HIGH-ACCURACY PARALLEL DIFFUSION SAMPLING: NEAR-LINEAR QUERIES AND SQUARE-ROOT DEPTH

Anonymous authors
Paper under double-blind review

ABSTRACT

Standard diffusion samplers are sequential: every denoiser call depends on the previous state. Parallel proposals reduce this dependency, but at high accuracy they can require far more denoiser calls. We give a parallel sampler with near-linear query complexity under only a finite-second-moment assumption. For any zero-mean distribution π on $\mathbb{R}^n$ with a known second-moment bound and exact access to its Gaussian posterior mean, the sampler uses $\widetilde{O}(n)$ expected full-vector queries and $\widetilde{O}(\sqrt{n})$ expected adaptive rounds. We use Euler discretization only for parallel proposals and correct them by rejection sampling against the high-accuracy FORS diffusion path of Chen et al. (2026). We introduce a mechanism we call a *causal guard* to handle rare, large changes in the diffusion drift. Compared with the recursive parallel result of Anari et al. (2026), this removes bounded support and replaces polynomial dependence on accuracy, smoothing, and support radius with polylogarithmic dependence.

1 INTRODUCTION

Diffusion models are useful across high-dimensional generation, including image synthesis and visuomotor control (Ho et al., 2020; Song et al., 2021b; Rombach et al., 2022; Chi et al., 2023). Their reverse computation is nevertheless temporal: the state at one noise level determines the score or denoiser query at the next. This makes neural evaluations a latency bottleneck even when many processors are available. Existing acceleration methods either shorten the trajectory (Song et al., 2021a; Lu et al., 2022) or keep several trajectory locations in flight (Shih et al., 2023). In an oracle model these optimize different resources—the total number of queries and their adaptive depth—and a method can improve one while worsening the other.

We study this tradeoff under minimal assumptions. Our theorem controls an early-stopped diffusion in total variation. Let $x_\star \sim \pi$ be a zero-mean random vector in $\mathbb{R}^n$ with a known bound $\mathbb{E}\|x_\star\|^2 \le M_2^2 < \infty$. For $\delta > 0$, the target is

$$\pi_\delta := \pi * \mathcal{N}(0, \delta I_n)\,.$$

The sharp total-variation statement keeps δ separate from its algorithmic tolerance: under a finite-second-moment assumption alone, an atomic π remains total-variation distance one from every nondegenerate Gaussian convolution. For comparison with π, choosing $\delta = \eta^2/(4n)$ and total-variation tolerance $\eta/4$ gives bounded-Lipschitz error at most η from π, by Gaussian coupling and the standard comparison between bounded-Lipschitz and total-variation distance.

Here I_n is the $n \times n$ identity matrix. We write $\mathrm{Law}(X)$ for the law of a random variable and $d_{\mathrm{TV}}(P, Q)$ for total-variation distance.

The algorithm may query the exact Gaussian posterior mean

$$f(t, y) := \mathbb{E}[x_\star \mid t x_\star + \sqrt{t}\, G = y], \qquad t > 0, \quad G \sim \mathcal{N}(0, I_n)\,,$$

where G is independent of $x_\star$. This is equivalent, by Tweedie's identity, to exact score access for every Gaussian smoothing. One evaluation of the full vector $f(t, y)$ is one query; any finite batch chosen

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.

without seeing answers from that batch is one adaptive round. The appendix fixes the pointwise version of f, states the coordinate change to the score, and reviews the stochastic-localization representation used in the proof.

We call $f(t, y)$ the localization drift at time t and state y; thus a frozen drift is simply a posterior-mean value held fixed across speculative steps.

The natural Euler step freezes f and adds a Gaussian increment. It is computationally attractive: after one query at a block anchor, all proposed increments can be drawn independently and all proposed states recovered by a parallel prefix scan. In the predecessor's Euler-target analysis (Anari et al., 2026), controlling its discretization bias incurs polynomial inverse-accuracy work. We instead keep Euler only as the proposal and make a first-order rejection-sampling (FORS) transition (Chen et al., 2026) the accuracy-determining target. FORS has polylogarithmic accuracy dependence, but its original diffusion sampler is sequential. The challenge is to fit this corrected target into recursive autospeculation.

We call the finite record of a proposed path segment's candidate and auxiliary randomness a *transcript*; the *scheduler* is the recursive rejection-correction routine that accepts such proposals or subdivides the segment after rejection.

The complementary mechanism comes from Anari et al. (2026), who show how a block proposal and a levelwise total-variation bound drive a recursive exact-correction scheduler. Their diffusion result already attains $\widetilde{O}(\sqrt{n})$ expected adaptive depth, but its stated expected-query bound depends polynomially on accuracy, smoothing, and the radius of a covering ball. We preserve the depth scale, obtain $\widetilde{O}(n)$ expected queries, and replace bounded support by a finite second-moment assumption. Up to logarithmic factors, our result preserves the predecessor's linear-query and square-root-depth dimension exponents while improving its dependence on accuracy, smoothing, and the target class.

The three ingredients. We show how to parallelize the sampler on one *coarse interval*, meaning a contiguous segment of localization time subdivided into equal fine steps, and reuse the argument on the polylogarithmically many disjoint intervals that cover the full time horizon. Adjacent intervals may use different fine-step sizes; the exact grid is postponed to Section I. The interval proof uses exactly three new ingredients.

I. A parallel speculative FORS trajectory. We first draw a frozen-drift Euler trajectory in parallel and recover all states by a prefix scan. We then augment those increments with independent proposal-side FORS variables, all recorded in the transcript and used to evaluate normalized target and proposal likelihoods. The guard and transcript scorer together use $O(1)$ batched oracle stages.

II. A causal large-drift guard. Localization controls drift variation under the true path, whereas the scheduler follows a corrected path, on which the drift can fluctuate arbitrarily across scales within a block. Therefore, small pathwise KL alone cannot transfer the required expectation. Before admitting a drift, the guard checks prefix-measurable budgets. On the first prospective violation it discards that drift, freezes the last admitted value, and neutralizes the current and all later correction factors. The algorithm continues rather than halting. The budgets then hold deterministically, while a rare-trigger coupling keeps the guarded target close to the intended one.

III. A levelwise total-variation certificate. The guard controls Gaussian mean shifts and the FORS estimate controls the remaining correction energy. Together they bound, at every level of the ordered binary fallback tree introduced in Section 2.1, the expected sum of conditional proposal–target distances by $O(\sqrt{n})$. This is a runtime certificate, not approximation error: the recursive scheduler corrects the proposal exactly.

The first and third items are precisely the local-depth and rejection-budget interfaces used by the scheduler of Anari et al. (2026); the guard makes the third valid for the path actually sampled. Section 2 gives the algorithm, the three-law proof diagram, and a top-down proof of the main theorem. Section 3 then places the result within diffusion acceleration and parallel-in-time sampling. All transition densities, guard updates, recursive routines, parameter checks, and complete proofs appear after the references.

2 ALGORITHM AND PROOF OF THE MAIN THEOREM

Theorem 2.1 (Main theorem). *There is a universal constant $\varepsilon_0 > 0$ such that the following holds for every $n \geq 1$, $\delta > 0$, and accuracy parameter $0 < \varepsilon \leq \varepsilon_0$. Let π be a zero-mean Borel probability measure on $\mathbb{R}^n$, and suppose a finite bound $M_2^2 \geq \mathbb{E}\|x_\star\|^2$ is known for $x_\star \sim \pi$. With exact access to $f(t, y)$,* HIGH-ACCURACY AUTOSPEC *returns X_{out} satisfying*

$$d_{\mathrm{TV}}(\mathrm{Law}(X_{\mathrm{out}}), \pi * \mathcal{N}(0, \delta I_n)) \leq \varepsilon.$$

If Q *is the number of full-vector oracle evaluations and* D *is the number of adaptive oracle rounds, then the same execution satisfies*

$$\mathbb{E}[\mathsf{Q}] = \widetilde{O}(n), \qquad \mathbb{E}[\mathsf{D}] = \widetilde{O}(\sqrt{n}).$$

Here $\widetilde{O}$ hides factors polylogarithmic in $n+1$, ε^{-1}, and $1+M_2^2/\delta$. No bounded-support, Lipschitz, or score-regularity assumption is made. The result is in the exact-real oracle model: random primitives and likelihood tests are exact, and an arbitrary finite batch of nonadaptive oracle calls counts as one round.

The explicit logarithmic exponents and all parameter choices are given in Section I. We keep those bookkeeping symbols out of the main argument. Throughout, the FORS score-error parameter is zero and its intrinsic-dimension parameter is bounded by the ambient data dimension n; we make no lower-intrinsic-dimensional claim.

