** *Fix a prefix z, exponent $\alpha > 1$, finite opening and continuation horizons L, H, and a candidate temperature $\tau > 0$. Let $\mathcal{A}$ be the finite set of possible openings with positive base-model probability. Assume $q(a \mid z) > 0$ for every $a \in \mathcal{A}$. For K independent draws with replacement from q, let $c_K(a)$ count the occurrences of a and let $b_K(a)$ be the normalized weight in Eq.* (D.2)*, with zero weight for an unobserved opening. Define*

$$B(a) = \sum_y p(y \mid za)^\alpha, \qquad S(a) = p(a \mid z)^\alpha B(a), \qquad Z_\alpha = \sum_{u\in\mathcal{A}} p(u \mid z)^\alpha, \tag{D.5}$$

where y ranges over continuations at the fixed horizon and $S(a)$ is our local–future comparison score. Then, almost surely as $K \to \infty$,

$$b_K(a) \longrightarrow \frac{p(a \mid z)^\alpha}{Z_\alpha}, \qquad W_K(a) := b_K(a)B(a) \longrightarrow \frac{S(a)}{Z_\alpha} \quad (a \in \mathcal{A}). \tag{D.6}$$

If S has a unique maximizer a^*, then $\arg\max_a W_K(a) = a^*$ for all sufficiently large K, almost surely. For SFPS's polar proposal and randomized stratification, under the assumptions of Theorem 5.1, the estimated row weights $\widehat{W}_{K,n}$ from n shared continuations also satisfy

$$\mathbb{P}\left(\arg\max_{a\in\mathcal{A}} \widehat{W}_{K,n}(a) = a^*\right) \longrightarrow 1 \qquad \text{as } K, n \to \infty, \tag{D.7}$$

where unobserved openings again have zero weight.

Proof. By the strong law, $c_K(a)/K \to q(a \mid z)$ simultaneously for every $a \in \mathcal{A}$. Therefore,

$$\frac{1}{K}\, c_K(a)\frac{p(a \mid z)^\alpha}{q(a \mid z)} = \frac{c_K(a)/K}{q(a \mid z)} p(a \mid z)^\alpha \longrightarrow p(a \mid z)^\alpha. \tag{D.8}$$

Let $A_K = \sum_{u\in\mathcal{A}} c_K(u)p(u \mid z)^\alpha/q(u \mid z)$ be the normalizer. Summing over the finite set gives $A_K/K \to Z_\alpha > 0$. Normalization and multiplication by $B(a)$ prove Eq. (D.6). Since the limit S/Z_α has a positive gap between its largest and second-largest entries, the exact row maximizers eventually agree.

For the estimated rows, Proposition D.1 gives $\mathbb{E}[\widehat{W}_{K,n}(a) \mid a^{(1)}, \dots, a^{(K)}] = W_K(a)$. Taking $\phi(u) = \mathbf{1}\{u = a\}$ in Eq. (D.4) gives $\mathbb{E}[(A_K/K)\widehat{W}_{K,n}(a)] = S(a)$. The factor A_K/K is common to all rows and leaves their argmax unchanged.

Equation (G.5) and Theorem G.2, with at most $|\mathcal{A}|$ distinct rows, give

$$\mathbb{E}\left[\max_{a\in\mathcal{A}} |\widehat{W}_{K,n}(a) - W_K(a)|^2 \,\middle|\, a^{(1)}, \dots, a^{(K)}\right] \le \frac{(H+1)|\mathcal{A}|^H}{n}. \tag{D.9}$$

Here the maximum squared error is at most the squared b_K-weighted norm on the observed rows. This bound is uniform in K. Together with Eq. (D.6), it gives uniform convergence in probability to S/Z_α as $K, n \to \infty$. An error smaller than half its leading gap preserves the maximizer, proving Eq. (D.7). □

Why the multiplicities matter. For two fixed candidates, once both have been observed,

$$\frac{b_i}{b_k} = \frac{c_i/(Kq(a_i \mid z))}{c_k/(Kq(a_k \mid z))} \left(\frac{p(a_i \mid z)}{p(a_k \mid z)}\right)^\alpha \longrightarrow \left(\frac{p(a_i \mid z)}{p(a_k \mid z)}\right)^\alpha. \tag{D.10}$$

The counts and proposal denominators work together: c_i/K records how often temperature sampling produces a candidate, and $1/q(a_i \mid z)$ corrects that frequency. The separate $1/G(y)$ factor estimates future evidence from shared continuations. Thus K controls the candidate-weight approximation and n controls the future-evidence approximation.

D.3 What a shared column says and how it should be aggregated

For positive entries, Eq. (6) follows by cancellation of the common denominator. Aggregated odds are not an unweighted average of these ratios:

$$\frac{\widehat{W}_i}{\widehat{W}_k} = \sum_j \frac{M_{kj}}{\sum_\ell M_{k\ell}} \frac{M_{ij}}{M_{kj}}. \tag{D.11}$$

Thus a challenger must have enough weighted relative evidence, not merely one favorable column. With zero entries the equivalent sum inequality remains valid without division. In particular, if every column has $M_{ij} \le M_{kj}$, their aggregate cannot reverse that comparison.

Column masses also matter. The matrices $\left(\begin{smallmatrix} .9 & .2 \\ .1 & .8 \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} .9 & 20 \\ .1 & 80 \end{smallmatrix}\right)$ have the same column-normalized allocations but opposite row-sum winners. Likewise, $\left(\begin{smallmatrix} 6 & .1 \\ 4 & 4 \end{smallmatrix}\right)$ has its largest cell in the first row and its largest row sum in the second. These finite examples explain the two choices in Eq. (7); they do not add a decision rule to the algorithm.

D.4 The two matrices use the same pairwise evidence

Within one future, the update $w_i' = w_i p_i(x \mid h)^\alpha / g(x \mid h)$ gives

$$\log \frac{w_i'}{w_k'} - \log \frac{w_i}{w_k} = \alpha \log \frac{p_i(x \mid h)}{p_k(x \mid h)}. \tag{D.12}$$

If $v_i = \log w_i - \frac{1}{m}\sum_k \log w_k$, then $v_i' - v_i$ is the centered vector of $\alpha \log p_i(x \mid h)$. This is the exact bridge from the per-token matrix to the log-score geometry: the proposal affects which token is observed but not its within-column comparative likelihood ratio. At the future leaf, summing these increments recovers the column ratio in Eq. (6). Across columns one must still sum the original contributions before taking logarithms; an average of centered log-positions would describe a different objective.

E Derivation of the fixed-reference polar rule

E.1 Why use inverse-reference weights?

