Reject, Resample, Repeat: Understanding Parallel Reasoning in Language Model Inference

Noah Golowich · Fan Chen · Dhruv Rohatgi · Raghav Singhal · Carles Domingo-Enrich · Dylan J. Foster · Akshay Krishnamurthy

Golowich et al. Reject, Resample, Repeat. arXiv:2603.07887, 2026.

Presentation overview

1. Set up guided generation as reward-tilted sampling.
2. Define the process reward model (PRM) as an approximate value function.
3. Analyze Sequential Monte Carlo (SMC) through coverage and PRM divergence.
4. Understand Sequential Monte Carlo with Rejection Sampling (SMC-RS) and myopic lower bounds.
5. Interpret experiments: sampling error vs. task accuracy.

One-slide thesis

Parallel inference-time reasoning can be studied as particle filtering over partial language-model generations.

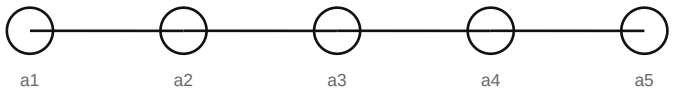
- Sequential Monte Carlo (SMC) gives a principled abstraction of aggregation and pruning.
- Two quantities control sampling error: action-level coverage and process reward model (PRM) accuracy.
- Modified algorithms can avoid some SMC pathologies.
- But sampling-theoretic success does not fully explain math accuracy.

1. Guided generation as sampling

Base model and reference distribution

$$\pi_{\mathrm{ref}} \in \Delta(\mathcal{A}^H)$$

Reference distribution: the distribution over complete H-step generations produced by the base language model.



one autoregressive sample over H steps

- A particle is a partial prefix; a full sample is a complete H-step sequence.
- The paper fixes a horizon H to make the theory clean.
- Inference-time algorithms manipulate samples from this reference distribution.

Reward-tilted target distribution

$$\pi^\star(a_{1:H}) \propto \pi_{\mathrm{ref}}(a_{1:H})\, r^\star(a_{1:H})$$

- Motivation: sample completions that are both model-plausible and high-reward.
- Terminal reward can represent correctness, preference, or another final score.
- With a 0/1 correctness reward, the target is the reference distribution conditioned on success.
- The theory asks whether inference-time algorithms can sample close to this target.

Process reward model (PRM)

True prefix value

$$V^\star(a_{1:h}) = \mathbb{E}_{\pi_{\text{ref}}}[r^\star(a_{1:H}) \mid a_{1:h}]$$

PRM score used by the algorithm

$$\widehat{V}(a_{1:h}) \approx V^\star(a_{1:h})$$

- True value: expected final reward if the base model continues from this prefix.
- PRM estimate: the score actually available to the inference-time algorithm.
- The analysis compares the ideal distribution induced by the true value with the distribution induced by the PRM estimate.

Intermediate target distributions

$$\pi_h^\star(x) \propto \pi_h(x)\, V^\star(x),$$

$$\widehat{\pi}_h(x) \propto \pi_h(x)\, \widehat{V}(x)$$

Top line: ideal prefix distribution.
Bottom line: PRM-induced approximation.

- The theory asks for total variation closeness at h = H.
- The analysis controls error accumulation across the horizon h = 1,…,H.

2. Sequential Monte Carlo (SMC)

Particles as partial generations

- A particle is one partial generation, not a model parameter.
- The particle cloud approximates a distribution over prefixes.
- Resampling reallocates compute toward prefixes with higher estimated reward.

The algorithm keeps a small population. Each step extends every prefix, scores the new prefixes, duplicates promising ones, and discards weak ones.

Algorithm 1: Sequential Monte Carlo

Sequential Monte Carlo

$$\textbf{Input: } \pi_{\text{ref}},\ \widehat{V},\ N$$

$$\widehat{\nu}_0 \leftarrow \delta_{\emptyset}$$

$$\textbf{for } h = 1, \ldots, H:$$

$$\widetilde{x}_{h-1}^{(i)} \sim \widehat{\nu}_{h-1}$$

$$x_h^{(i)} \sim \pi_{\text{ref}}(\cdot \mid \widetilde{x}_{h-1}^{(i)})$$

$$w_h^{(i)} \leftarrow \widehat{V}(x_h^{(i)}) / \widehat{V}(\widetilde{x}_{h-1}^{(i)})$$

$$\widehat{\nu}_h \leftarrow \text{weighted empirical measure over } x_h^{(i)}$$

$$\textbf{output } x_H \sim \widehat{\nu}_H$$

$$w_h = \frac{\widehat{V}(\text{new prefix})}{\widehat{V}(\text{parent prefix})}$$

Weighting intuition: the parent was already selected for being promising. The ratio asks whether the newly sampled action made the prefix more or less promising.