2.1 FROM SEQUENTIAL FORS TO THE PARALLEL ALGORITHM

The sequential high-accuracy law. The proof compares three path laws: the ideal law $\mathbf{P}$, the unguarded FORS law $\mathbf{Q}_0$, and the guarded law $\mathbf{Q}_g$ used by the parallel sampler. The ideal process is $Y_t = tx_\star + B_t$, where $(B_t)_{t\geq 0}$ is standard Brownian motion independent of $x_\star$. The first law, $\mathbf{P}$, is the joint law of $(Y_{t_i})_{i=0}^M$ on the algorithm's discrete time grid specified in Section I. Starting from the Gaussian initialization and applying the stopped, capped FORS transition at every fine step gives the unguarded path law $\mathbf{Q}_0$. Given the current state, one can sample the next $\mathbf{Q}_0$ transition exactly as defined; proceeding one step at a time therefore samples $\mathbf{Q}_0$ exactly. Combining the FORS estimate adapted from Chen et al. (2026) with initialization, cap control, and the KL chain rule makes $\mathbf{Q}_0$ a high-accuracy approximation to $\mathbf{P}$. This sampler is nevertheless sequential: the oracle queries for one transition depend on the state selected by the preceding transition.

Euler as a parallel proposal. A frozen-drift Euler block has the opposite strengths. Once its entering state and anchor drift are fixed, all Gaussian increments can be drawn at once and all proposed states recovered by a parallel prefix scan. It is thus a natural parallel proposal. It is not our accuracy-determining target: in the Euler-target analysis of Anari et al. (2026), the stated high-accuracy guarantee uses polynomially many fine steps in the inverse error, and the complete bounded-support guarantee also retains polynomial dependence on smoothing and the covering radius. We instead keep the FORS correction as the target and use frozen Euler geometry only to speculate.

From $\mathbf{Q}_0$ to a guarded target. To parallelize, we use the recursive autospeculation framework of Anari et al. (2026). For each proposed time block, the scheduler needs a proposal that can be generated in constant oracle depth and an evaluable proposal–target likelihood ratio; work-efficient recursion additionally requires a levelwise bound on their conditional total-variation distances. Freezing one drift supplies the parallel proposal, while an augmented FORS transcript exposes its normalized likelihood. The remaining obstruction is the levelwise comparison: on atypical $\mathbf{Q}_0$ histories, the drift can fluctuate arbitrarily across scales within a block. Small pathwise KL between $\mathbf{P}$ and $\mathbf{Q}_0$ does not transfer the expectation of the resulting variation functional.

We therefore introduce the guarded path law $\mathbf{Q}_g$. Before admitting a new drift, the guard checks prefix-measurable variation budgets. At the first prospective violation, it discards that drift, freezes the last admitted value thereafter, and neutralizes the current and later FORS correction factors; it does not halt the sampler. The budgets now hold deterministically. Under a common-randomness

coupling, $\mathbf{Q}_0$ and $\mathbf{Q}_g$ differ only if this rare trigger occurs, so the guard costs little total variation. The frozen proposal is then compared with conditional $\mathbf{Q}_g$ blocks, and recursive autospeculation removes that proposal discrepancy exactly.

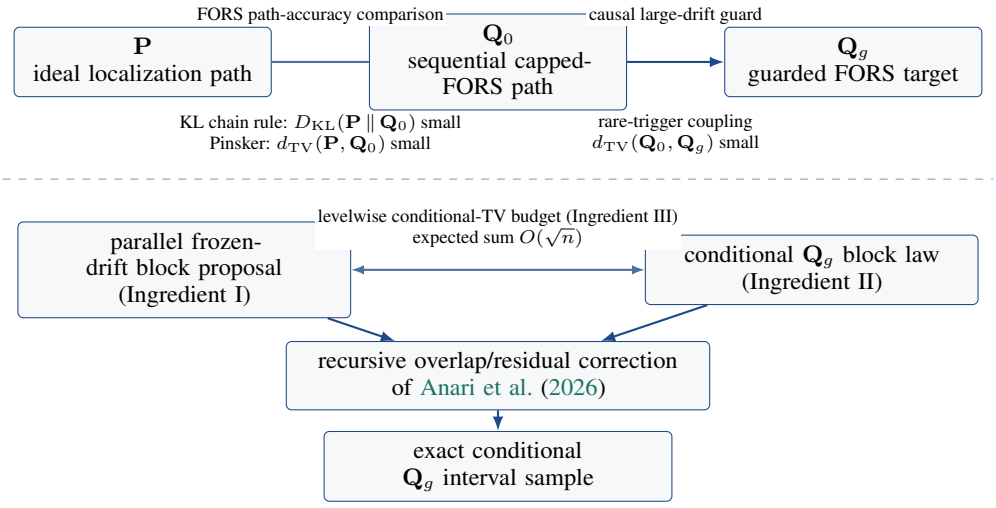

Figure 1: From sequential accuracy to parallel execution. The top row contains the two comparisons that contribute output error. The bottom row controls rejections only: recursive autospeculation removes the proposal discrepancy exactly.

The two rows of Figure 1 have different roles. The top row is the complete accuracy comparison, $\mathbf{P}$ versus $\mathbf{Q}_0$ versus $\mathbf{Q}_g$. The bottom row is computational: the levelwise proposal–target distances control the scheduler's runtime, but no proposal error accumulates in its output.

We next make the recursion explicit for one coarse interval. Suppose the interval contains $N = 2^h$ consecutive single-step transitions. Regard them as the leaves of an ordered binary fallback tree. The root is the whole interval; each internal node is a contiguous block B whose children B_{L} and B_{R} are its consecutive left and right halves. A leaf is one single-step transition. This tree is a fallback template, not a schedule that runs an entire level at once: as Figure 2 shows, the sampler opens the children of a node only if its whole-block proposal is rejected.

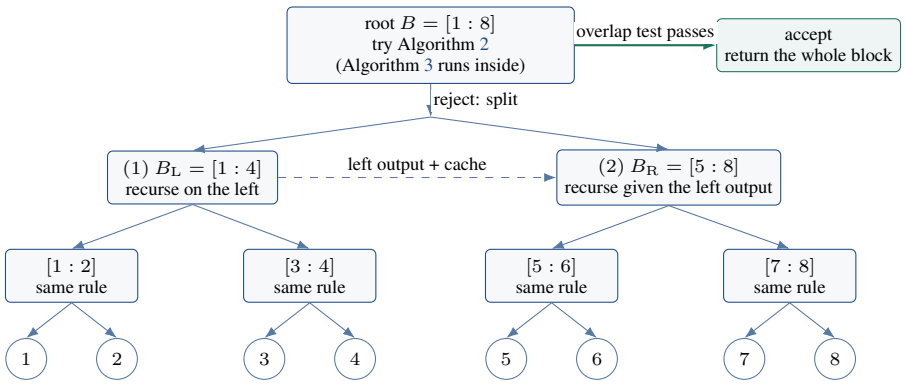

Figure 2: Binary fallback recursion on one coarse interval ($N = 8$ shown). At every internal node, Algorithm 2 first proposes the entire block, using Algorithm 3 to generate all Euler states in parallel. Acceptance returns the block without visiting its descendants. After rejection, the sampler recursively constructs the left child and then the right child conditional on the left, concatenates them, and applies the parent's residual test. Independent complete fallback candidates are batched in parallel; the two children within one candidate are ordered.

The overlap test accepts the proposal–target overlap; after a rejection, the residual test recovers the remaining target mass. The following informal routine states only this control flow. The exact batch schedule and density formulas are deferred to Section H.

Algorithm 1: Recursive guarded-FORS sampler on one coarse interval (informal; see Algorithm 6 for the detailed version).

Input: A block B, its incoming history, and its guard cache (state, frozen flag, last drift, and partial budgets)
Output: An exact conditional $\mathbf{Q}_g$ transcript on B and its outgoing guard cache
if $|B| = 1$ **then**
  sample the single guarded FORS transition exactly and **return** it with its updated cache;
Call Algorithm 2 with B and the entering state/cache carried by e to obtain a whole-block proposal X, its overlap score, and its provisional outgoing cache; this call invokes Algorithm 3;
if *the overlap test accepts* X **then**
  return X and its provisional cache;
Discard the rejected proposal's cache and split $B = B_{\mathrm{L}} \dot{\cup} B_{\mathrm{R}}$ into consecutive halves;
while *no fallback candidate has been accepted* **do**
  launch the next geometrically growing batch of independent candidates;
  foreach *candidate index* k *in the batch* in parallel **do**
    recursively sample $(\mathsf{Z}_{\mathrm{L}}^{(k)}, c_{\mathrm{L}}^{(k)})$ on B_{L} from entry e;
    recursively sample $(\mathsf{Z}_{\mathrm{R}}^{(k)}, c_{\mathrm{R}}^{(k)})$ on B_{R} from the entry obtained by appending the returned $\mathsf{Z}_{\mathrm{L}}^{(k)}$ and carrying its outgoing cache $c_{\mathrm{L}}^{(k)}$;
    concatenate the children and evaluate the parent's residual test using the same guarded-target and proposal block densities;
  after the batch finishes, if any candidate passes, **return** the first one in fixed index order with its outgoing cache;

The complete sampler draws the Gaussian initialization once, processes the polylogarithmically many coarse intervals chronologically, and calls Algorithm 1 at each root. It commits only the returned transcript and cache, then rescales the final localization state to variance δ. Thus the outer loop is chronological, while every accepted whole-block proposal skips all recursion below that node.

The block interface keeps the guard and FORS bookkeeping outside the Euler primitive. Its appendix implementation also specifies the stopped transcript and the normalized densities used by the scheduler.