Consider a continuous positive diagonal quadratic measure $L_b(e) = \sum_i h(b_i)e_i^2$, with the same function h in every dimension. Splitting a candidate of mass b into identical copies of masses

$b_1 + b_2 = b$ gives proportional contributions and errors, $e_\ell = b_\ell e/b$. Invariance requires

$$h(b)e^2 = h(b_1)(b_1 e/b)^2 + h(b_2)(b_2 e/b)^2. \tag{E.1}$$

With $F(b) = b^2 h(b)$, this is $F(b_1 + b_2) = F(b_1) + F(b_2)$. Continuity implies $F(b) = cb$, first on rational subdivisions and then on all positive $b \le 1$. Hence $h(b) = c/b$; the converse follows by substitution. This identifies the metric only within the stated class, not among all possible sampling objectives.

The proposal itself respects the same split, since proportional copies obey $\sum_\ell f_\ell(x)^2/b_\ell = f(x)^2/b$. By induction all subsequent normalized proposals remain unchanged. Summing the copies' final weights recovers the original row. This explains why duplicate grouping preserves the construction, while selecting unmerged individual copies by argmax would not.

For a zero-sum probability perturbation e with $b + e > 0$, Taylor expansion gives

$$\mathrm{KL}(b + e\|b) = \tfrac{1}{2}\|e\|_b^2 + O(\|e\|^3). \tag{E.2}$$

The weighted metric is the local quadratic geometry of relative entropy (Amari, 1998). Grouping coordinates contracts it by weighted Cauchy–Schwarz: $\sum_C(\sum_{i\in C} e_i)^2/(\sum_{i\in C} b_i) \le \sum_i e_i^2/b_i$, with equality within a group only for proportional relative errors. This is consistent with divergence contraction (Csiszár, 1967); grouping distinct candidates can erase comparisons, unlike merging identical copies.

E.2 Magnitude, allocation, and angle

Write $f = \mu b + e$, where $\mu = \sum_i f_i$ and $\sum_i e_i = 0$. In the inner product $\langle x, y\rangle_b = \sum_i x_i y_i/b_i$, we have $\|b\|_b = 1$ and $\langle b, e\rangle_b = 0$. Therefore $r^2 = \|f\|_b^2 = \mu^2 + \|e\|_b^2$. For nonzero nonnegative f, its angle from b satisfies

$$\cos\theta = \frac{\mu}{r}, \qquad \tan^2\theta = \left\|\frac{f}{\mu} - b\right\|_b^2 = \chi^2(f/\mu\|b). \tag{E.3}$$

This proves Eq. (13). Equivalently, it is ordinary Euclidean geometry after dividing coordinate i by $\sqrt{b_i}$. The reference vector becomes $\sqrt{b}$, a unit vector. No inverse trigonometric calculation is needed in the implementation.

The contribution radius and the centered-log radius answer different questions. The first preserves absolute importance mass across columns; the second describes confidence in a normalized candidate distribution. A common positive scale changes contribution size without changing the allocation. A common positive multiplier of centered log-scores changes concentration. Their connection is the explicit sequence $f \to M_{:j} \to \widehat{W} \to \widehat{\pi}$, not an identification of these radii.

E.3 Norm stabilization and one-step optimality

Proof of Proposition 4.3. Set $S = \sum_x r(x)$. For every nonzero token column, $g(x) = r(x)/S$. Then $\|f(x)/g(x)\|_b = S$ exactly. Also $\sum_x g(x)f(x)/g(x) = \sum_x f(x)$, proving both identities. □

Let $F = \sum_x f(x)$. The expected squared deviation under any supported proposal is

$$\mathcal{E}(g) = \mathbb{E}_{x\sim g}\|f(x)/g(x) - F\|_b^2 = \sum_x \frac{r(x)^2}{g(x)} - \|F\|_b^2. \tag{E.4}$$

Cauchy–Schwarz gives $(\sum_x r(x))^2 \le (\sum_x r(x)^2/g(x))(\sum_x g(x))$, with equality exactly when $g(x) \propto r(x)$ on the nonzero support. This is the standard vector importance-sampling optimum (Owen, 2013; Llorente & Martino, 2025), instantiated for the candidate matrix and fixed reference.

More precisely, for $g^*(x) = r(x)/S$,

$$\mathcal{E}(g) - \mathcal{E}(g^*) = S^2\chi^2(g^*\|g). \tag{E.5}$$

For the scalar-mass proposal $g_{\mathrm{mass}}(x) = \mu(x)/Z$, $Z = \sum_x \mu(x)$, substitution of $r = \mu\sec\theta$ yields

$$\mathcal{E}(g_{\mathrm{mass}}) - \mathcal{E}(g^*) = Z^2\,\mathrm{Var}_{g_{\mathrm{mass}}}[\sec\theta(x)]. \tag{E.6}$$

There is no improvement if the angular factor is constant. The rule's benefit is not a theorem that larger angles are always desirable; it is an exact comparison of proposals for the same vector estimator.

To separate unavoidable directional variation from proposal choice, define the unit contribution direction $u_x = f(x)/r(x)$. Expanding $\|F\|_b^2$ in Eq. (E.4) gives

$$\mathcal{E}(g) = \frac{1}{2}\sum_{x,t} r(x)r(t)\|u_x - u_t\|_b^2 + S^2\chi^2(g^*\|g). \tag{E.7}$$

The first term is the minimum one-step vector error; the polar rule removes the second term. Norm stabilization therefore does not mean zero vector error: the direction remains random.

E.4 When no token supplies a new row comparison

Proposition E.1 (Rank-one characterization)**.** *For positive w_i and $\alpha > 0$, the token matrix in Eq. (9) has rank one if and only if all its candidate-conditioned next-token distributions are identical. In that case the polar and scalar-mass proposals agree and the row-weight ratios are unchanged.*

Proof. Rank one makes the nonzero rows proportional. Dividing out w_i, taking the positive αth root, and using $\sum_x p_i(x \mid h) = 1$ makes the proportionality constant one. Conversely, for a common distribution p_h, every column is $f(x) = p_h(x)^\alpha w$. Both proposals are proportional to $p_h(x)^\alpha$, and every row is multiplied by $\sum_x p_h(x)^\alpha$ after the update. □

Rank is not used as a gate. Later distributions can differ after a rank-one step, and nonproportional columns can cancel on integration; neither rank condition certifies an answer change.

F Aggregation, decision boundaries, and finite error

F.1 Strong column movements can cancel

Write each nonzero future column as $M_{:j} = \mu_j\eta_j$, where $\mu_j = \sum_i M_{ij}$ and η_j is a probability vector. With $\omega_j = \mu_j/\sum_\ell \mu_\ell$, the normalized row estimate is $\bar\eta = \sum_j \omega_j\eta_j$. The weighted variance identity gives

$$\sum_j \omega_j\|\eta_j - b\|_b^2 = \|\bar\eta - b\|_b^2 + \sum_j \omega_j\|\eta_j - \bar\eta\|_b^2. \tag{F.1}$$

To prove it, expand $\eta_j - b = (\bar\eta - b) + (\eta_j - \bar\eta)$; the weighted cross term vanishes. By Eq. (E.3), the left side is the mass-weighted mean of $\tan^2\theta_j$. The first term is net movement and the second is disagreement. Thus large per-column angles need not produce a large change in the aggregate. For instance, equal-mass allocations $(.9, .1)$ and $(.1, .9)$ average to $(.5, .5)$. No angle-only selection rule follows from the decomposition.