Then resampling turns these weights into the next population of prefixes.

What SMC buys over Best-of-N

Method	When scoring happens	What gets reused
Best-of-N	after full generations	nothing until the end
SMC	at every step	promising prefixes are replicated

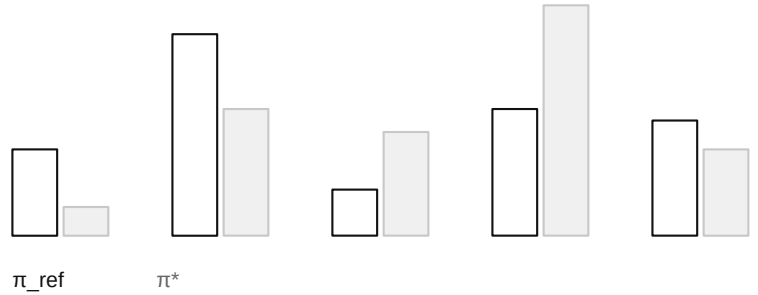
The cost is bias risk: if the PRM is wrong early, resampling can delete useful modes.

3. Guarantees for SMC

Quantity 1: action-level coverage

$$\frac{\pi^\star(a_{h+1} \mid a_{1:h})}{\pi_{\mathrm{ref}}(a_{h+1} \mid a_{1:h})} \le C_{\mathrm{act}}$$

coverage = target mass must not hide where base has tiny probability



- Small coverage constant: target next-actions are visible under the reference distribution.
- Large coverage constant: important continuations are rare under the base model.
- This is a proposal-quality condition.

Quantity 2: PRM-induced distribution error

$$D_{\chi^2}(\pi_h^\star \,\|\, \widehat{\pi}_h) \le C_{\chi^2},$$

$$\widehat{\pi}_h(x) \propto \pi_h(x)\, \widehat{V}(x)$$

- This measures average-case PRM error under the target-like distribution.
- It is weaker than worst-case uniform accuracy of V-hat relative to V-star.
- But it can be sensitive to tail failures.

Main SMC bound

Theorem 3.2

$$D_{\mathrm{TV}}\big(\mathbb{E}[\widehat{\nu}_H],\, \pi_H^\star\big) \le \sqrt{\frac{C_{\mathrm{act}}}{N}} \left(H + \sum_{h=1}^{H-1} \sqrt{D_{\chi^2}(\pi_h^\star \,\|\, \widehat{\pi}_h)} \right)$$

$$D_{\chi^2}(\pi_h^\star \,\|\, \widehat{\pi}_h) \le C_{\chi^2}\ \forall h \quad \Longrightarrow \quad D_{\mathrm{TV}} \le H \sqrt{\frac{C_{\mathrm{act}}(C_{\chi^2}+1)}{N}}$$

- More particles reduce error at roughly a square-root rate.
- Long horizons, poor coverage, and inaccurate PRM scores increase error.

Proof intuition

1. SMC maintains a weighted empirical approximation to the PRM-induced prefix distribution.

2. The product of average weights estimates the partition function Z.

3. Total variation error is bounded by deviation of this estimator.

4. Coverage and PRM divergence bound the variance of the per-step increments.

Heavy tails and alternative coverage

- χ^2 divergence is interpretable but tail-sensitive.
- The paper introduces a coverage-style quantity for the ideal-vs-PRM-induced prefix distributions.
- This yields a guarantee with an additive tail-mass term.
- Useful lesson: distributional guarantees depend heavily on how we measure PRM error.

4. Beyond vanilla SMC

A pathology: even perfect PRMs do not save vanilla SMC

Even with $\hat{V} = V^*$, vanilla SMC may require $\Omega(\sqrt{H})$ particles for nontrivial sampling error.

- This is not just looseness in the analysis.
- The issue comes from how SMC normalizes and resamples finite particle populations.
- It motivates a more local rejection-sampling step.

SMC-RS: Sequential Monte Carlo with Rejection Sampling

SMC-RS

$$S_0 = \{\emptyset, \ldots, \emptyset\} \ (N \text{ copies})$$

for $h = 1, \ldots, H$:

$S_h \leftarrow \emptyset$

while $|S_h| < N$:

$x_{h-1} \sim \mathrm{Unif}(S_{h-1})$

$x_h \sim \pi_{\mathrm{ref}}(\cdot \mid x_{h-1})$

accept x_h with prob. $\widehat{V}(x_h)/(\eta\,\widehat{V}(x_{h-1}))$

if accepted, add x_h to S_h

output a uniform particle from S_H

The accept/reject step tries to sample more directly from the PRM-guided local proposal.