Algorithm 2: Guarded-FORS block proposal (informal; see Algorithms 4 and 5 for the detailed version).

Input: A non-leaf block B with common step variance a, its entering state x_s, and incoming guard cache
Use the guard at x_s to obtain a frozen anchor drift b_s;
Call Algorithm 3 with (B, x_s, b_s, a) to generate all speculative states in parallel;
Augment this Euler path with the auxiliary FORS transcript, complete the guard pass, and evaluate the guarded-target/proposal likelihood ratios;
return the transcript, speculative states, the target/proposal and proposal/target density ratios, and the provisional outgoing guard cache;

The parallel Euler primitive. Fix m consecutive transitions $B = \{s, s+1, \ldots, s+m-1\}$ with common variance a. Once the entering state and frozen anchor are supplied, trajectory generation contains no oracle call, guard operation, or FORS auxiliary variable.

Why the whole trajectory is parallel. All m Gaussian increments are drawn concurrently, and one prefix scan recovers the states. The scan has $O(\log m)$ bounded-fan-in arithmetic depth and no oracle depth. This Euler path is only a proposal. In Algorithm 4, it is augmented after generation with proposal-side selector and FORS variables sampled independently of the Euler increments, then scored and corrected exactly to the guarded block law by recursive autospeculation.

Algorithm 3: PARALLELEULERBLOCK: frozen-drift Euler proposal.

Input: $B = \{s, \dots, s+m-1\}$, entering state x_s, frozen drift b_s, and step variance a
foreach $i \in B$ in parallel **do**
  Draw $G_i \sim \mathcal{N}(0, I_n)$ and set $\Delta_i \leftarrow a b_s + \sqrt{a}\, G_i$;
Compute all exclusive prefix sums of $(\Delta_s, \dots, \Delta_{s+m-1})$ in parallel and set

$$x^{\mathrm{sp}}_{s+q} \leftarrow x_s + \sum_{p=0}^{q-1} \Delta_{s+p}, \qquad q = 0, 1, \dots, m.$$

return $(\Delta_i)_{i \in B}$ and $(x^{\mathrm{sp}}_{s+q})_{q=0}^{m}$;

2.2 WHY THE ALGORITHM WORKS

The three comparisons in Figure 1 are formalized after the references in Section B. We summarize what each proves.

FORS path accuracy. The corrected transition and its one-step error bound follow from Chen et al. (2026, Proposition F.3) after the coordinate change in the proof of Theorem D.4. Together with Gaussian initialization, finite-cap control, and the KL chain rule, this bounds $D_{\mathrm{KL}}(\mathbf{P} \,\|\, \mathbf{Q}_0)$ by the chosen internal path-error budget. Pinsker's inequality supplies the $\mathbf{P}$-versus-$\mathbf{Q}_0$ total-variation bound.

Guarded transfer. The drift-variation estimate needed for rejection holds under $\mathbf{P}$, but under $\mathbf{Q}_0$ the drift can fluctuate arbitrarily across scales within a block. The guard checks the corresponding hierarchy of drift-variation budgets before admitting each new mean. Hence every budget in the hierarchy holds deterministically on $\mathbf{Q}_g$. If a threshold would be exceeded, the run does not halt and no error is accumulated step by step: the process freezes once, and a common-randomness coupling charges the entire modification to the event that this first trigger occurs. Martingale variation bounds make that event rare under $\mathbf{P}$; the $\mathbf{P}$-versus-$\mathbf{Q}_0$ total-variation bound transfers the estimate to $\mathbf{Q}_0$. This proves that $d_{\mathrm{TV}}(\mathbf{Q}_0, \mathbf{Q}_g)$ is small.

Levelwise rejection control. Condition on the guarded history entering a block. The guard bounds the Gaussian mean-shift energy of its frozen proposal, while a separate FORS correction-energy estimate bounds the auxiliary factors. Summing Hellinger/KL comparisons over disjoint nodes at any fixed tree level gives an expected total conditional TV of $O(\sqrt{n})$. The expectation is over the actual entering law; no pointwise claim is needed for exceptional histories. This quantity can be much larger than ε because it controls runtime, not output accuracy.

Theorem 2.2 (One-interval composition, informal interface)**.** *Fix any one coarse interval and condition on any cache-consistent interval entry. The recursive sampler using Algorithm 2 returns an exact sample from the conditional $\mathbf{Q}_g$ interval law. Its expected query cost is uniform over cache-consistent entries and is nearly linear in n; averaged over the actual entering law, its expected adaptive depth is $\widetilde{O}(\sqrt{n})$. The hidden factors are polylogarithmic in the global problem parameters.*

The precise statement is Theorem B.6. Ingredient I proves the constant local oracle-depth interface. Ingredient II makes the variation budget valid for the implemented target. Ingredient III supplies the levelwise rejection budget. The recursive scheduler theorem of Anari et al. (2026), restated and proved in Sections H and J, then gives exactness and the two cost bounds. This proves Theorem 2.2.

2.3 PROOF OF THE MAIN THEOREM

Proof. Choose the deterministic grid and internal error budget from Definition I.1. The initialization lemma and the corrected FORS path estimate make $d_{\mathrm{TV}}(\mathbf{P}, \mathbf{Q}_0)$ smaller than its allocated part of ε. The guard-transfer proposition makes $d_{\mathrm{TV}}(\mathbf{Q}_0, \mathbf{Q}_g)$ smaller than the remaining part. By the triangle inequality and data processing, the terminal marginal of $\mathbf{Q}_g$ is therefore within ε of the terminal marginal of $\mathbf{P}$.

It remains to identify the law produced by the implementation. Apply the precise one-interval theorem to the first interval. Its output is an exact conditional $\mathbf{Q}_g$ transcript, together with the guard state required at the next boundary. Apply the same theorem successively to the remaining disjoint intervals. Induction gives the full $\mathbf{Q}_g$ path law. In particular, no proposal TV error and no interval-wise approximation error accumulate: all frozen proposals are corrected exactly inside the scheduler.

At terminal precision $T = \delta^{-1}$, the stochastic-localization representation says that rescaling the terminal marginal of $\mathbf{P}$ by T^{-1} gives $\pi * \mathcal{N}(0, \delta I_n)$. Rescaling preserves total variation, which proves the accuracy claim. There are only polylogarithmically many coarse intervals. Summing their query counts and adaptive depths and applying the parameter bounds in Lemma I.2 gives $\mathbb{E}[\mathsf{Q}] = \widetilde{O}(n)$ and $\mathbb{E}[\mathsf{D}] = \widetilde{O}(\sqrt{n})$ for the same execution. The direct-Gaussian edge case and the explicit logarithmic exponents are checked in Section I. □

3 Related work

Diffusion models and faster sampling. Modern diffusion models connect denoising score matching, reverse Markov chains, and continuous-time reverse SDEs (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b). Their usefulness in image generation and control has created substantial demand for faster inference (Rombach et al., 2022; Chi et al., 2023). One response is to reduce the number of sequential model evaluations: DDIM introduces non-Markovian and deterministic samplers (Song et al., 2021a), specialized solvers use higher-order or semianalytic integration (Lu et al., 2022), and distillation or consistency training changes the model to permit very short trajectories (Salimans & Ho, 2022; Song et al., 2023). These methods are highly effective, but their empirical step count is a different objective from exact simulation or worst-case adaptive oracle depth.

Parallelization across diffusion time. A complementary thread evaluates several prospective time steps concurrently. ParaDiGMS uses parallel Picard refinement, ParaTAA accelerates a triangular fixed-point system, and Parareal-based samplers refine several time slices at once (Shih et al., 2023; Tang et al., 2024; Selvam et al., 2024). Recent empirical systems pursue the same latency goal using a learned Picard fixed-point predictor or draft-and-refine skip transitions (So et al., 2025; Bai et al., 2026); they target hardware latency and empirical output quality rather than worst-case oracle depth or total variation to the smoothed data law. Theoretical blockwise Picard and randomized-midpoint schemes give parallel guarantees under score-accuracy and regularity assumptions (Chen et al., 2024; Gupta et al., 2025; Zhou & Sugiyama, 2025). Speculative diffusion methods instead draft a trajectory and verify it against a target chain (Bortoli et al., 2025; Hu et al., 2025; Soen et al., 2026). Our work belongs to this parallelization thread, but treats adaptive posterior-mean queries as the sequential resource and proves total-variation accuracy for the Gaussian-smoothed data law under only a finite second moment.

Diffusion theory and high accuracy. Stochastic localization (Eldan, 2013) is equivalent, after a coordinate change, to a denoising diffusion (Montanari, 2023); Gaussian-channel information identities provide the posterior estimates used here (El Alaoui & Montanari, 2022; Guo et al., 2005). Weak-assumption convergence analyses established polynomial-accuracy guarantees for score-based sampling (Chen et al., 2023b; Lee et al., 2023), and later work obtained nearly linear dimension dependence from a finite second moment (Benton et al., 2024). The direct high-accuracy input to this paper is FORS (Chen et al., 2026), which corrects a Gaussian predictor through a randomized line integral and a Poisson-product Bernoulli factory. FORS achieves polylogarithmic inverse-accuracy dependence from first-order information, but its diffusion implementation remains sequential. We expose a normalized, stopped transcript for this transition and verify the exact-posterior one-step estimate needed by our argument. Connections to Bernoulli factories and exact diffusion simulation are reviewed in Section A.