F.2 The exact weighted distance needed to replace a leader

Suppose b is positive with unique leader k. Define

$$d(b) = \min_{j\neq k} \frac{b_k - b_j}{\sqrt{b_k + b_j - (b_k - b_j)^2}}. \tag{F.2}$$

Every probability vector with a different strict leader has $\|\eta - b\|_b > d(b)$, and this threshold is sharp. Here is a direct proof. Fix a challenger j, put $\Delta = b_k - b_j$, and let $c_i = \mathbf{1}\{i = j\} - \mathbf{1}\{i = k\}$. Since $\sum_i(\eta_i - b_i) = 0$ and $\sum_i b_i c_i = -\Delta$, weighted Cauchy–Schwarz gives

$$|(\eta_j - b_j) - (\eta_k - b_k)| \le \sqrt{b_k + b_j - \Delta^2}\,\|\eta - b\|_b. \tag{F.3}$$

Overtaking requires the left-hand signed difference to exceed Δ. A largest challenger minimizes the bound: for $x = b_k$, $t = b_j < x$, the derivative of $(x-t)^2/[x+t-(x-t)^2]$ is $-(x-t)(3x+t)/[x+t-(x-t)^2]^2 < 0$.

For that challenger set $s = b_k + b_j$ and $D = s - (b_k - b_j)^2$. The allocation

$$\eta_k = \eta_j = \frac{2b_k b_j}{D}, \qquad \eta_i = \frac{s b_i}{D} \quad (i \notin \{k, j\}) \tag{F.4}$$

is positive, sums to one, and has k, j tied for the lead. Substitution yields $\|\eta - b\|_b = d(b)$. An arbitrarily small transfer from k to j gives a strict reversal. Combining Eq. (F.2) with Eq. (F.1) shows that a changed leader requires enough *net* movement, not merely large individual column deviations. This boundary is an analysis result, not an angle threshold used by SFPS.

F.3 An exact relative-entropy boundary

The corresponding minimum of $\mathrm{KL}(\eta\|b)$ over probability vectors for which some challenger ties or exceeds k is

$$\min_{j \neq k} -\log\bigl(1 - b_k - b_j + 2\sqrt{b_k b_j}\bigr). \tag{F.5}$$

To derive it, keep the previous c_i and consider $q_t(i) = b_i e^{tc_i}/Z(t)$, $t \geq 0$. For any $\eta_j \geq \eta_k$,

$$\mathrm{KL}(\eta\|b) = \mathrm{KL}(\eta\|q_t) + t(\eta_j - \eta_k) - \log Z(t) \geq -\log Z(t). \tag{F.6}$$

The normalizer $Z(t) = 1 - b_k - b_j + b_k e^{-t} + b_j e^{t}$ is minimized at $t = \frac{1}{2}\log(b_k/b_j)$. At this value $q_t(k) = q_t(j)$ and both nonnegative terms above vanish when $\eta = q_t$. Taking the union over challengers proves Eq. (F.5). A largest challenger minimizes it and produces a leading tie, since $\sqrt{b_k b_j} \geq b_j \geq b_i$ for other i. This is a constrained exponential-tilt calculation; the local quadratic relation in Eq. (E.2) and the exact boundary serve different purposes.

F.4 From estimation error to a stable row choice

Let W have unique leader k, and set $\delta = \min_{j \neq k}(W_k - W_j)/\sqrt{b_k + b_j}$. For the unnormalized error $e = \widehat{W} - W$, Cauchy–Schwarz gives $|e_j - e_k| \leq \sqrt{b_j + b_k}\|e\|_b$. Hence every misranking, including a tie resolved against k, requires $\|e\|_b \geq \delta$, so

$$\mathbb{P}(\arg\max_i \widehat{W}_i \neq k) \leq \min\left\{1, \frac{\mathbb{E}\|\widehat{W} - W\|_b^2}{\delta^2}\right\}. \tag{F.7}$$

The denominator here differs from Eq. (F.3): unnormalized errors need not sum to zero. For coupled columns the numerator retains all covariances,

$$\mathbb{E}\|\widehat{W} - W\|_b^2 = \frac{1}{n^2}\sum_{j,\ell}\mathbb{E}\langle M_{:j} - W, M_{:\ell} - W\rangle_b. \tag{F.8}$$

Neither an independent-sample law nor an SMC central limit theorem can be imported without its assumptions (Chopin, 2004). The next section proves a bound for the actual coupling.

For comparison only, if columns were independent and each difference $M_{kj} - M_{ij}$ lay in an interval of width R_i, Hoeffding's inequality (Hoeffding, 1963) would give $\mathbb{P}(\arg\max \widehat{W} \neq k) \leq \sum_{i \neq k} \exp[-2n(W_k - W_i)^2/R_i^2]$. We do not apply this independent-column bound to the stratified decoder or to the experimental task scores. Likewise, a deterministic log-score error at most ε perturbs each log-gap by at most 2ε. Such a gap certificate is conditional on estimation accuracy; it is not a correctness certificate or a stopping rule.

G Sequential proposals and convergence

G.1 The estimator is exact even though the proposal is local

Induction on the update in Eq. (14) gives

$$w_i^{(t)} = b_i \prod_{s=1}^{t} \frac{p(y_s \mid z a_i y_{<s})^{\alpha}}{g(y_s \mid y_{<s})}. \tag{G.1}$$

The denominator is the probability of the history actually sampled. A proposal may depend on that column's past without invalidating this identity. At a future leaf it is precisely Eq. (5); no empirical count or parser score replaces the target contribution.

The distinction between a local proposal and a full-future optimum is important. Put $X_i(y) = b_i p_i(y)^\alpha$. If every complete future could be enumerated, the vector-optimal path proposal would be

$$G^*(y) = \frac{\|X(y)\|_b}{\sum_u \|X(u)\|_b}. \tag{G.2}$$

Its next-token conditional sums the norms of *all leaves* below that token. SFPS instead uses the norm of the current prefix contribution, which is immediately available from the model. Computing the exact conditional of Eq. (G.2) would require the remaining-tree calculation that the sampler is trying to avoid.

For a finite illustration, take one candidate and first-token probabilities $(1/2, 1/2)$. The first alternative has a deterministic second token; the second has two equiprobable continuations. Leaf probabilities are $(1/2, 1/4, 1/4)$. With $\alpha = 2$, Eq. (G.2) gives $(2/3, 1/6, 1/6)$, whereas the sequential polar rule gives $(1/2, 1/4, 1/4)$. Both yield valid importance estimates. Thus one-step optimality is not global path optimality, even with only one candidate.

