Reasoning with Sampling: Your Base Model is Smarter Than You Think

Selim-Antoine Lali   Gabriel Bus   Théo Cadène

Introduction to Generative AI Models — Final Project

February 2, 2026

Paper: Aayush Karan, Yilun Du (2025) — arXiv:2510.14901
Repo : github.com/SelimLali/LLM-reasoning

Outline

Motivation: what does RL really add?

- Frontier "reasoning" LLMs are often built via post-training RL (especially RLVR / GRPO).
- Open debate: does RL create *new* reasoning behaviors or mostly *sharpen* the base distribution?
- Core question of the paper:

    Can we unlock similar reasoning gains ***without training****, using only inference-time sampling and compute?*

Main contributions

1. Define a principled sharpening target: the **power distribution** p^{α} $(\alpha > 1)$.
2. Propose a **training-free MH sampler** adapted to autoregressive generation (blockwise resampling).
3. Show strong results across tasks/models: near GRPO on math, often **better out-of-domain**, with **less diversity collapse**.

Benchmarks: MATH500, HumanEval, GPQA (Diamond), AlpacaEval 2.0.

Preliminaries: autoregressive LLM

- Vocabulary $\mathcal{X}$, sequence $x_{0:T} = (x_0, \ldots, x_T)$.
- Joint distribution factorization:

$$p(x_{0:T}) = \prod_{t=0}^{T} p(x_t \mid x_{<t}).$$

- Standard decoding samples tokens sequentially from $p(x_t \mid x_{<t})$.

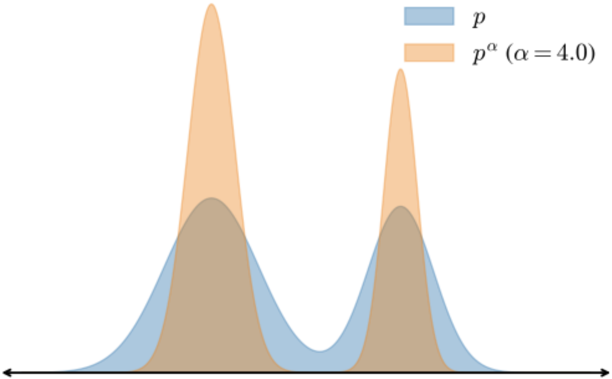
Goal: improve single-shot correctness *without* training, curated data, or external verifiers at inference.

Power distributions as explicit sharpening

- Target distribution:

$$\pi(x) \propto p(x)^{\alpha}, \qquad \alpha \geq 1.$$

- For $\alpha > 1$: concentrates mass on higher-likelihood sequences under the base model.
- Intuition: many correct reasoning traces already exist in p, but are under-sampled.



Why low temperature is *not* sampling from p^α

Common heuristic: temperature / local tempering

$$p_{\text{temp}}(x_t \mid x_{<t}) = \frac{p(x_t \mid x_{<t})^\alpha}{\sum_{x' \in \mathcal{X}} p(x' \mid x_{<t})^\alpha}, \qquad \tau = \frac{1}{\alpha}.$$

Key proposition

Sampling sequentially from $p_{\text{temp}}(x_t \mid x_{<t})$ does **not** produce samples from $\pi(x) \propto p(x)^\alpha$ in general.

Reason: p^α conditionals depend on a *sum of exponentiated probabilities over future completions*, whereas temperature does a purely local reweighting.

A "critical window" toy example (2 tokens)

Let $\mathcal{X} = \{a, b\}$, sequences $\{aa, ab, ba, bb\}$, and

$$p(aa) = 0.00, \quad p(ab) = 0.40, \quad p(ba) = 0.25, \quad p(bb) = 0.25, \qquad \alpha = 2.$$

- Under $\pi(x) \propto p(x)^2$: choosing $x_0 = a$ is favored due to the strong future path ab.
- Under low-temperature: $x_0 = b$ can be favored since it has *two* medium-quality futures.

