From Rejection Sampling to NUTS: Theory and Empirical Trade-offs in Bayesian Robot Posterior Inference

Samuel Patel | Guanzhong Yang | Nathan Tan | Kai Samaroo

Imperial College London

Abstract

This poster compares Rejection Sampling, MH, HMC, and NUTS within a single theory-to-evidence pipeline for Bayesian sampling under intractable normalization. Our goal is to compare both sample quality and computational cost using acceptance rate, IACT, ESS, ESS/s, and time per sample. Toy visualizers are used only to illustrate algorithmic behavior, while quantitative conclusions are drawn from a 2D robot posterior surrogate under fixed non-adaptive tuning.

Comparison Goal

- **Methods compared:** Rejection Sampling, MH, HMC, and NUTS.
- **Primary question:** which method gives the best trade-off between mixing quality and wall-clock efficiency?
- **Metrics:** acceptance, IACT, ESS, ESS/s, and time per sample.
- **Scope:** toy examples are illustrative; quantitative ranking is based on the 2D robot posterior surrogate only.

Section I — Bayesian Target and Why Sampling Is Needed

$$\pi(\theta \mid x) = \frac{\pi(\theta) f(x \mid \theta)}{Z(x)}, \quad Z(x) = \int_\Theta \pi(\theta) f(x \mid \theta)\, d\theta.$$

$$\log \tilde{\pi}(\theta \mid \mathcal{D}) = \log \pi(\theta) + \frac{R(\theta)}{5{,}000{,}000}, \quad \theta = [k_{px}, k_{py}]^\top, \quad \pi(\theta) = \mathcal{N}(0, 1.2 I_2), \quad R(\theta) = \sum_{t=0}^{499} \gamma^t r_t(\theta),\ \gamma = 0.99,\ dt = 0.01.$$

- **Narrative Role:** Define both the generic Bayesian target and the robot-case surrogate before selecting algorithms.
- **Strict Mathematical Premise:** Generic case: $\pi(\theta) \ge 0$, $\int_\Theta \pi(\theta) d\theta = 1$, $0 < Z(x) < \infty$; toy visualizer uses a 2D three-component Gaussian mixture; robot case uses reward return $R(\theta)$ as a likelihood proxy.
- **Flow:** Use unnormalized targets for acceptance-ratio methods and estimate expectations from post-burn-in samples.
- **Advantages:** Gives a rigorous uncertainty-aware objective.
- **Limitations:** Robot target is a reward-based posterior surrogate (not a fully specified noise likelihood).

With the posterior target fixed and the normalizing constant unavailable in practice, we introduce rejection sampling as an exact i.i.d. benchmark whose acceptance is controlled by envelope quality M.