G.2 The labelled stratification used by the decoder

At each future-token position t, draw a fresh uniform permutation σ_t of $\{0, \dots, n-1\}$ and independent jitters $\epsilon_{jt} \sim \mathrm{Uniform}[0, 1)$. Column j receives

$$U_{jt} = \frac{\sigma_t(j) + \epsilon_{jt}}{n}, \qquad y_{jt} = F^{-1}_{g(\cdot|y_{j,<t})}(U_{jt}). \tag{G.3}$$

Permutations and jitters are independent across t. Proposals depend on the fixed candidate pool and their own column history, not on other columns. Finished columns keep their labels and their unused strata are not reassigned. This samples *uniforms* without repeated stratum labels at a position, not future strings without replacement. Different columns may use different inverse CDFs and may produce identical strings.

Lemma G.1 (Marginal law and pairwise dependence)**.** *Each labelled column has marginal future law G. For distinct labels j, ℓ and $n \geq 2$, the joint law of their length-H uniform streams is at total variation distance*

$$\varepsilon_{n,H} = 1 - (1 - 1/n)^H \leq \min\{1, H/n\} \tag{G.4}$$

from two independent uniform streams. Their future and matrix-column pair is at distance at most $\varepsilon_{n,H}$ from two independent draws under G.

Proof. Condition on all histories before position t. The stratum of one label is uniform on the n equal intervals; mixing the uniform jitter over them gives $U_{jt} \sim \mathrm{Uniform}[0, 1)$. The randomness is fresh at every position. The inverse CDF and the chain rule give the marginal path law G.

For a pair of labels, the ordered stratum indices are uniformly distributed over the $n(n-1)$ distinct pairs. Two independent uniforms have this same conditional law given that their strata differ. At one position the joint density of the labelled pair relative to independent uniforms is $n/(n-1)$ on different-stratum pairs and zero elsewhere. Across H positions it is $[n/(n-1)]^H$ on the event that the strata differ at every position, which has independent-stream probability $(1 - 1/n)^H$. The distance between a law and its conditioning on an event is one minus the event probability, proving Eq. (G.4).

For fixed pool and horizon, each column's tokens and weights are a deterministic measurable function of its own stream. Total variation cannot increase under this mapping. Unused uniforms after termination may be retained as auxiliary variables, so the same bound covers early EOS. □

This argument uses the actual time-refreshed stratification. It does not assume negative covariance of all integrands. General comparisons between stratification or resampling schemes require their corresponding conditions (Douc et al., 2005).

G.3 A BOUND FOR THE ACTUAL DEPENDENT MATRIX

The finite future tree and full support imply a finite constant C with $\|M_{:j}\|_b \le C$ for every leaf. The polar rule also gives an explicit, though conservative, bound when $\alpha \ge 1$:

$$C \le m^{H/2}. \tag{G.5}$$

Indeed, at a live step, nonnegativity and $\sum_x p_i(x \mid h)^\alpha \le 1$ imply

$$\sum_x \|f(x)\|_b \le \sum_i \frac{w_i}{\sqrt{b_i}} \sum_x p_i(x \mid h)^\alpha \le \sum_i \frac{w_i}{\sqrt{b_i}} \le \sqrt{m}\,\|w\|_b. \tag{G.6}$$

Proposition 4.3 and $\|b\|_b = 1$ prove Eq. (G.5) by induction. With ended actions, apply the same argument to the live subvector and add the constant squared norm of the ended rows. Their total is still at most m^H. A renormalization of the live reference multiplies every token radius by a common factor and leaves the proposal unchanged.

Theorem G.2 (Mean-square bound under fresh stratification)**.** *Under the conditions of Theorem 5.1, with $\|M_{:j}\|_b \le C$, the actual stratified row estimator satisfies*

$$\mathbb{E}\|\widehat{W} - W\|_b^2 \le \frac{C^2}{n} + \frac{n-1}{n} C^2 \varepsilon_{n,H} \le \frac{(H+1)C^2}{n}. \tag{G.7}$$

The same bound holds with early termination and exactly known ended-action rows.

Proof. Every column has mean W by Proposition D.1. Its centered second moment is $\mathbb{E}\|M_{:j}\|_b^2 - \|W\|_b^2 \le C^2$. For distinct labels, the function $\langle M_{:j}, M_{:\ell}\rangle_b$ lies in $[0, C^2]$, since matrix entries are nonnegative. Under independent copies with the same marginals its expectation is $\|W\|_b^2$. Lemma G.1 therefore gives

$$|\mathbb{E}\langle M_{:j} - W, M_{:\ell} - W\rangle_b| \le C^2 \varepsilon_{n,H}. \tag{G.8}$$

Here we use the elementary total-variation bound for an expectation of a function with range width C^2. Insert the n diagonal and $n(n-1)$ off-diagonal bounds into Eq. (F.8). For $n = 1$, the diagonal bound suffices. Exact rows and frozen columns are deterministic functions of the same streams, so the argument is unchanged. □

For fixed H, this tends to zero even though the columns are dependent. The coefficient can grow rapidly with horizon. The bound does not predict small error at the experimental rollout count or lower variance than independent sampling.

G.4 PROOF OF DECISION CONVERGENCE AND ITS SCOPE

Proof of Theorem 5.1. For a finite supported tree, C is finite and independent of n. Equation (G.7) proves mean-square convergence. A unique row leader has a positive finite margin δ as in Eq. (F.7). Combining the two bounds makes the misranking probability tend to zero. □

Writing $Z = \sum_i W_i > 0$, continuity of normalization gives

$$\frac{\widehat{W}}{\sum_i \widehat{W}_i} \xrightarrow{\mathbb{P}} \frac{W}{Z}. \tag{G.9}$$

If W has ties, the selected row converges to its maximizing *set* in the sense that the probability of choosing a strict nonmaximizer vanishes. Convergence to one particular tied label is not asserted. These claims concern the finite-pool target, not the correctness of its leader.

Increasing n refines an estimate; increasing the horizon H changes the target B_i. More precisely, extending a short future h by r tokens gives

$$B_i^{H+r} = B_i^H\, \mathbb{E}_{h \sim p_i(h)^\alpha / B_i^H} C_i(h), \tag{G.10}$$

where $C_i(h)$ is the additional continuation power sum, or one if h has ended. For $\alpha > 1$, each $C_i(h) \le 1$, but ratios across candidates need not move monotonically. If all these factors lie in $[c, C_0]$ with $c > 0$, a short-horizon log-leader gap exceeding $\log(C_0/c)$ cannot reverse. Without such a restriction, longer lookahead can change either comparison. No horizon-increasing correctness theorem follows from fixed-horizon consistency.

G.5 Candidate coverage and the within-row continuation

The row theorem conditions on which candidates were proposed. For a set of useful openings $\mathcal{A}_+$, independent proposal draws omit it with probability $(1-q(\mathcal{A}_+ \mid z))^K$. No number of future columns can recover a missing row. Lemma D.2 establishes candidate-weight convergence as $K \to \infty$ and consistent selection of the unique maximizer of $S(a)$ as $K, n \to \infty$. We next analyze the continuation selected within a row.