Recursive autospeculation. Hu et al. (2025) exactly parallelize a prescribed finite-step Euler/DDPM chain, but their guarantee does not include the chain's discretization error to the target law. The closest prior end-to-end high-accuracy result is Anari et al. (2026). Its diffusion theorem uses recursive sequence-level rejection in the same exact Gaussian-posterior-mean model and attains $\widetilde{O}(\sqrt{n})$ expected adaptive depth. On targets contained in a known ball, however, its stated

expected-query guarantee has polynomial dependence on the covering radius, smoothing, and inverse accuracy. On that subclass, our theorem preserves the same dimension exponents—linear expected queries and square-root expected depth, up to logarithms—while replacing those polynomial terms by polylogarithmic dependence; it also requires only a finite second moment.

Table 1: Closely related oracle-model guarantees. The last two rows achieve ε total-variation accuracy for $\pi * \mathcal{N}(0, \delta I_n)$; the first exactly samples a prescribed Euler/DDPM chain. Write $L_R = 1 + |\log(nR_{\sup}/(\varepsilon\delta))|$ and $L_M = \log\big(\mathrm{e}(n+1)(1+M_2^2/\delta)/\varepsilon\big)$. Anari et al. assume a known support radius $R_{\sup} > 0$; our result assumes zero mean and a known second-moment bound M_2^2.

Result	Expected queries	Expected rounds
Hu et al. (2025)†	$\widetilde{O}(n)$	$\widetilde{O}(n^{2/3})$
Anari et al. (2026)	$O\big(\max\{n/\varepsilon^2, R_{\sup}^2/\delta, R_{\sup}^4/\delta^2\}L_R^2\big)$	$O(\sqrt{n}\,L_R^4)$
This work	$O((n+L_M)L_M^3)$	$O(\sqrt{n}\,L_M^4)$

† For dimension-level comparison, the first row specializes the $O(K)$ expected-query and $O\big(K^{2/3}(\beta nh)^{1/3}\big)$ expected-round bounds of Hu et al. (2025) to $K = \widetilde{O}(n)$, $\beta = O(1)$, and $nh = O(1)$, where h is the largest step size. Hu et al. state the moment condition as $\mathrm{Tr}(\mathrm{Cov}(x_\star)) \le \beta n$; under our normalization $\mathbb{E}x_\star = 0$, one has $\mathrm{Tr}(\mathrm{Cov}(x_\star)) = \mathbb{E}\|x_\star\|^2$. Exactness remains relative to the prescribed discrete chain. The last row follows by summing Theorem B.6 over the $J = O(L_M)$ intervals, using $N = O((n+L_M)L_M)$ from Lemma I.2.

None of these works proves a matching lower bound for every posterior-mean algorithm. A more detailed comparison, including the exact theorem provenance and distinctions between query depth, arithmetic work, and PRAM implementation, appears in Section A.

4 Limitations and Conclusion

The result is an expected-cost statement in an exact posterior-mean oracle model. It treats any finite nonadaptive batch as one round and does not charge for Gaussian or Poisson sampling, vector arithmetic, transcript storage, or numerical precision. It does not analyze a fixed processor budget. Finally, the theorem proves no lower bound and does not claim optimal logarithmic exponents.

Subject to these limitations, the construction separates the two roles often conflated in a diffusion discretization. A frozen Euler block supplies a cheap parallel proposal, while FORS supplies high accuracy. The stopped transcript, causal guard, and levelwise comparison make this division compatible with an exact recursive scheduler, giving near-linear expected oracle queries and square-root expected adaptive oracle depth under only a finite second moment. We expect that score-approximation errors can be propagated through the parallel sampler, and leave a rigorous treatment of this extension to future work.

AI use statement

We used generative AI tools to help formulate mathematical claims, identify proof ingredients, assist in writing and checking proofs, and provide feedback on research methodology. We also used them to draft and restructure text, improve readability, search for and summarize literature, identify relevant references, format citations, create diagrams, and check LaTeX consistency. Data generation, empirical experimentation, and empirical-result interpretation are not applicable to this work. The authors manually reviewed the mathematical arguments against the complete proofs and cited sources, reviewed all AI-assisted text and artifacts, and take responsibility for the final content.

Reproducibility statement

The theorem is stated with its complete oracle assumptions in the main text. The appendices give explicit algorithms, all parameter choices, and proofs of the new interfaces used by the main argument. Theorem D.4 identifies the coordinates for the cited FORS estimate. An accompanying

Lean 4 development proves the paper-facing main-theorem contract for a concrete candidate-first protocol and exact law and prefix-state bridges to the Euler-first presentation used here. Its guard stores a larger sufficient raw-grid cache than the compact paper presentation; these representation choices leave the theorem statement and guarantees unchanged.

REFERENCES

Beatrice Acciaio, Mathias Beiglböck, Friedrich Penkner, Walter Schachermayer, and Josef Temme. A trajectorial interpretation of doob's martingale inequalities. *The Annals of Applied Probability*, 23(4):1494–1505, 2013. doi: 10.1214/12-AAP878.

Nima Anari, Sinho Chewi, and Thuy-Duong Vuong. Fast parallel sampling under isoperimetry. In *Proceedings of the Thirty-Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pp. 161–185. PMLR, 2024a.

Nima Anari, Ruiquan Gao, and Aviad Rubinstein. Parallel sampling via counting. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pp. 537–548. Association for Computing Machinery, 2024b. doi: 10.1145/3618260.3649744.

Nima Anari, Carlo Baronio, CJ Chen, Alireza Haqi, Frederic Koehler, Anqi Li, and Thuy-Duong Vuong. Parallel sampling via autospeculation. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, pp. 1168–1179. Association for Computing Machinery, 2026. doi: 10.1145/3798129.3800828. Technical references are to arXiv version 1, 11 November 2025.

Runsheng Bai, Chengyu Zhang, and Yangdong Deng. DRiffusion: Draft-and-refine process parallelizes diffusion models with ease. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16551–16560, 2026.

Joe Benton, Valentin De Bortoli, Arnaud Doucet, and George Deligiannidis. Nearly d-linear convergence bounds for diffusion models via stochastic localization. In *The Twelfth International Conference on Learning Representations*, 2024.

Alexandros Beskos and Gareth O. Roberts. Exact simulation of diffusions. *The Annals of Applied Probability*, 15(4):2422–2444, 2005. doi: 10.1214/105051605000000485.

Valentin De Bortoli, Alexandre Galashov, Arthur Gretton, and Arnaud Doucet. Accelerated diffusion models via speculative sampling. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 12590–12631. PMLR, 2025.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling, 2023a.

Fan Chen, Sinho Chewi, Constantinos Daskalakis, and Alexander Rakhlin. High-accuracy sampling for diffusion models and log-concave distributions. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. Outstanding Paper Award; technical references are to arXiv version 2, 27 April 2026.

Haoxuan Chen, Yinuo Ren, Lexing Ying, and Grant M. Rotskoff. Accelerating diffusion models with parallel sampling: Inference at sub-linear time complexity. In *Advances in Neural Information Processing Systems*, volume 37, pp. 133661–133709, 2024. doi: 10.52202/079017-4248.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R. Zhang. Sampling is as easy as learning the score: Theory for diffusion models with minimal data assumptions. In *The Eleventh International Conference on Learning Representations*, 2023b.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin C. M. Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.026.

Ahmed El Alaoui and Andrea Montanari. An information-theoretic view of stochastic localization. *IEEE Transactions on Information Theory*, 68(11):7423–7426, 2022. doi: 10.1109/TIT.2022.3180298.

Ronen Eldan. Thin shell implies spectral gap up to polylog via a stochastic localization scheme. *Geometric and Functional Analysis*, 23(2):532–569, 2013. doi: 10.1007/s00039-013-0214-y.

Khashayar Gatmiry, Sitan Chen, and Adil Salim. High-accuracy and dimension-free sampling with diffusions. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. Technical references are to arXiv version 1, 15 January 2026.

Dongning Guo, Shlomo Shamai, and Sergio Verdú. Mutual information and minimum mean-square error in gaussian channels. *IEEE Transactions on Information Theory*, 51(4):1261–1282, 2005. doi: 10.1109/TIT.2005.844072.

Shivam Gupta, Linda Cai, and Sitan Chen. Faster diffusion sampling with randomized midpoints: Sequential and parallel. In *The Thirteenth International Conference on Learning Representations*, pp. 97663–97698, 2025. Technical references are to arXiv version 2, 17 October 2024.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Hengyuan Hu, Aniket Das, Dorsa Sadigh, and Nima Anari. Diffusion models are secretly exchangeable: Parallelizing DDPMs via auto speculation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 24270–24289. PMLR, 2025.

Xunpeng Huang, Difan Zou, Hanze Dong, Yi Zhang, Yian Ma, and Tong Zhang. Reverse transition kernel: A flexible framework to accelerate diffusion inference. In *Advances in Neural Information Processing Systems*, volume 37, pp. 95515–95578, 2024. doi: 10.52202/079017-3027.