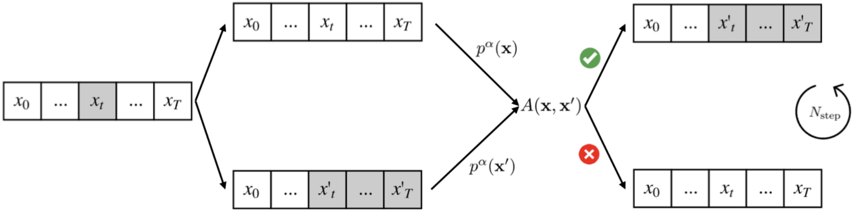
Takeaway: local tempering can prefer "many average futures" over "few very good futures".

Sampling an unnormalized target: Blockwise MH Power Sampling

- Direct sampling from $\pi(x) \propto p(x)^\alpha$ is intractable (normalization sums over $\mathcal{X}^T$).
- MH builds a Markov chain with stationary distribution π.
- Proposal $x' \sim q(\cdot \mid x)$, accept with:

$$A(x', x) = \min\left(1, \frac{\pi(x')\, q(x \mid x')}{\pi(x)\, q(x' \mid x)}\right).$$

- Full-sequence MH mixes poorly $\Rightarrow$ use **block schedule**.
- At each stage: extend prefix by proposal sampling, then run MH steps that resample a suffix starting at random index.



Algorithm 1: Power Sampling (paper)

Algorithm 1 Power Sampling for Autoregressive Models (paper's Algorithm 1)

Require: Base LLM p, proposal LLM p_{prop}, power α, max length T
Require: Block size B, MH steps N_{MCMC}

Define unnormalized targets $\pi_k(x_{0:kB}) \propto p(x_{0:kB})^\alpha$
for $k = 0$ to $\lceil T/B \rceil - 1$ **do**
  Initialize by extending current prefix $x_{0:kB}$ with proposal sampling:
  for $t = kB + 1$ to $(k+1)B$: sample $x_t^{(0)} \sim p_{\text{prop}}(\cdot \mid x_{<t})$
  Set current state $x \leftarrow x^{(0)}$
  for $n = 1$ to N_{MCMC} **do**
    Sample an index m uniformly from $\{1, \dots, (k+1)B\}$
    Construct proposal x' by keeping prefix $x'_{0:m-1} = x_{0:m-1}$ and resampling suffix:
    for $t = m$ to $(k+1)B$: sample $x'_t \sim p_{\text{prop}}(\cdot \mid x'_{<t})$
    Compute acceptance:

$$A(x', x) \leftarrow \min\left(1, \frac{\pi_{k+1}(x')}{\pi_{k+1}(x)} \cdot \frac{p_{\text{prop}}(x \mid x')}{p_{\text{prop}}(x' \mid x)}\right)$$

    Draw $u \sim \text{Uniform}(0,1)$
    if $u \leq A(x', x)$ **then**
      accept: $x \leftarrow x'$
    end if
  end for
  Fix the new prefix: $x_{0:(k+1)B} \leftarrow x$
end for
return $x_{0:T}$

Inference-time compute scaling

- Compute knob: N_{MCMC} (number of MH refinement steps per block).
- Expected number of generated tokens (paper's estimate):

$$\mathbb{E}[\#\text{generated tokens}] \approx \frac{N_{\text{MCMC}}\, T^2}{4B}.$$

- Trade-off: more compute $\Rightarrow$ better approximation to p^α $\Rightarrow$ higher single-shot accuracy.

Why power distributions can improve reasoning

- Many reasoning failures come from **pivotal tokens** early in the chain ("critical windows").
- Power distributions favor tokens that lead to **fewer but stronger** future completions.
- Temperature sampling can favor tokens with **many average** continuations (undesirable in critical windows).