For fixed pool, write the importance score of leaf y in row i as $J_i(y) = b_i p_i(y)^\alpha / G(y)$. Lemma G.1 applied to the indicator of a leaf gives

$$\mathrm{Var}\left(\frac{1}{n}\sum_j \mathbf{1}\{y_j = y\}\right) \le \frac{1}{n} + \frac{n-1}{n}\varepsilon_{n,H}. \tag{G.11}$$

The mean is $G(y)$, so every positive-probability leaf is observed with probability tending to one. On the finite tree, a union bound extends this to all supported leaves. Consequently, once the correct target row is selected, the within-row maximum converges to the maximizing set of J_i. With unique row and leaf maximizers, the complete block converges in probability to this finite-pool row-first decision. This leaf maximizes an *importance contribution*; it need not maximize the raw likelihood $p_i(y)$.

Each committed block changes the next comparison. The result concerns block decisions, not a stationary MCMC law, an unobserved correct answer, or a trained policy. It needs no repair or stopping mechanism.

G.6 Convergence through a bounded sequence of block commitments

The fixed-pool theorem can be extended to the whole generated string, provided the comparison target is stated precisely. Fix the vocabulary, action proposal, candidate count K, block horizons, and finite committed-output cap. Let $\mathcal{D}_n$ be SFPS with n shared futures per block. Define a reference decoder $\mathcal{D}_\infty$ that draws candidate openings from the *same* proposal at each prefix, but uses exact integration for the row comparison:

$$i_\infty = \arg\max_i b_i B_i, \qquad y_\infty = \arg\max_{y: G_b(y)>0} \frac{b_{i_\infty} p_{i_\infty}(y)^\alpha}{G_b(y)}. \tag{G.12}$$

It commits $a_{i_\infty} y_\infty$, or only the action if it already ends. The law G_b is the same sequential polar proposal defined from this prefix and pool. It does not depend on n; changing n changes the coupling and number of draws, not the proposal for an individual labelled column.

The reference retains the finite decoder's row target and importance-based future readout, and remains random through its candidate pool.

Theorem G.3 (Finite-cap output convergence)**.** *Assume full proposal support and exact arithmetic. Suppose the maximizing row and, for a live maximizing row, the maximizing future in Eq.* (G.12) *are unique for every prefix and candidate pool reachable by $\mathcal{D}_\infty$. Suppose each nonterminal commitment advances the output by at least one token. Then the distribution of the complete response under $\mathcal{D}_n$ converges in total variation to that under $\mathcal{D}_\infty$ as $n \to \infty$.*

Proof. First fix a reachable prefix and pool. The row error probability tends to zero by Theorem 5.1. Let y_∞ be the reference maximizing future, with $q_* = G_b(y_\infty) > 0$. If N_* counts the labelled columns equal to this leaf, Eq. (G.11) and Chebyshev's inequality give

$$\mathbb{P}(N_* = 0) \le \frac{1/n + ((n-1)/n)\varepsilon_{n,H}}{q_*^2} \longrightarrow 0. \tag{G.13}$$

When the correct row is selected and the unique maximizing leaf has been observed, the finite and reference commitments agree. Ended actions require only the row event. Hence their one-block disagreement probability tends to zero.

At a finite cap there are finitely many prefixes, and fixed K and finite action horizons give finitely many candidate pools at each prefix. The supremum ϵ_n of these one-block error probabilities over

reference-reachable states therefore tends to zero. Couple the two decoders by using the same candidate-proposal randomness whenever their committed prefixes agree. Their candidate pools then coincide until the first disagreement. If at most C nonterminal blocks fit under the cap, a conditional union bound gives

$$\mathbb{P}\{\mathcal{D}_n \neq \mathcal{D}_\infty\} \leq C\epsilon_n \longrightarrow 0. \tag{G.14}$$

The total variation distance between the two output laws is at most the disagreement probability under any coupling, proving the claim. □

Rare maximizing futures can make the finite-sample constants large. This is convergence to the *same* finite-horizon block decoder, not to a global power law, an RL policy, or a correct answer. Without uniqueness, convergence to a maximizing set does not identify one output law under the deterministic tie rule.

H Implementation and evaluation conventions

In Algorithm 1, L and H are the maximum opening and continuation lengths in tokens. The draws $a^{(1)}, \ldots, a^{(K)}$ are independent samples from q, $\varnothing$ is the empty string, and w_{ij} is row i's weight in continuation j. At token step t, $\mathbf{F}_j$ and g_j denote its matrix and proposal. Sampling uses $x_j = \mathrm{CDF}_{g_j}^{-1}(U_{jt})$, where $\mathrm{CDF}_{g_j}^{-1}$ is the inverse cumulative distribution and $U_{jt} \in [0, 1)$ comes from the randomized stratification in Appendix G.2.

H.1 Termination, support, and numerical evaluation

An opening ending with the end-of-sequence (EOS) token has the deterministic row $M_{ij} = b_i$ and is excluded from future-token proposals. Live rows use $b_i^{\mathrm{live}} = b_i / \sum_{\ell \in \mathrm{live}} b_\ell$: this common rescaling leaves the normalized proposal unchanged. Stored importance weights keep their original scale for comparison with ended rows.

Futures stop at EOS or their horizon and freeze without relabelling. Horizons shorten at the output cap. With no remaining future its factor is one; with all actions ended every row is exact. Readout commits the selected pair, breaking ties by stored order, without rollback or an angle gate.

Products are evaluated in log space. At a live token position, let $\ell_i = \log w_i$ and $\ell_i(v) = \ell_i + \alpha \log p(v \mid z a_i h)$. The implementation computes

$$\log r(v) = \tfrac{1}{2} \operatorname{logsumexp}_i[2\ell_i(v) - \log b_i], \qquad \log g(v) = \log r(v) - \operatorname{logsumexp}_u \log r(u). \tag{H.1}$$

After selecting v, update $\ell_i \leftarrow \ell_i(v) - \log g(v)$ and aggregate rows with log-sum-exp. Any subtracted log scale must be restored before comparing columns. Independent column normalization without that scale changes the target.

Importance sampling requires support coverage (Hesterberg, 1995). The polar proposal uses all nonzero model probabilities, without top-k or nucleus truncation. Floating-point underflow is outside the exact-arithmetic theory. Table 5 distinguishes probability and weight arithmetic.

H.2 Operation counts and what the two matrices require

If the action and future windows are full, one round draws $KL + nH$ tokens and commits $L + H$ of them. Thus its nominal generated-token expansion is

$$\frac{KL + nH}{L + H}. \tag{H.2}$$

Sharing avoids n separate futures per action, not scoring in each context. For m distinct live actions, it requires up to mnH candidate–future token evaluations. The m-row vocabulary matrix can be batched and discarded after each proposal; the final matrix stores mn weights, besides text and model state.