Section II — Exact Baseline: Rejection Sampling

$$\alpha(y) = \frac{f(y)}{M g(y)}, \qquad f(y) \le M g(y)\ \forall y.$$

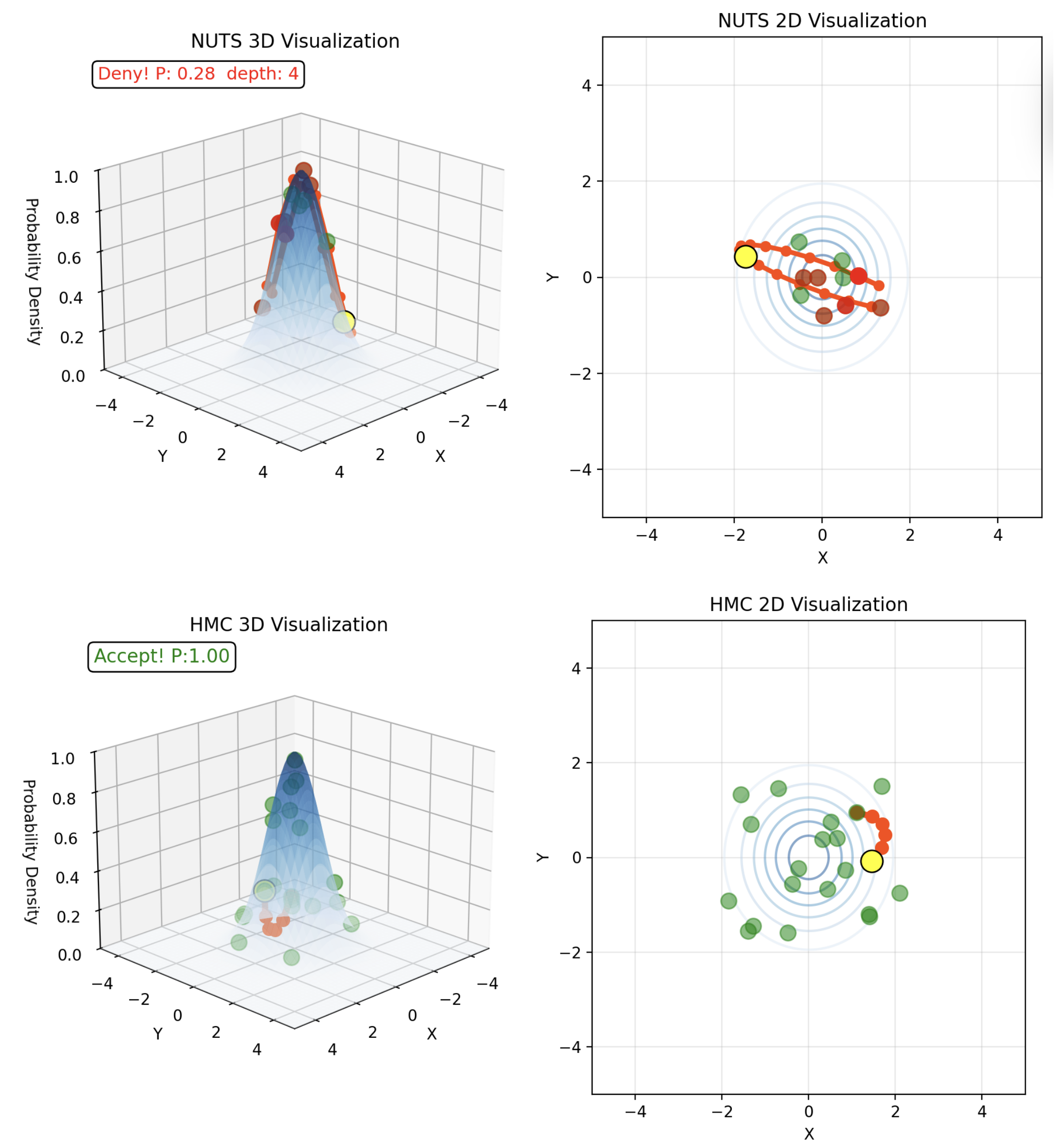

Fig. 1: NUTS single-modal normal visualizer   Fig. 2: HMC single-modal normal visualizer

- **Narrative Role:** Set a correctness baseline before correlated samplers.
- **Strict Mathematical Premise:** $\operatorname{supp}(f) \subseteq \operatorname{supp}(g)$ and finite envelope $M \ge \sup_y \frac{f(y)}{g(y)} < \infty$.
- **Flow:** Draw $y \sim g$, accept by $\alpha(y)$, repeat until accepted.
- **Advantages:** i.i.d. accepted samples exactly follow target.
- **Limitations:** In high dimensions, envelope design is often inefficient.

- **What to check:** How acceptance changes with posterior geometry.
- **What we observed:** In multimodal geometry, envelope inflation lowers acceptance and motivates MCMC.

When high-dimensional or multimodal geometry makes tight envelopes difficult and acceptance drops, we shift to Markov kernels that preserve π and recover correctness through stationarity and ergodicity.

Section III — MCMC Principle: Stationarity over Independence

$$\pi(x) T(y \mid x) = \pi(y) T(x \mid y)$$

- **Narrative Role:** Replace exact i.i.d. sampling with asymptotically correct Markov-chain sampling.
- **Strict Mathematical Premise:** T is π-irreducible, aperiodic, Harris recurrent; diagnostics are computed after burn-in under common retained length and wall-clock timing conventions.
- **Flow:** Build transition kernel, run chain, then estimate ACT/IACT with $\widehat{\mathrm{IACT}} = 1 + 2\sum \hat{\rho}_k$, $\widehat{\mathrm{ESS}} = N/\widehat{\mathrm{IACT}}$, and $\mathrm{ESS/s} = \widehat{\mathrm{ESS}}/t_{\mathrm{wall}}$.
- **Advantages:** Scales where direct methods fail.
- **Limitations:** Poor mixing yields high autocorrelation and low ESS; fairness still depends on comparable protocol choices.

After invariance is formalized by detailed balance, Metropolis–Hastings gives a constructive recipe: propose from $q(\cdot \mid x)$ and correct by an acceptance ratio without requiring $Z(x)$.

Section IV — MH: General-Purpose Corrector

$$\alpha(x, y) = \min\left\{1, \frac{\tilde{\pi}(y) q(x \mid y)}{\tilde{\pi}(x) q(y \mid x)}\right\}.$$

- **Narrative Role:** Turn arbitrary proposals into valid posterior samplers.
- **Strict Mathematical Premise:** Reverse proposal terms are defined where forward moves have nonzero mass; here $q(\cdot \mid x) = \mathcal{N}(x, \sigma^2 I_2)$ with fixed $\sigma = 0.15$.
- **Flow:** Propose $y \sim q(\cdot \mid x)$, then accept/reject by $\alpha(x, y)$ with non-adaptive tuning.
- **Advantages:** Simple and broadly applicable.
- **Limitations:** Random-walk behavior can be severe in correlated geometry.