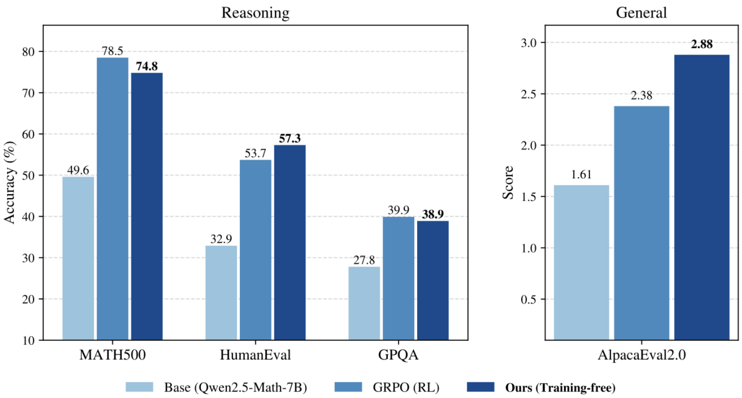
Practical interpretation

Power sampling is a training-free way to emulate distribution sharpening effects, closer to what RL might be doing.

Experimental setup (paper)

- Tasks:
  - **MATH500** (math accuracy),
  - **HumanEval** (unit tests),
  - **GPQA Diamond** (science MCQ),
  - **AlpacaEval 2.0** (LLM judge helpfulness).
- Models: Qwen2.5-Math-7B, Qwen2.5-7B, Phi-3.5-mini-instruct.
- Baselines: Base sampling, Low-temp, **GRPO** (RL, trained on MATH), Training-free MH
- Typical hyperparams: $T_{\text{max}} = 3072$, $B = 192$, $\alpha = 4$, proposal distribution chosen as the base model with temperature $\frac{1}{\alpha}$.

Main results (paper): headline plot



- Power sampling nearly matches GRPO on in-domain math.
- Often **outperforms GRPO out-of-domain** (e.g., HumanEval / AlpacaEval), suggesting better generalization.

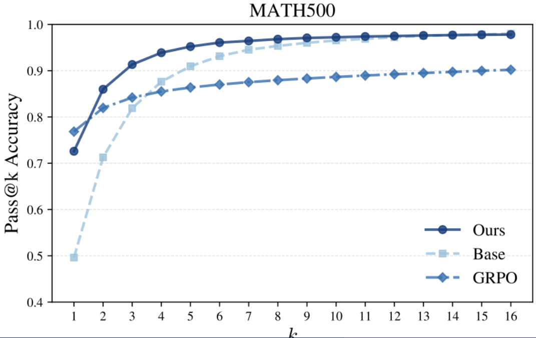
Main results table from the paper

Model	Method	MATH500	HumanEval	GPQA	AlpacaEval 2.0
Qwen2.5-Math-7B	Base	0.496	0.329	0.278	1.61
	Low-temp	0.690	0.512	0.353	2.09
	Power Sampling	**0.748**	**0.573**	**0.389**	**2.88**
	GRPO (MATH)	**0.785**	**0.537**	**0.399**	**2.38**
Qwen2.5-7B	Base	0.498	0.329	0.278	7.05
	Low-temp	0.628	0.524	0.303	5.29
	Power Sampling	**0.706**	**0.622**	**0.318**	**8.59**
	GRPO (MATH)	**0.740**	**0.561**	**0.354**	**7.62**
Phi-3.5-mini-instruct	Base	0.400	0.213	0.273	14.82
	Low-temp	0.478	0.585	0.293	18.15
	Power Sampling	**0.508**	**0.732**	**0.364**	**17.65**
	GRPO (MATH)	**0.406**	**0.134**	**0.359**	**16.74**

Pass@k and hyperparameter sensitivity

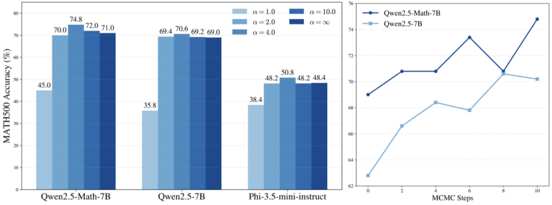
Pass@k (diversity proxy)

- A problem is solved if ≥ 1 of k samples is correct.
- Power sampling improves pass@k vs GRPO for $k > 1$ in the paper.



Hyperparameters

- Strong performance around $\alpha \approx 4$.
- Increasing N_{MCMC} improves accuracy up to ~ 10 steps.