Generated tokens, scored positions, and wall-clock time are different measures. Latency also depends on caching, prefix length, hardware, and batching; none is a universal FLOP ratio.

Relation to other sequence-sampling constructions. Sequence samplers include MCMC Power (Karan & Du, 2025; Metropolis et al., 1953; Hastings, 1970), QUEST (Faria et al., 2024), twisted sequential Monte Carlo (Zhao et al., 2024), and later power samplers (Ji et al., 2026; Azizi et al., 2026; Nguyen et al., 2026; Zhou et al., 2026); compute-allocation methods provide complementary approaches (Snell et al., 2024; Wu et al., 2024). SFPS shares continuations across candidate rows and uses a fixed-reference vector norm. Other methods can also produce position shifts. Costs compare only with the tested MCMC Power implementation.

H.3 Experimental settings and resources

Table 5: SFPS **experimental settings.** GPQA uses $K = 5$ and $n = 5$.

Item	Setting
Candidate draws / shared continuations	$K = 8$ / $n = 4$
Opening / continuation horizons	$L = 32$ / $H = 96$ tokens
Power exponent / opening temperature	$\alpha = 4$ / $\tau = 0.6$
Maximum response length	3,072 committed tokens
Qwen2.5-Math-7B implementation	
Opening / continuation batch size	4 / 4
Model / probability / weight arithmetic	bfloat16 / float32 log-softmax / float64 weights
Attention / prefill chunk	SDPA / 128 tokens
Hardware	NVIDIA H100 GPU
Software	PyTorch 2.5.1+cu124; Transformers 4.47.1

Table 6: **Models and public GRPO checkpoints.** GRPO checkpoints are released models trained on mathematical data.

Model	Base and GRPO checkpoint identifiers	GRPO training data
Qwen2.5-Math-7B (Yang et al., 2024)	**Base:** `Qwen/Qwen2.5-Math-7B` **GRPO:** `stellalisy/rethink_rlvr_reproduce-ground_truth-qwen2.5_math_7b-lr5e-7-kl0.00-step150` (stellalisy, 2025)	DeepScaleR mathematics (Shao et al., 2025)
Qwen2.5-Math-1.5B (Yang et al., 2024)	**Base:** `Qwen/Qwen2.5-Math-1.5B` **GRPO:** `kong3125/Qwen2.5-MATH-1.5B-BASE-grpo` (kong3125, 2025)	`jhn9803/hendrycks-math-with-answers`
Llama-3.2-3B-Instruct (Meta, 2024)	**Base:** `meta-llama/Llama-3.2-3B-Instruct` **GRPO:** `TMLR-Group-HF/GT-Llama-3.2-3B-Instruct-MATH` (TMLR-Group-HF, 2025)	MATH training set

Table 7: **Evaluation data and cost accounting.** Each method returns one final answer per problem.

Benchmark	Evaluation data	Problems	Metric
MATH500	`HuggingFaceH4/MATH-500` (Lightman et al., 2023; HuggingFaceH4, 2024)	500	Accuracy
AI-MO AMC	`AI-MO/aimo-validation-amc` (AI-MO, 2024)	83	Accuracy
GPQA Diamond	Diamond subset of `Idavidrein/gpqa` (Rein et al., 2023)	198	Accuracy
HumanEval	Python tasks with unit tests (Chen et al., 2021)	164	pass@1

Quantity	Convention
Generated tokens	All sampled openings and continuations, including uncommitted text
Inference time	Measured wall-clock latency
Base-relative cost	Per-task ratio to the Base mean, before rounding
Power-relative cost	Equal-weight mean of the four task-wise ratios to MCMC Power

I Evidence and selection: experimental analysis

We examine how future evidence changes candidate preferences, how the matrix combines that evidence, and how polar sampling estimates it. The analyses below connect the decoder's operations to its observed choices.

I.1 Evaluation setup

All reported benchmark and mechanism-analysis results are averaged over three random seeds. Mechanism analyses use the 500 Qwen2.5-Math-7B MATH500 responses underlying the SFPS result, comprising 2,820 decoding blocks. Of these, 1,880 blocks have at least two distinct candidates (*competitive blocks*); 1,864 have four continuation columns. Accuracy and inference costs are reported in Tables 1–2.

Accuracy intervals use paired question bootstrap. Block-level intervals resample whole questions with their blocks; local MSE is averaged within each question and then across questions. All displayed intervals are 95% intervals. The four-task cost averages use the accounting in Table 7.

I.2 Why aggregate future evidence?

Initial-b selects $\arg\max_i b_i$, before aggregating future evidence. **Equal-column** selects $\arg\max_i \sum_j \eta_{ij}$, giving every normalized column equal weight. Both keep the original generation settings and within-row continuation selection. Their accuracies are 74.0% and 74.2%, versus 77.6% for SFPS (Table 3). The net gains are 18 answers (29 gains, 11 losses) and 17 answers (25 gains, 8 losses), respectively. Thus using aggregate evidence and preserving column weights each improves the evaluated decoder.

I.3 How future evidence changes the preferred candidate

At a fixed block, b is the initial candidate distribution after proposal and duplicate corrections, and $\widehat{\pi}$ is the normalized row support. Write $\widehat{B}_i = \widehat{W}_i / b_i$ for the estimated future support of candidate i. Equation (5) then gives

$$\log \frac{\widehat{\pi}_j}{\widehat{\pi}_k} = \underbrace{\log \frac{b_j}{b_k}}_{\text{initial preference}} + \underbrace{\log \frac{\widehat{B}_j}{\widehat{B}_k}}_{\text{future evidence}} .$$

For the initial leader k, candidate j overtakes it when its future advantage exceeds $\log(b_k/b_j)$. This is the local–future competition of Proposition 2.1, applied to the weights used by the decoder.

Figure 4(a) plots the same selected candidate before and after aggregation. Orange diamonds mark 418 strict promotions out of 1,880 competitive blocks (22.23%): 292 promote the runner-up and 126 promote a candidate initially ranked third or lower. There is one additional initial-tie resolution; the 419 leader changes occur in 268 questions. Mean selected-candidate support in these changes rises from 0.1844 to 0.8463. A further 701 blocks reorder lower-ranked candidates while retaining the leader. Future evidence therefore changes both the strength and the ordering of candidate preferences.

Proposal and duplicate corrections precede this comparison and change the raw-likelihood leader in 137 blocks. The promotion rate measures the subsequent $b \to \widehat{\pi}$ update. Row-sum and maximum-cell selection agree in 97.71% of competitive blocks; 1,081 of the 1,864 four-column matrices contain repeated continuation strings.

I.4 Why keep a continuation's weight and direction together?

For a saved matrix M, let μ_j be column j's total contribution and η_{ij} its normalized support for candidate i. The selected row maximizes $\sum_j \mu_j \eta_{ij}$ (Eq. (5)). We change these stored contributions to identify what affects the decision.