Xunpeng Huang, Yingyu Lin, Lijing Kuang, Hanze Dong, Difan Zou, Yian Ma, and Tong Zhang. Almost linear convergence under minimal score assumptions: Quantized transition diffusion. In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 3441–3487. PMLR, 2026.

Nishant Jain and Tong Zhang. A sharp KL-convergence analysis for diffusion models under minimal assumptions. In *The Fourteenth International Conference on Learning Representations*, 2026.

M. S. Keane and George L. O'Brien. A bernoulli factory. *ACM Transactions on Modeling and Computer Simulation*, 4(2):213–219, 1994. doi: 10.1145/175007.175019.

Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence of score-based generative modeling for general data distributions. In *Proceedings of the 34th International Conference on Algorithmic Learning Theory*, volume 201 of *Proceedings of Machine Learning Research*, pp. 946–985. PMLR, 2023.

Gen Li and Changxiao Cai. Provable acceleration for diffusion models under minimal assumptions, 2024.

Gen Li and Yuling Yan. $O(d/T)$ convergence theory for diffusion probabilistic models under minimal assumptions. *Journal of Machine Learning Research*, 26(292):1–55, 2025.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural Information Processing Systems*, volume 35, pp. 5775–5787, 2022. doi: 10.52202/068431-0418.

Andrea Montanari. Sampling, diffusions, and stochastic localization, 2023.

Şerban Nacu and Yuval Peres. Fast simulation of new coins from old. *The Annals of Applied Probability*, 15(1A):93–115, 2005. doi: 10.1214/105051604000000549.

Tianxiang Pan, Baitao Gong, Mo Guang, Hongwei Yong, Tianpeng Jiang, Yaqian Li, Zheng Cao, and Kaiwen Long. Trajectory-level speculative decoding for diffusion language models, 2026. arXiv:2608.27514.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Nikil Roashan Selvam, Amil Merchant, and Stefano Ermon. Self-refining diffusion samplers: Enabling parallelization via parareal iterations. In *Advances in Neural Information Processing Systems*, volume 37, pp. 5429–5453, 2024. doi: 10.52202/079017-0176.

Andy Shih, Suneel Belkhale, Stefano Ermon, Dorsa Sadigh, and Nima Anari. Parallel sampling of diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4263–4276, 2023. doi: 10.52202/075280-0188.

Junhyuk So, Jiwoong Shin, Chaeyeon Jang, and Eunhyeok Park. PCM: Picard Consistency Model for fast parallel sampling of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23313–23322, 2025.

Alexander Soen, Hisham Husain, Valentin De Bortoli, and Arnaud Doucet. Accelerating speculative diffusions via block verification, 2026. Version 1, 11 June 2026.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265. PMLR, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, pp. 11895–11907, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 2023.

Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit. Blockwise parallel decoding for deep autoregressive models. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Zhiwei Tang, Jiasheng Tang, Hao Luo, Fan Wang, and Tsung-Hui Chang. Accelerating parallel sampling of diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 47800–47818. PMLR, 2024.

Martin J. Wainwright. Fast score-based sampling via log-concave reductions. In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 6592–6621. PMLR, 2026.

Huanjian Zhou and Masashi Sugiyama. Parallel simulation for log-concave sampling and score-based diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79192–79225. PMLR, 2025.

A Extended related work

This appendix supplies context omitted from the compressed discussion in Section 3. It also makes explicit which comparisons are and are not meaningful in our oracle model.

A.1 Different notions of fast diffusion sampling

Diffusion probabilistic models reverse a gradual noising process. Early modern formulations learned either finite reverse dynamics or the score of Gaussian-smoothed data laws (Sohl-Dickstein et al., 2015; Song & Ermon, 2019); DDPM connected this picture directly to denoising score matching (Ho et al., 2020), and the continuous-time framework unified reverse SDEs and probability-flow ODEs (Song et al., 2021b). These constructions now support applications ranging from latent image synthesis to diffusion policies (Rombach et al., 2022; Chi et al., 2023), making inference latency a practical concern.

The word "fast" can mean fewer neural evaluations, lower latency for one sample, lower total arithmetic work, or less memory. It can also refer to error against the data law, error against a selected discrete chain, or an empirical quality metric. Step-reduction methods such as DDIM and DPM-Solver (Song et al., 2021a; Lu et al., 2022), and model-changing methods such as progressive distillation and consistency models (Salimans & Ho, 2022; Song et al., 2023), chiefly reduce serial neural evaluations in practical generative models. Our theorem instead counts full-vector oracle calls and adaptive oracle stages, and measures error in total variation against the intended Gaussian-smoothed law. It therefore does not dominate, and is not dominated by, empirical compute–quality guarantees.

A.2 Weak assumptions and high accuracy

The Gaussian-observation process in this paper is Eldan's stochastic localization (Eldan, 2013). Its relation to denoising diffusion is developed by Montanari (2023), while the posterior and information identities used in our normalization appear in El Alaoui & Montanari (2022) and Guo et al. (2005). General-distribution analyses first proved polynomial convergence guarantees from square-integrable score error (Chen et al., 2023b; Lee et al., 2023). Subsequent work obtained nearly linear dimension dependence under a finite second moment (Benton et al., 2024), and sharpened the step–accuracy analysis of standard DDPM transitions (Li & Yan, 2025; Jain & Zhang, 2026). These results establish the minimal-assumption regime motivating our target class, but an Euler-type accuracy analysis still leaves polynomial inverse-accuracy dependence.

Other high-accuracy approaches change the transition or the reduction. Examples include accelerated minimal-assumption sampling (Li & Cai, 2024), reverse-transition reductions to strongly log-concave subproblems (Huang et al., 2024; Wainwright, 2026), quantized transition diffusion (Huang et al., 2026), and approximation-based guarantees for smoothed bounded-support laws (Gatmiry et al., 2026). Their learned primitives, assumptions, or subproblems differ from ours, so their stated complexities are not direct posterior-mean-oracle comparisons.

The direct high-accuracy input here is FORS (Chen et al., 2026). It represents a log tilt by a randomized line integral and converts independent estimators into a rejection bit through a Poisson-product identity. This connects to Bernoulli factories (Keane & O'Brien, 1994; Nacu & Peres, 2005) and to Poisson devices for exact diffusion simulation (Beskos & Roberts, 2005). Our stopped transcript has two purposes not present in the endpoint description: it makes the normalized likelihood finitely evaluable by the recursive scheduler, and it keeps expected oracle occupation independent of the safety caps. The causal guard solves a separate change-of-measure problem and is not part of the FORS factory. The exact-score estimate used here follows from Chen et al. (2026, Proposition F.3); the proof of Theorem D.4 identifies its variables with ours.

A.3 Parallel time and speculative correction

Practical parallel-in-time samplers keep several trajectory locations in flight. ParaDiGMS applies Picard refinement, ParaTAA accelerates a triangular fixed-point problem, and Parareal variants iteratively correct multiple time slices (Shih et al., 2023; Tang et al., 2024; Selvam et al., 2024). Theoretical blockwise Picard, randomized-midpoint, and later simulation analyses give parallel

guarantees under regularity and score-accuracy assumptions (Chen et al., 2024; Gupta et al., 2025; Zhou & Sugiyama, 2025). Some have lower dimension dependence than our bound, but use different smoothness hypotheses, accuracy metrics, or parallel-space accounting. Parallel Langevin sampling under log-Sobolev and smooth-potential assumptions provides another structural counterpart (Anari et al., 2024a).

Recent empirical systems pursue complementary latency goals. PCM trains a consistency predictor for the Picard fixed point and uses model switching, whereas DRiffusion uses skip transitions to draft future diffusion states and refines them with the original denoiser (So et al., 2025; Bai et al., 2026). They report wall-clock speed and empirical output quality rather than total-variation guarantees in an exact-oracle model.

Draft-and-verify methods began in blockwise autoregressive decoding (Stern et al., 2018); speculative sampling then supplied a target-preserving rejection correction for language models (Chen et al., 2023a). Anari et al. (2024b) brought same-oracle autospeculation into worst-case sampling theory. For diffusion chains, Bortoli et al. (2025) verify continuous vector-valued drafts against a selected discrete target, Hu et al. (2025) use stochastic localization to parallelize a finite-step DDPM without a separate draft model, and Soen et al. (2026) study improved block verification and a training-free drafter. Exactness relative to a chosen discrete chain does not by itself give high-accuracy total variation to the undiscretized target; our FORS comparison supplies that additional layer.

Trajectory-level speculative decoding has also been developed for discrete diffusion language models, where a tree of candidate denoising trajectories is verified through blockwise evaluation with bidirectional attention masking (Pan et al., 2026). This is structurally related to trajectory drafting, but its discrete-token verification problem differs from continuous-state diffusion sampling.