SMC-RS guarantee

$$D_{\mathrm{TV}}\big(\mathbb{E}[\widehat{\nu}_H],\, \pi_H^\star\big) \;\le\; \frac{1}{\sqrt{N}} \sum_{h=1}^{H} \sqrt{D_{\chi^2}(\pi_h^\star \,\|\, \widehat{\pi}_h)}.$$

- Expected time complexity: O(NHη).
- If D_χ² ≤ ε² and ε ≤ 1/H, then O(1) particles can give o(1) sampling error.
- This avoids the vanilla-SMC Ω(√H) pathology in the perfect/near-perfect PRM regime.

Lower bound for myopic particle filters

Any myopic particle-filtering method needs at least about Ω(log H / log log H) particles in the paper’s lower-bound construction.

- Myopic = maintain depth-h particles using only current/past PRM information.
- The lower bound applies under constant-factor PRM imperfections.
- Avoiding it likely requires some form of lookahead or non-myopic search.

5. Experiments

Experiment 1: prompt switching

1. Use one prompt as the reference distribution and another as the target distribution.
2. Define a likelihood-ratio reward so the target prompt distribution is exactly the reward-tilted distribution.
3. Run the inference algorithm and measure how close its output distribution is to the target distribution.
4. Now coverage and PRM error can be varied and measured directly.

$$r^\star(a_{1:H}) = \frac{M(a_{1:H} \mid p^\star)}{M(a_{1:H} \mid p_{\mathrm{ref}})}$$

Purpose: isolate the sampling theory in a setting where the target distribution is known.

Prompt switching: what the theory predicts

Mechanism	What changes	Observed effect
PRM-induced distribution error	the scorer pushes particles toward the wrong prefixes	sampling error increases
action-level coverage	target-useful actions become rare under the base model	sampling error increases
number of particles N	the empirical population becomes larger	sampling error decreases

Takeaway: the theorem is useful for predicting distributional sampling error in controlled settings.

Particle count: a sanity check

Increasing N makes the empirical particle cloud a better approximation to the target distribution.

- N is the number of particles maintained at each depth.
- As N grows, sampling error decreases in the prompt-switching experiments.
- Baselines: Best-of-N and sequential importance sampling (SIS).
- The trend is consistent with a Monte Carlo rate: better with more particles, but with diminishing returns.

Experiment 2: math problem solving

- Benchmarks: Math500, AIME24, AIME25.
- Base model generates partial solution traces.
- The PRM scores prefixes during generation.
- SMC prunes weak prefixes and replicates stronger ones.
- Metric: final answer correctness.

Important shift: the theorem controls distributional sampling error; the benchmark reports answer accuracy.

So the math experiments are evidence that resampling can help problem solving, not a direct validation of total-variation theory.

Math result: accuracy is not sampling error

Sampling objective	Problem-solving objective
match the full target distribution	find one correct final answer
missing modes is a problem	aggressive pruning can help
measured by total variation / divergence	measured by accuracy

Main empirical lesson: a PRM can improve accuracy by concentrating probability mass even when it worsens distributional fidelity.

6. Discussion and takeaways

What is conceptually strong

- It turns many “parallel reasoning” heuristics into a recognizable Monte Carlo algorithm class.
- It gives finite-particle guarantees with interpretable conditions.
- It identifies algorithmic refinements, especially rejection-sampling variants.
- It provides lower bounds showing that myopic filtering has fundamental limits.

Main caveats

- The target is a distributional sampling objective; applications often care about best-answer accuracy.
- The fixed-horizon action abstraction hides variable-length reasoning and prompt effects.
- The PRM is assumed to estimate expected terminal reward, but practical PRMs can be miscalibrated.
- The most interesting math result is partly a mismatch with the theory.

Main takeaway

RRR says: parallel inference-time reasoning is particle filtering; it works when the base model covers useful actions and the PRM points particles toward the right intermediate distributions.

- SMC gives a baseline theory for generate-score-prune methods.
- SMC-RS shows local rejection sampling can fix important pathologies.
- Math accuracy likely needs a theory weaker than total-variation sampling.

Discussion questions

1. What task objectives are better than total variation distance for math reasoning?
2. Can PRMs be trained to optimize coverage of some correct solution rather than distributional fidelity?
3. What forms of lookahead would evade the myopic lower bound while remaining computationally realistic?
4. How should we compare SMC, tree search, and backtracking when evaluator calls dominate cost?

Reference

Golowich, N.; Chen, F.; Rohatgi, D.; Singhal, R.; Domingo-Enrich, C.; Foster, D. J.; Krishnamurthy, A. “Reject, Resample, Repeat: Understanding Parallel Reasoning in Language Model Inference.” arXiv:2603.07887, 2026.

The deck paraphrases definitions, theorem statements, algorithms, and experimental interpretations from the paper.