Column size. Deleting the largest-weight column changes 26.82% of selections (500/1,864; interval [24.81, 29.14]), compared with 0.59% for the smallest (11/1,864; [0.27, 0.95]). In Figure 4(c), farther right means that removing the column has greater influence. The contrast shows that continuations make unequal contributions to the choice.

We also vary the size exponent in $R_i(\gamma) = \sum_j \mu_j^\gamma \eta_{ij}$. Equal weighting ($\gamma = 0$) changes 22.18% of selections, whereas tested exponents from 0.5 to 2 change at most 2.07% relative to SFPS ($\gamma = 1$). Table 8 and Figure 5(a) show that keeping column size matters, while most selections persist under nearby weightings.

Table 8: **Column-weight exponent sweep.** Changes are relative to $\gamma = 1$; "First" uses the first competitive block of each question.

γ	Changed blocks	Changed (%)	First changed (%)	Mean entropy (nats)
0.00	417	22.18	21.40	0.5124
0.25	99	5.27	2.60	0.3826
0.50	39	2.07	1.20	0.2961
0.75	12	0.64	0.40	0.2436
1.00	0	0.00	0.00	0.2083
1.50	14	0.74	0.60	0.1638
2.00	22	1.17	0.80	0.1370

Weight–direction pairing. A permutation σ of column indices gives $R_i(\sigma) = \sum_j \mu_{\sigma(j)} \eta_{ij}$: the same weights and directions are present, but their pairing changes. Across all 24 permutations of each four-column matrix, 24.73% of assignments change the selected row on average, and 47.75% of matrices admit at least one change. The rates are higher among strict promotions (Table 9). Together with the exponent sweep, this distinguishes modest changes in weighting from changes to which candidates the evidence supports.

Table 9: **Weight–direction pairing.** Mean changed averages over all 24 permutations; any changed records at least one changed selection.

Four-column group	Matrices	Mean changed (%)	Any changed (%)
All competitive	1,864	24.73	47.75
First competitive block	500	25.10	51.20
Strict promotion	418	62.57	94.02
Other blocks	1,446	13.80	34.37

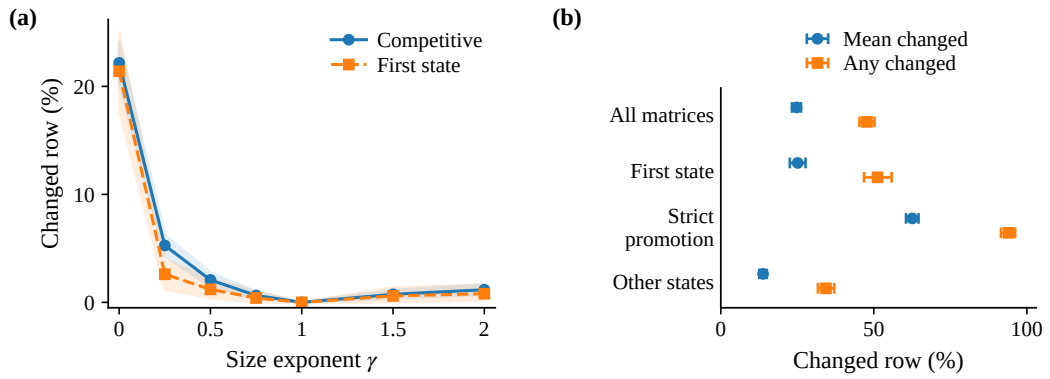


Figure 5: **Column size and evidence allocation.** (a) Horizontal: size exponent γ; vertical: selections changed from $\gamma = 1$. Blue circles/solid line use all competitive blocks; orange squares/dashed line use the first per question. Lower means greater agreement, with zero at the reference $\gamma = 1$. (b) Rows are matrix groups; horizontal position measures changed selections. Blue circles average over permutations; orange squares count matrices with any change. Farther right means greater dependence on the pairing. Bands and bars are 95% question-bootstrap intervals. *Column size and the candidates it supports jointly determine the decision.*

I.5 Direction changes choices at fixed strength

For the centered-log update $v' = cv + d$, we hold c, $\|d\|_2$, and $\|v'\|_2$ fixed and vary the residual direction. The study uses blocks with $m \geq 4$, a unique initial leader, $c > 0$, and $\|d\|_2 > 0$. Eight Gaussian vectors per block are projected onto the zero-sum subspace orthogonal to v and rescaled to the observed residual length. Across 1,086 blocks, 56.53% of these alternatives change the selected row, and 92.45% of blocks admit a different selection (Table 10). Equal strength can therefore produce different choices: the direction retained by the contribution vector matters.

Table 10: **Matched-strength directional comparisons.** Each eligible block uses eight alternative residual directions.

Quantity	Value
Eligible blocks / questions	1,086 / 477
Directions per block / total alternatives	8 / 8,688
Alternative direction selects a different row	56.53%
Blocks with at least one different selection	92.45%
Observed / alternative changes initial leader	29.37% / 51.42%
Mean support entropy, observed / alternative (nats)	0.22149 / 0.03298
Maximum relative error in matched final radius	1.44×10^{-15}

Scaling accumulated evidence gives scores $\log b_i + t \log \widehat{B}_i$, with $t \geq 0$ controlling its influence. The first-crossing analysis finds 418 crossings below $t = 1$, 366 above it, 1,095 unique-leader blocks without a detected finite crossing, and one initial tie. The 418 crossings below one recover the strict promotions above, linking the observed choices to the boundary in Proposition 2.1.

I.6 How polar sampling improves evidence estimation

Let $F_0 = \sum_x f(x)$ be the summed token-contribution vector. For a supported proposal g, we evaluate the relative weighted-vector mean-squared error

$$R(g) = \frac{\sum_x \|f(x)\|_b^2 / g(x) - \|F_0\|_b^2}{\|F_0\|_b^2}. \tag{I.1}$$

The loss uses the fixed initial weights b. Its norm is invariant to splitting an identical candidate into proportional copies (Appendix E). On each matrix, Polar uses $g(x) \propto \|f(x)\|_b$, Mass uses $g(x) \propto \sum_i f_i(x)$, and Euclidean uses $g(x) \propto \|f(x)\|_2$.

Figure 4(b) evaluates 4,266 active probes before tokens 1, 17, and 49 of the first continuation in the first three blocks. The problem-balanced means and 95% intervals are 0.194507 [0.180454, 0.209218] for Polar, 0.908607 [0.844733, 0.980586] for Mass, and 0.903042 [0.840026, 0.974465] for Euclidean. Polar reduces the measured error by 78.6% relative to Mass. This quantifies the variance saved by the norm-based proposal in the shared evidence update (Proposition 4.3).