More precisely, specialize Hu et al. (2025) to our zero-mean normalization and suppose $\mathbb{E}\|x_\star\|^2 \le \beta n$. For a K-step chain with maximum step size h, they prove exactness for that chain and expected adaptive complexity $O\big(K^{2/3}(\beta n h)^{1/3}\big)$. Their optimized algorithm uses $O(K)$ expected model evaluations in total. These bounds underlie the Hu row of Table 1, which further specializes to $K = \widetilde{O}(n)$, $\beta = O(1)$, and $nh = O(1)$. Neither statement controls the discretization error between the chosen chain and the target law.

A.4 Precise relation to recursive autospeculation

The direct predecessor is Anari et al. (2026). Its diffusion theorem uses recursive sequence-level rejection in the same exact Gaussian-posterior-mean model; a posterior mean for a variance-v observation at z is $f(1/v, z/v)$ in our coordinates. The theorem, stated for targets in a known covering ball, has $\widetilde{O}(\sqrt{n})$ expected adaptive depth and an expected-query bound with polynomial dependence on inverse accuracy, smoothing, and the covering radius. We use the theorem as stated in arXiv v1, as specified in the bibliography.

On this bounded-support subclass one may take $M_2^2 \le R_{\mathrm{sup}}^2$. Our result keeps the predecessor's dimension exponents while replacing the polynomial dependence by logarithmic dependence on inverse accuracy and $1+R_{\mathrm{sup}}^2/\delta$; within our zero-mean normalization it also replaces bounded support by a finite second moment. Conversely, Anari et al. give a PRAM implementation interpretation, whereas Theorem 2.1 makes only the exact-real oracle claim.

At each recursive node, the overlap step is a maximal coupling: for densities p, q relative to a common measure λ, its one-shot agreement mass is $\int \min\{p, q\}\, \mathrm{d}\lambda = 1 - d_{\mathrm{TV}}(P, Q)$, the largest possible for any coupling of P and Q. This nodewise fact explains why the local acceptance loss is exactly total variation. Neither paper proves a model-wide lower bound of $\widetilde{\Omega}(n)$ queries or $\widetilde{\Omega}(\sqrt{n})$ rounds.

B The three path laws: construction, comparison, and composition

This appendix gives the exact versions of the claims used in the main-paper roadmap. Each of its three subsections begins with a proof-location guide to the detailed arguments that follow. Throughout we are in the nontrivial branch of Definition I.1, but retain only the structural grid notation needed here. Its fine grid is split into J consecutive coarse intervals, each containing the same power-of-two

number N of single-step transitions. Write $M \coloneqq JN$ for the total number of transitions, so

$$t_0 < t_1 < \cdots < t_M = T, \qquad \mathcal{I}_j \coloneqq \{jN, \ldots, (j+1)N - 1\}, \quad 0 \le j < J.$$

The concrete time spacings, transcript caps, and batching rate are specified in Definition I.1; the guard thresholds are defined in Section F. None is needed to read the interfaces below. Every claim tied to our sampler uses those prescribed choices, whose numerical conditions are verified in Lemma I.2.

Let $\mathbf{P}$ be the law of the true localization path $(Y_{t_i})_{i=0}^{M}$. Starting from $X_0 \sim \mathcal{N}(0, t_0 I_n)$, let $\mathbf{Q}_0$ be the selected-state path law obtained from the unguarded stopped, capped FORS transition at every grid step. Let $\mathbf{Q}_g^{\mathrm{tr}}$ be the joint law of the Gaussian initialization and the guarded stopped transition transcripts, and let $\mathbf{Q}_g$ be its decoded selected-state-path law, with path $(X_i^g)_{i=0}^{M}$, and set

$$\mathfrak{d}_{\mathrm{path}} \coloneqq D_{\mathrm{KL}}(\mathbf{P} \,\|\, \mathbf{Q}_0)\,.$$

The complete sampler returns $X_{\mathrm{out}} \coloneqq X_M^g / T$.

B.1 Global path comparisons

Proof locations. Proposition B.1 is proved in Section E.4; Propositions B.2 and B.3 are proved together in Section F.4. The shared numerical conditions are checked in Section I.2.

The next proposition is obtained from the verified exact-posterior one-step bound in Theorem D.4, finite-cap domination, and the relative entropy chain rule.

Proposition B.1 (Corrected path accuracy)**.** *For the specification in Definition I.1,*

$$\mathfrak{d}_{\mathrm{path}} \le \frac{\varepsilon^2}{32}, \qquad d_{\mathrm{TV}}(\mathbf{P}, \mathbf{Q}_0) \le \frac{\varepsilon}{8}.$$

Guard rule. Within a coarse interval, the binary fallback tree partitions the N local steps into nested consecutive subblocks. Before admitting the tentative drift $f(t_i, X_i^g)$, the guard considers every subblock containing the new step. At each tree level it computes, from the prefix seen so far, the sum over started subblocks of the largest squared change from that subblock's first drift. It admits the tentative value only when all these prefix quantities remain below the interval thresholds prescribed in Definition I.1. Otherwise it discards the tentative value, repeats the preceding admitted drift from that step onward, and makes the current and all later FORS correction factors neutral. The chain continues rather than halting. Interval-local variation records reset at the next coarse boundary, whereas the frozen flag persists globally. Write b_i for the drift admitted at step i, and let A_g be the event that the guard ever triggers. The exact prefix functional and thresholds are given in Section F.

Proposition B.2 (Accuracy cost of guarding)**.** *Under the guard rule above,* $\mathbb{P}_{\mathbf{P}}(A_g) \le \varepsilon/4$*. Moreover,*

$$\mathbb{P}_{\mathbf{Q}_0}(A_g) \le \frac{\varepsilon}{4} + \sqrt{\mathfrak{d}_{\mathrm{path}}/2}, \qquad d_{\mathrm{TV}}(\mathbf{Q}_0, \mathbf{Q}_g) \le \frac{\varepsilon}{4} + \sqrt{\mathfrak{d}_{\mathrm{path}}/2},$$

and consequently

$$d_{\mathrm{TV}}(\mathrm{Law}(X_M^g), \mathrm{Law}(Y_T)) \le \frac{\varepsilon}{4} + 2\sqrt{\mathfrak{d}_{\mathrm{path}}/2} \le \varepsilon.$$

Thus Proposition B.2 isolates the accuracy cost of the intervention.

Proposition B.3 (Guarded drift stability)**.** *Fix any coarse interval j and any level of its binary tree. Under $\mathbf{Q}_g$, the interval's time length multiplied by the expected aggregate drift variation at that level is $O(n)$. Explicitly, if $\mathscr{B}$ denotes the subblocks at that level, viewed as subsets of $\{0, \ldots, N-1\}$, then*

$$(t_{(j+1)N} - t_{jN})\,\mathbb{E}_{\mathbf{Q}_g}\left[\sum_{B \in \mathscr{B}} \max_{k \in B} \|b_{jN+k} - b_{jN+\min B}\|^2\right] = O(n).$$

The stronger analysis-only bound and the cache implementation are given in Section F.

This stability estimate is the input used in the rejection analysis.

B.2 Block proposal and rejection interfaces

Proof locations. Proposition B.4 is proved in Section G.2. For Proposition B.5, the blockwise KL estimate is established in Section G.3 and aggregated over each tree level in Section G.4.

An incoming history and guard cache are *cache-consistent* when replaying the accepted history produces exactly that cache.

Proposition B.4 (Constant-oracle-depth block interface)**.** *Fix a non-leaf consecutive block $B = \{s, s+1, \dots, s+m-1\}$ in one coarse interval and condition on a cache-consistent incoming history $\mathfrak{h}_{<B}$ and its guard cache. The pure Euler primitive in Algorithm 3, once supplied the admitted block-head drift, simultaneously generates the m frozen proposal increments and recovers all $m+1$ speculative states by a parallel prefix scan; it uses no oracle call. The complete block interface in Algorithm 4 obtains that anchor, augments the Euler increments into a transcript Z_B with law $\nu_B^{\mathfrak{h}_{<B}}$, and evaluates the normalized likelihood ratio*

$$\frac{\mathrm{d}\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}}{\mathrm{d}\nu_B^{\mathfrak{h}_{<B}}}(\mathsf{Z}_B).$$

The complete interface uses $O(1)$ adaptive oracle stages and $O(m)$ expected full-vector queries, uniformly over the finite transcript caps used by the sampler. On every supplied transcript in the common support, the separately callable scorer evaluates both likelihood ratios in $O(1)$ adaptive stages; under either the target or proposal law it uses $O(|B|)$ expected queries. At a leaf, ExactGuardedStep *returns the exact conditional target transcript and outgoing guard cache in $O(1)$ oracle stages and $O(1)$ expected queries.*

Proposition B.5 (Levelwise total-variation certificate)**.** *Fix any coarse interval and any internal level of its binary tree. For a node B at that level, draw $(X_0^g, \mathsf{Z}) \sim \mathbf{Q}_g^{\mathrm{tr}}$ and let $\mathcal{H}_{<B}$ denote the accepted history preceding B. Let $\mathbf{Q}_{g,B}^{\mathcal{H}_{<B}}$ be the conditional guarded target law and $\nu_B^{\mathcal{H}_{<B}}$ its speculative FORS proposal. Then*

$$\mathbb{E}_{\mathbf{Q}_g^{\mathrm{tr}}} \sum_{B \text{ at that level}} d_{\mathrm{TV}}\left(\mathbf{Q}_{g,B}^{\mathcal{H}_{<B}}, \nu_B^{\mathcal{H}_{<B}}\right) = O(\sqrt{n+1}). \tag{1}$$

The expectation includes the actual guarded history entering the interval and the conditional target path within it. No pointwise bound is asserted for a fixed exceptional entering history. At leaves, the proposal is the exact conditional target, so its rejection probability is zero.