- **What to check:** Acceptance, IACT, ESS, ESS/s, and wall-clock cost under the shared protocol.
- **What we observed:** MH (robot posterior, seed 42): accept 0.907, IACT $(39.1, 75.5)$, ESS $(7.7, 4.0)$, ESS/s $(1.70, 0.88)$, 15.3 ms/sample.

Because robot-posterior diagnostics under MH show strong autocorrelation and random-walk behavior, we move to Hamiltonian dynamics to exploit gradients and generate longer effective moves.

Section V — HMC: Geometry-Aware Proposals

$$H(x, m) = U(x) + K(m), \quad U(x) = -\log \tilde{\pi}(x), \quad K(m) = \tfrac{1}{2} m^\top M^{-1} m.$$

- **Narrative Role:** Reduce random-walk inefficiency with Hamiltonian dynamics.
- **Strict Mathematical Premise:** $U \in C^1$ with locally Lipschitz ∇U; implementation uses fixed $(\varepsilon, L) = (0.2, 20)$ and $M = I_2$ (no dual averaging).
- **Flow:** Sample momentum, integrate L leapfrog steps (trajectory length $\varepsilon L = 4.0$), then Metropolis correction.
- **Advantages:** Longer effective moves with lower autocorrelation.
- **Limitations:** Higher per-step compute and tuning burden.

- **What to check:** Whether mixing gain justifies compute cost.
- **What we observed:** HMC (robot posterior, seed 42): accept 0.973, IACT $(4.9, 8.2)$, ESS $(61.9, 36.6)$, ESS/s $(0.66, 0.39)$, 313.5 ms/sample; about $8\times$ IACT gain vs MH but lower throughput in this 2D case.

Although HMC sharply improves mixing, it still requires manual path-length tuning (L) and step-size coupling; NUTS targets this burden via U-turn stopping, but our current implementation is fixed-step rather than fully adaptive.

Section VI — NUTS: Adaptive Trajectory Length

- **Narrative Role:** Reduce HMC path-length hand tuning while preserving invariance.
- **Strict Mathematical Premise:** Leapfrog is reversible and volume-preserving; NUTS uses slice variable $u \sim \mathrm{Unif}(0, \exp(-H(x, m)))$, symmetric tree selection, and U-turn stopping.
- **Flow:** Current run uses fixed $\varepsilon = 0.3$, $\texttt{max\_depth} = 5$, unit mass matrix, and recursive tree expansion.
- **Advantages:** Robust across heterogeneous posterior geometries.
- **Limitations:** This is a simplified fixed-step NUTS (no dual averaging), so quantitative ranking is deferred.

- **What to check:** Implementation fidelity versus standard NUTS and benchmark readiness.
- **What we observed:** Slice variable, recursive tree building, and U-turn checks are implemented; step-size adaptation is absent (fixed $\varepsilon = 0.3$, $\texttt{max\_depth} = 5$, $M = I$), so NUTS evidence here remains qualitative.

With this scope explicit, we bridge to the robot case: data enter through deterministic discounted return $R(\theta)$, producing a 2D posterior surrogate whose geometry directly controls MH random-walk behavior and HMC/NUTS trajectory efficiency.

Section VII — Theory-to-Evidence Closure (Robot Posterior)

- **Narrative Role:** State comparison scope and fairness assumptions explicitly.
- **Strict Mathematical Premise:** 2D target, seed (42), same burn-in/retained lengths, non-adaptive tuning, and wall-clock timing including all leapfrog work.
- **Flow:** For $\theta = [k_{px}, k_{py}]^\top$, report $\widehat{\mathrm{IACT}}$, $\widehat{\mathrm{ESS}} = N/\widehat{\mathrm{IACT}}$, and ESS/s.
- **Advantages:** Makes empirical trade-offs auditable and methodologically explicit.
- **Limitations:** Results are local to this 2D reward-based surrogate with numerical gradients.