I.7 Where generated responses lie under the base model

For the response-distribution analysis, a separate MATH500 baseline batch is scored alongside the SFPS responses using Qwen2.5-Math-7B and a common plain prompt. For response $y = (y_1, \ldots, y_T)$, with T content tokens, prompt z, and vocabulary $\mathcal{V}$, we compute

$$\mathrm{LL}(y) = \frac{1}{T} \sum_{t=1}^{T} \log p(y_t \mid z, y_{<t}), \tag{I.2}$$

$$\mathrm{Conf}(y) = \frac{1}{T} \sum_{t=1}^{T} \sum_{v \in \mathcal{V}} p(v \mid z, y_{<t}) \log p(v \mid z, y_{<t}). \tag{I.3}$$

LL measures probability assigned to the generated tokens; Conf is negative next-token entropy. Higher scores mean stronger base-model support and lower uncertainty, respectively. Scoring uses the full, untempered vocabulary, excludes prompt and EOS tokens, and retains preceding context across 64-token chunks in bfloat16.

For this batch, Base uses temperature 1.0; Low temperature and GRPO use 0.25. Power uses exponent 4, proposal temperature 0.25, 10 MCMC steps per stage, and 192-token blocks. Baseline generation uses top-$k = 50$, top-$p = 1$, and a 3,072-token cap; GRPO generates with its chat template.

Figure 6 shows densities on $[-0.3, 0]$; Figures 7–8 retain the full distributions. Histogram normalization includes all scorable responses, including the left tails. The rightward concentration of SFPS responses complements the candidate-level analysis: future-supported choices coexist with high base-model likelihood and low uncertainty, consistent with the likelihood weighting in Eq. (4).

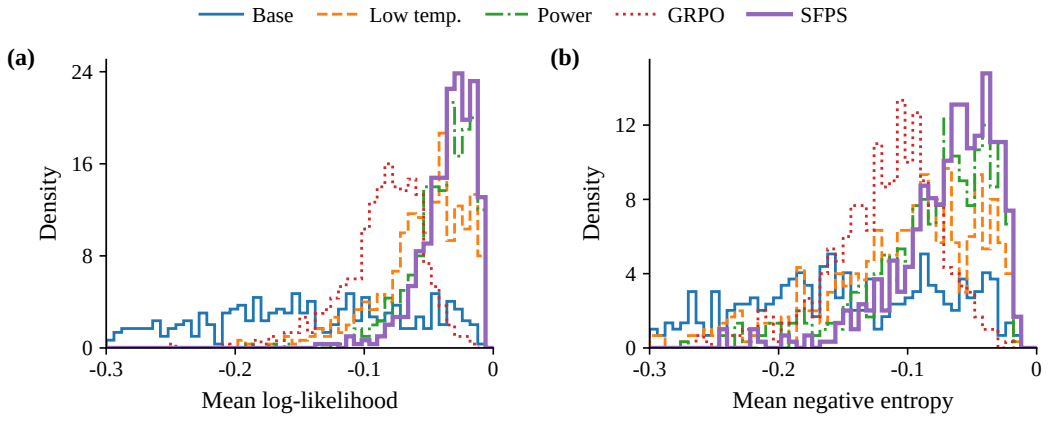


Figure 6: **Response-score densities.** Blue solid: Base; orange dashed: low temperature; green dash-dot: Power; red dotted: GRPO; purple solid: SFPS. Horizontal axes show (a) mean token log-likelihood and (b) mean negative vocabulary entropy. Rightward means stronger support and lower uncertainty. Height is density: taller bins contain more responses per unit score. SFPS *concentrates responses near the high-support, low-uncertainty end.*

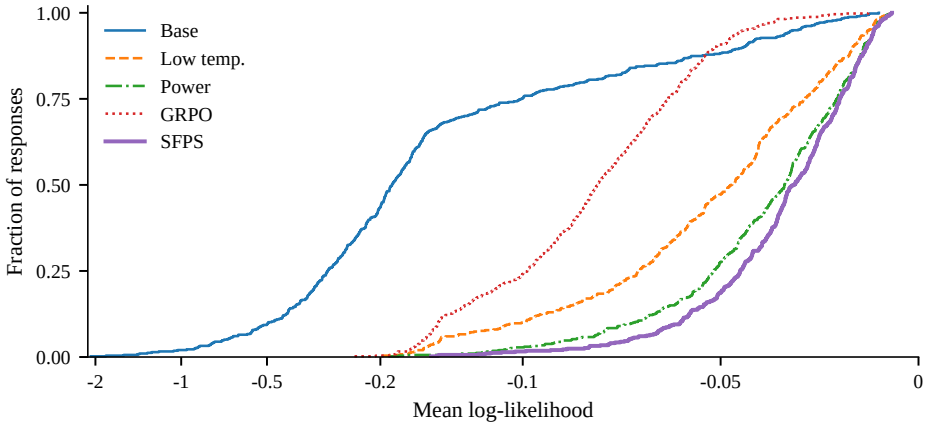


Figure 7: **Full likelihood distributions.** Colors and line styles follow Figure 6. Horizontal: mean token log-likelihood on a symmetric-log scale; vertical: fraction of responses at or below that score. At the same height, farther right means higher likelihood; at the same score, lower means fewer low-scoring responses. SFPS *and Power avoid much of Base's low-likelihood tail, with* SFPS *slightly farther right through much of the central range.*

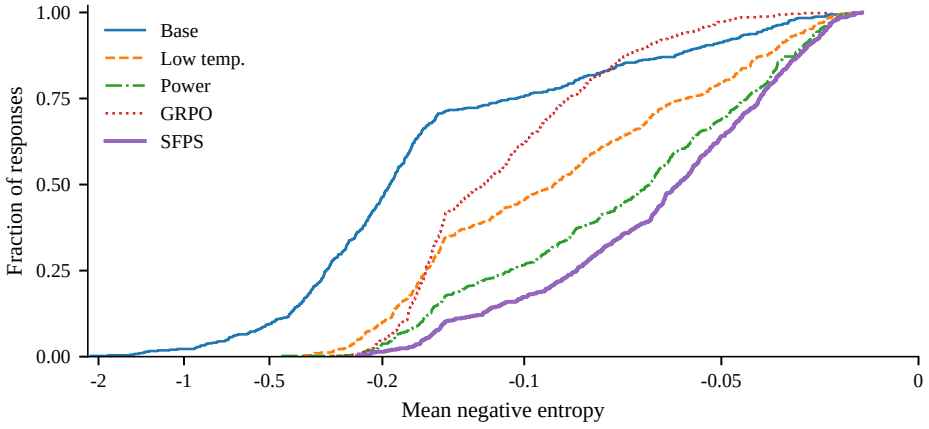


Figure 8: **Full negative-entropy distributions.** Colors and line styles follow Figure 6. Horizontal: mean negative vocabulary entropy, with the long left tail compressed; vertical: fraction of responses at or below each score. Farther right at the same height means lower next-token uncertainty. Steeper rises mean more concentrated scores. SFPS *shifts the central range toward lower uncertainty, extending the pattern in Figure 6(b) across the full distribution.*