B.3 Conditioning and composition across coarse intervals

Proof locations. Theorem B.6 is proved in Section H.3. The generic scheduler statement Proposition B.7 is proved in Section H.2, with a self-contained fresh-call verification of its work and depth recurrences in Section J.

Recall that the grid is divided into the consecutive coarse intervals $\mathcal{I}_0, \dots, \mathcal{I}_{J-1}$ defined at the beginning of this appendix. Interval j runs from t_{jN} to $t_{(j+1)N}$ and contains N adjacent single-step transitions. The complete sampler processes these intervals chronologically,

$$\mathcal{I}_0 \longrightarrow \mathcal{I}_1 \longrightarrow \cdots \longrightarrow \mathcal{I}_{J-1}.$$

On interval $\mathcal{I}_j$, it runs the recursive sampler on the binary tree whose N leaves are the single-step transitions in that interval. Once the resulting interval transcript is accepted, the sampler commits it and, when $j < J-1$, moves to $\mathcal{I}_{j+1}$. After the final interval it returns the terminal state. Thus the binary recursion occurs *inside* one coarse interval; the coarse intervals themselves are processed sequentially.

Information passed between intervals. For interval $\mathcal{I}_j$, write

$$\mathsf{Z}^{(j)} := (\mathsf{Z}_i)_{i \in \mathcal{I}_j}$$

for its accepted transcript. Before starting $\mathcal{I}_j$, the sampler carries two pieces of information. First, $\mathcal{H}_j$ consists of the initialization and all accepted transcripts from $\mathcal{I}_0, \dots, \mathcal{I}_{j-1}$; these transcripts determine all previously accepted states. Second, $\mathfrak{c}_j$ is the guard cache after the preceding boundary resets. It stores the current boundary state X^g_{jN}, whether the guard has permanently triggered, and the last admitted drift. We use the single shorthand

$$\mathsf{Entry}_j := (\mathcal{H}_j, \mathfrak{c}_j).$$

The cache is deterministic given the accepted history and is carried explicitly only to avoid replaying earlier oracle calls.

After accepting a transcript z on $\mathcal{I}_j$, the sampler appends z to the history, advances the current state to $X^g_{(j+1)N}$, retains the permanent frozen flag and last admitted drift, and resets the guard fields used only inside $\mathcal{I}_j$. Denote this deterministic boundary update by $\mathsf{Next}_j(e, z)$. Its output is exactly the information needed to start $\mathcal{I}_{j+1}$, or the terminal entry when $j = J - 1$. Under the guarded law

$$\mathsf{Entry}_{j+1} = \mathsf{Next}_j(\mathsf{Entry}_j, \mathsf{Z}^{(j)}) \quad \text{almost surely.}$$

In short, one pass through an interval has the interface

$$\mathsf{Entry}_j \xrightarrow{\text{recursive scheduler on } \mathcal{I}_j} \mathsf{Z}^{(j)} \xrightarrow{\mathsf{Next}_j} \mathsf{Entry}_{j+1}.$$

Theorem B.6 (Exact sampling on one coarse interval)**.** *Fix $n \geq 1$, $\delta > 0$, and a target accuracy $0 < \varepsilon \leq \varepsilon_0$ as in Theorem 2.1. Let $x_\star \sim \pi$ satisfy $\mathbb{E}x_\star = 0$ and suppose a finite bound $M_2^2 \geq \mathbb{E}\|x_\star\|^2$ is known. Assume exact access to the posterior-mean oracle $f(t, y)$. Run* ParameterSetup$(n, \delta, \varepsilon, M_2^2)$ *from Definition I.1, assume it returns its nontrivial* Recursive *specification, and use its grid, transcript caps, guard thresholds, and batching rate. Here ε is the requested final total-variation accuracy for $\pi * \mathcal{N}(0, \delta I_n)$; the conditional interval sampling statement below is exact.*

For every $0 \leq j < J$ and every cache-consistent entry e, the root call of the recursive scheduler in Algorithm 6 on the N local transitions with an empty accepted prefix returns an exact sample from the conditional guarded transcript law on $\mathcal{I}_j$ given $\mathsf{Entry}_j = e$. Moreover, $\big(\mathsf{Z}^{(j)}, \mathsf{Next}_j(e, \mathsf{Z}^{(j)})\big)$ has the exact conditional joint law of the guarded interval transcript and the complete entry to interval $j + 1$.

Let Q_j and D_j be the costs when the incoming entry is distributed as produced by the preceding intervals. Then

$$\mathbb{E}[\mathsf{Q}_j] = O(N \log(N + 1)), \qquad \mathbb{E}[\mathsf{D}_j] = O\big((1 + \sqrt{n + 1}) \log^3(N + 1)\big).$$

After substituting the prescribed value of N, these are respectively $\widetilde{O}(n)$ and $\widetilde{O}(\sqrt{n})$; here $\widetilde{O}$ hides factors polylogarithmic in $n + 1$, ε^{-1}, and $1 + M_2^2/\delta$.

The query estimate holds uniformly after conditioning on every cache-consistent e. The depth estimate is averaged over the entries that the sampler actually reaches.

Starting from the Gaussian initialization, apply Theorem B.6 successively to $\mathcal{I}_0, \dots, \mathcal{I}_{J-1}$. If the entry to $\mathcal{I}_j$ has the correct law, exact conditional sampling on that interval followed by the deterministic boundary update gives the correct entry to $\mathcal{I}_{j+1}$. Induction therefore reconstructs the complete law $\mathbf{Q}_g^{\mathrm{tr}}$. The intervals need not be independent, and no approximation error accumulates between them; only their query counts and adaptive depths are added.

For completeness, the scheduler statement used in the proof is stated next.

Proposition B.7 (Recursive autospeculation scheduler)**.** *Let a perfect binary fallback tree have m leaves, exact leaf proposals, constant pointwise local depth, conditional expected proposal and residual-test cost $O(q_{\mathrm{loc}}|B|)$ at every node, and exact-leaf cost $O(q_{\mathrm{loc}})$. Let $\mathfrak{R}_\star$ be the largest, over tree levels, of the expected sum of conditional rejection probabilities at that level. With the geometric batching rate chosen as the reciprocal of the tree height (and rate one for a one-leaf tree), the scheduler samples the target law exactly and satisfies*

$$\mathbb{E}[\mathsf{Q}] = O(q_{\mathrm{loc}} m \log(m+1)), \qquad \mathbb{E}[\mathsf{D}] = O\big((1+\mathfrak{R}_\star)\log^3(m+1)\big). \tag{2}$$

C PRELIMINARIES

This section fixes the Gaussian-channel normalization and collects the probabilistic inputs used below. Stochastic localization originates with Eldan (2013); its connection to denoising diffusion is developed by Montanari (2023). For the channel realization, posterior description, and sufficiency statement used here, see El Alaoui & Montanari (2022, Section 3). The exact denoiser coordinates also appear in Anari et al. (2026, Section 6). We cite those results and Guo et al. (2005) rather than reproducing their proofs, and give only the short specializations needed for our conventions and bounds.

C.1 THE GAUSSIAN LOCALIZATION CHANNEL

Let $x_\star \sim \pi$, so that $\mathbb{E}x_\star = 0$, and let $(B_t)_{t\ge 0}$ be a standard Brownian motion in $\mathbb{R}^n$, independent of $x_\star$. Define

$$Y_t = t x_\star + B_t.$$

For $t > 0$, introduce the partition function

$$Z_t(y) := \int_{\mathbb{R}^n} \exp\left(\langle y, x\rangle - \frac{t}{2}\|x\|^2\right) \pi(\mathrm{d}x). \tag{3}$$

For every $y \in \mathbb{R}^n$, define the canonical posterior law

$$\pi_t^y(\mathrm{d}u) := \frac{\exp\left(\langle y, u\rangle - \frac{t}{2}\|u\|^2\right)}{Z_t(y)}\pi(\mathrm{d}u).$$

Completing the square shows that the integrand remains bounded after multiplication by any fixed polynomial in x. Thus $Z_t(y)$ is finite and positive, π_t^y is well defined, differentiation under the integral is valid, and the posterior mean has the pointwise version

$$f(t,y) := \int u\, \pi_t^y(\mathrm{d}u) = \nabla \log Z_t(y)$$

holds pointwise for every $t > 0$ and $y \in \mathbb{R}^n$.