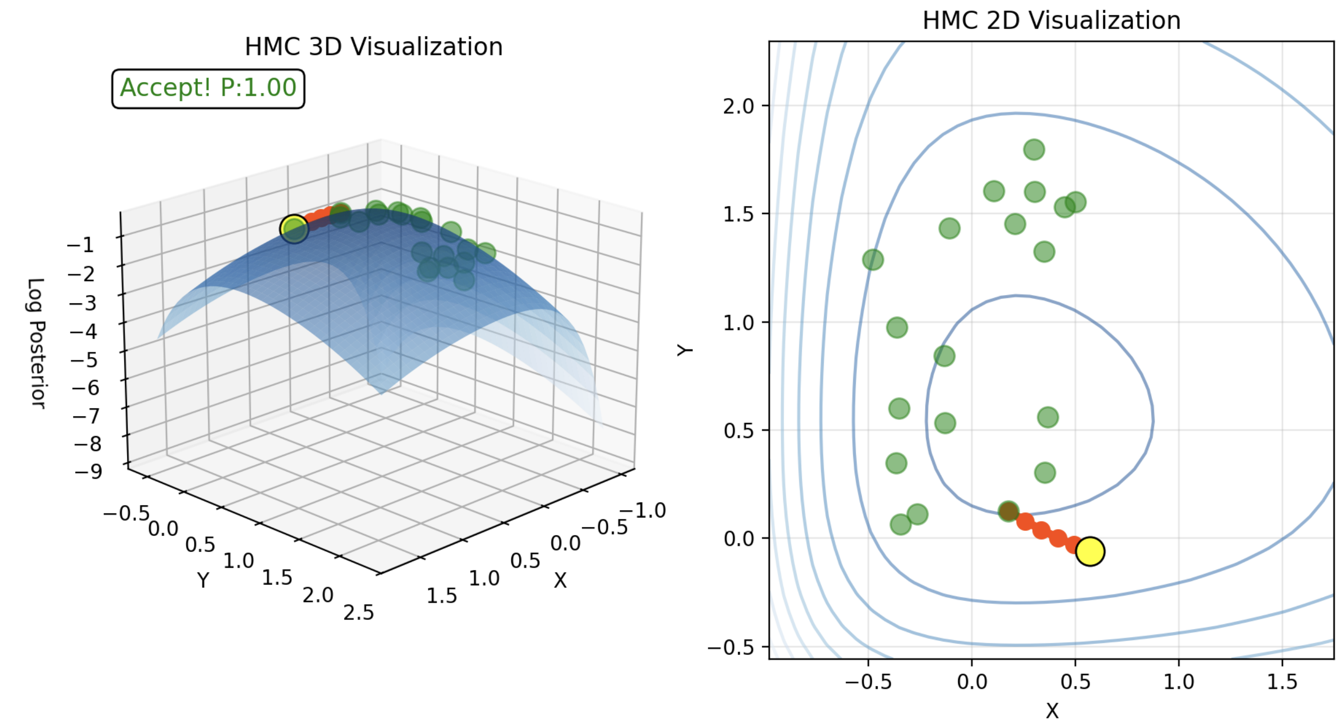

Fig. 3: Robot simulation visualizer

k_{px}, k_{py}: proportional gains inferred in the robot posterior surrogate.

Metric	MH	HMC
Acceptance rate	0.907	0.973
IACT (k_{px}/k_{py})	39.1 / 75.5	4.9 / 8.2
ESS (k_{px}/k_{py})	7.7 / 4.0	61.9 / 36.6
ESS/s (k_{px}/k_{py})	1.70 / 0.88	0.66 / 0.39
Time per sample	15.3 ms	313.5 ms

Section VIII — Conclusion

This poster compares sampler performance under explicit scope: 2D target, seed 42, fixed non-adaptive tuning, and wall-clock accounting that includes all leapfrog work.

Claim 1 (mixing): HMC mixes much faster than MH on the robot posterior surrogate, with much lower IACT ($4.9/8.2$ vs $39.1/75.5$) and correspondingly higher ESS per draw.

Claim 2 (wall-clock efficiency): despite poor mixing, MH achieves the highest ESS/s ($1.70/0.88$ vs $0.66/0.39$) because its per-sample cost is much lower (15.3 ms vs 313.5 ms).

Claim 3 (anisotropy): both samplers mix more slowly in the k_{py} direction than in k_{px}, indicating residual posterior anisotropy that a unit mass matrix does not fully remove.

Claim 4 (scope and recommendation): in this low-dimensional setting, well-tuned MH remains a strong baseline, while geometry-aware methods improve sample quality; current fixed-step NUTS is qualitative only, and quantitative ranking requires adaptive benchmarking.

References

- Vahid Tavakol Aghaei, Arda Agababaoğlu, Sinan Yildirim, and Ahmet Onat, "A real-world application of Markov chain Monte Carlo method for Bayesian trajectory control of a robotic manipulator," *ISA Transactions*, 125:580–590, 2022.
- Radford M. Neal, "MCMC using Hamiltonian dynamics," in *Handbook of Markov Chain Monte Carlo*, 2011, pp. 113–162.
- Matthew D. Hoffman and Andrew Gelman, "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo," *Journal of Machine Learning Research*, 15(47):1593–1623, 2014.