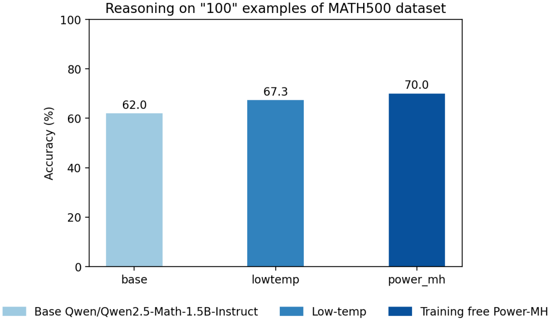


Our reproduction on MATH500 (reduced-scale)

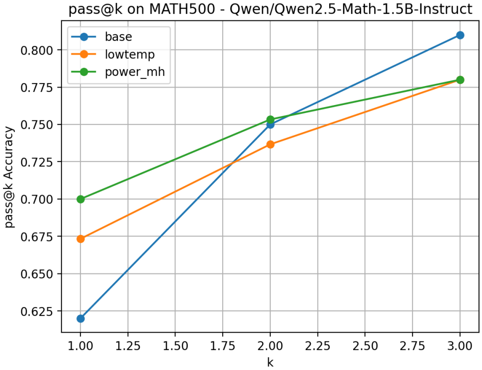
- Goal: reproduce the main MATH500 trends with constrained compute.
- Model: **Qwen/Qwen2.5-Math-1.5B-Instruct** (smaller than paper's 7B).
- Data: subset of **100 problems** (from MATH500), **3 seeds** $\Rightarrow$ pass@k with $k \leq 3$.
- Maximum generation length: 1024 new tokens for all methods.
- Methods:
  - Base sampling,
  - Low-temperature sampling ($\alpha = 4 \Rightarrow \tau = 0.25$),
  - Training-free Power Sampling (Power-MH) ($\alpha = 4$, $B = 192$, $N_{\text{MCMC}} = 3$).

Our results: pass@1 and pass@k

Single-shot accuracy (pass@1)



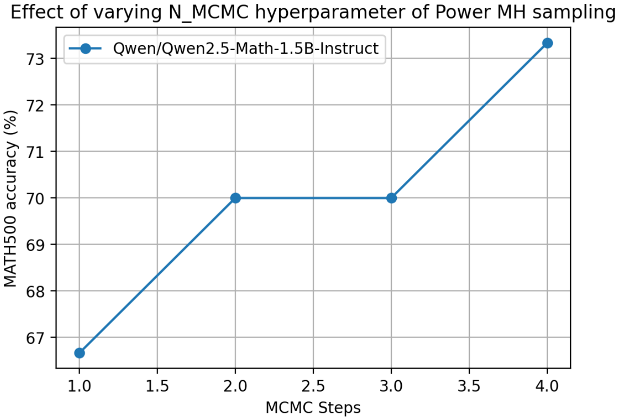
Pass@k (3 seeds)



Power-MH improves over base and slightly over low-temp in our 100-example subset.
Takeaway: the results should be interpreted cautiously. The reduced-scale setup introduces noticeably higher variance: with only 100 problems, a few hard or easy items can significantly shift results.

Effect of N_{MCMC}

- Small sweep on a 30-problem subset (to keep it tractable).
- Fixed: $\alpha = 4$, $B = 192$, $\text{max\_new\_tokens} = 1024$.
- Observed monotonic improvement from $N_{\text{MCMC}} = 1$ to 4.



Limitations & open questions

- **Compute overhead:** multiple resamplings and MH steps (inference-time expensive).
- **Likelihood vs correctness:** higher base-model likelihood correlates with correctness only in some domains; optimal α may vary.
- **Mixing depends on proposal:** proposal quality, block size, and schedule matter.
- **Scope:** strongest when base model already contains latent competence; unknown for truly novel skills.

Conclusion

- Base LLMs may contain more usable reasoning than standard decoding reveals.
- **Power sampling** targets p^{α} and approximates it with blockwise MH resampling.
- Empirically: near GRPO on in-domain math, often better out-of-domain, with improved pass@k (less diversity collapse).
- Big picture: part of "reasoning" can be reframed as **inference-time distribution shaping**.

Questions?

References

A. Karan and Y. Du.
Reasoning with Sampling: Your Base Model is Smarter Than You Think.
arXiv:2510.14901, 2025.