Let $\mathcal{F}_t = \sigma(Y_s : 0 \le s \le t)$. Sufficiency of Y_t for the constant signal observed in Brownian noise gives the square-integrable martingale

$$\widehat{x}_{\star,t} := f(t, Y_t) = \mathbb{E}[x_\star \mid \mathcal{F}_t].$$

Set

$$m(t) := \mathbb{E}\|x_\star - \widehat{x}_{\star,t}\|^2, \qquad \widehat{x}_{\star,0} := 0, \qquad m(0) := \mathbb{E}\|x_\star\|^2.$$

Lemma C.1 (Standard localization identities)**.** *For $0 \le s \le t$,*

$$\mathbb{E}\|\widehat{x}_{\star,t} - \widehat{x}_{\star,s}\|^2 = m(s) - m(t). \tag{4}$$

Furthermore, for every $t \ge 0$,

$$m(t) \le \frac{M_2^2}{1 + tM_2^2/n} \le \min\{M_2^2, n/t\}, \tag{5}$$

where n/t is interpreted as $+\infty$ at $t = 0$. Finally, for $T = \delta^{-1}$,

$$Y_T/T \sim \pi * \mathcal{N}(0, \delta I_n).$$

The martingale and energy identities are the conditional-expectation formulation of stochastic localization (El Alaoui & Montanari, 2022, Section 3); the first display is the L^2 Pythagorean identity. For the precise trace-only bound used here, let $\Sigma = \mathbb{E}[x_\star x_\star^\top]$ have eigenvalues $\lambda_1, \dots, \lambda_n$. The posterior mean has no larger risk than the optimal linear estimator in the equivalent channel $\sqrt{t}\, x_\star + G$, where $G \sim \mathcal{N}(0, I_n)$ is independent of $x_\star$. That estimator's risk is

$$\sum_{k=1}^n \frac{\lambda_k}{1 + t\lambda_k} \le n \frac{M_2^2/n}{1 + tM_2^2/n};$$

see also Guo et al. (2005, Sections II-B and II-D). The inequality is Jensen's inequality and monotonicity of $u/(1+tu)$, using $\sum_k \lambda_k \le M_2^2$; the endpoint law follows directly from $Y_T/T = x_\star + B_T/T$.

C.2 Initialization

The singular time $t = 0$ is avoided by starting from a small positive precision.

Lemma C.2 (Gaussian initialization). *For every $t > 0$,*

$$D_{\mathrm{KL}}(\mathrm{Law}(Y_t) \,\|\, \mathcal{N}(0, tI_n)) \le \frac{t}{2}\mathbb{E}\|x_\star\|^2 \le \frac{tM_2^2}{2}.$$

Consequently, the setup quantities t_0 and ζ defined in (53) and (54) satisfy

$$D_{\mathrm{KL}}(\mathrm{Law}(Y_{t_0}) \,\|\, \mathcal{N}(0, t_0 I_n)) \le \zeta/4.$$

Indeed, scaling Y_t by $t^{-1/2}$ and applying the Gaussian-mixture entropy identity of Guo et al. (2005, Equation (65)) gives, with $I(\,\cdot\,;\,\cdot\,)$ denoting mutual information,

$$D_{\mathrm{KL}}(\mathrm{Law}(Y_t) \,\|\, \mathcal{N}(0, tI_n)) = \frac{t}{2}\mathbb{E}\|x_\star\|^2 - I(x_\star; \sqrt{t}\, x_\star + G),$$

which proves the first claim by nonnegativity of mutual information. The second follows from $t_0 M_2^2 \le \zeta/2$, which also holds when $M_2^2 = 0$.

C.3 One-step information inequalities

Fix $t > 0$, a step $a > 0$, and a state $x \in \mathbb{R}^n$. Draw $\widetilde{x}_\star \sim \pi_t^x$ and $G' \sim \mathcal{N}(0, I_n)$ independently, and let

$$b := f(t, x), \qquad \mu_{t,a}^x := \mathrm{Law}\big(x + a\widetilde{x}_\star + \sqrt{a}\, G'\big), \qquad q_{t,a}^x := \mathcal{N}(x + ab, aI_n). \tag{6}$$

By Bayes' rule and the independent increments of Brownian motion, $\mu_{t,a}^x$ is the canonical version of $\mathrm{Law}(Y_{t+a} \mid Y_t = x)$ for every $x \in \mathbb{R}^n$. Define the posterior-mean increment energy

$$\Delta_{t,a}(x) := \mathbb{E}_{Y \sim \mu_{t,a}^x}\|f(t + a, Y) - b\|^2.$$

Lemma C.3 (Gaussian-mixture KL). *For every t, a, x as above,*

$$D_{\mathrm{KL}}\big(\mu_{t,a}^x \,\big\|\, q_{t,a}^x\big) \le \frac{a}{2}\Delta_{t,a}(x).$$

To map the cited identities to this notation, for $s \ge 0$ define

$$\mathrm{mmse}_x(s) := \mathbb{E}\big\|\widetilde{x}_\star - \mathbb{E}[\widetilde{x}_\star \mid \sqrt{s}\,\widetilde{x}_\star + G']\big\|^2.$$

Applying the mean-matched Gaussian-mixture identity (Guo et al., 2005, Equation (65)) to the centered input $\sqrt{a}\,(\widetilde{x}_\star - b)$, followed by vector I–MMSE (Guo et al., 2005, Theorem 2), gives

$$\begin{aligned} D_{\mathrm{KL}}\big(\mu_{t,a}^x \,\big\|\, q_{t,a}^x\big) &= \frac{1}{2}\int_0^a \big(\mathrm{mmse}_x(0) - \mathrm{mmse}_x(s)\big)\,\mathrm{d}s \\ &\le \frac{a}{2}\big(\mathrm{mmse}_x(0) - \mathrm{mmse}_x(a)\big) = \frac{a}{2}\Delta_{t,a}(x). \end{aligned}$$

Here monotonicity of MMSE gives the inequality, and the last equality is the conditional-variance decomposition. Finite second moment suffices for the cited vector identity; the positive-precision Gaussian tilt above makes all posterior quantities finite.

We will also transfer bounded functionals from a true path to an approximate path. The following elementary asymmetric form matches the available divergence orientation.

Lemma C.4 (Bounded change of measure). *Let $P \ll Q$ be probability measures and let F be measurable with $0 \le F \le C < \infty$. Then*

$$\mathbb{E}_Q F \le 2\mathbb{E}_P F + 2C D_{\mathrm{KL}}(P \,\|\, Q)\,.$$

Proof. Let $\lambda = P + Q$, $p = \frac{\mathrm{d}P}{\mathrm{d}\lambda}$, and $q = \frac{\mathrm{d}Q}{\mathrm{d}\lambda}$. Minkowski's inequality and $\int(\sqrt{p} - \sqrt{q})^2\,\mathrm{d}\lambda \le D_{\mathrm{KL}}(P \,\|\, Q)$ give

$$\sqrt{\mathbb{E}_Q F} \le \sqrt{\mathbb{E}_P F} + \sqrt{C}\left(\int(\sqrt{p} - \sqrt{q})^2\,\mathrm{d}\lambda\right)^{1/2} \le \sqrt{\mathbb{E}_P F} + \sqrt{C D_{\mathrm{KL}}(P \,\|\, Q)}.$$

Squaring and using $(u + v)^2 \le 2u^2 + 2v^2$ proves the claim. □

D A FIRST-ORDER CORRECTED TRANSITION

The Gaussian predictor $q^x_{t,a}$ in (6) is explicit but is not accurate enough at the relatively coarse time-step sizes needed for near-linear query complexity. We correct it by a bounded random line integral. The resulting endpoint density has an implicit normalizer; the construction in Section E instead gives an explicit normalized density on a finite first-zero-stopped transcript space.

D.1 THE TRANSITION AS A GAUSSIAN TILT

Fix $t > 0$, $a > 0$, and $x \in \mathbb{R}^n$, and set $b = f(t,x) = \int u\, \pi^x_t(\,\mathrm{d}u)$. For every positive-definite covariance matrix $\Sigma \in \mathbb{R}^{n\times n}$, define

$$\varphi_\Sigma : \mathbb{R}^n \to (0,\infty), \qquad \varphi_\Sigma(u) := \frac{1}{(2\pi)^{n/2}(\det\Sigma)^{1/2}} \exp\!\left(-\frac{1}{2}\langle u, \Sigma^{-1}u\rangle\right).$$

Thus φ_Σ is the density of $\mathcal{N}(0,\Sigma)$. By Bayes' rule and (3), the transition kernel $\mu^x_{t,a}$ defined in (6) has, for every $y \in \mathbb{R}^n$, the Lebesgue density

$$\frac{\mathrm{d}\mu^x_{t,a}}{\mathrm{d}y}(y) = \frac{\varphi_{aI_n}(y-x)Z_{t+a}(y)}{Z_t(x)},$$

where I_n is the $n\times n$ identity matrix and Z_s is the partition function in (3). Likewise, the Gaussian predictor $q^x_{t,a}$ from (6) has density

$$\frac{\mathrm{d}q^x_{t,a}}{\mathrm{d}y}(y) = \varphi_{aI_n}(y-x)\exp\!\left(\langle y-x, b\rangle - \frac{a}{2}\|b\|^2\right).$$

Define $\psi(y) := \log Z_{t+a}(y) - \langle b, y\rangle$. Dividing the two densities and normalizing with respect to $q^x_{t,a}$ gives the exact identity

$$\frac{\mathrm{d}\mu^x_{t,a}}{\mathrm{d}q^x_{t,a}}(y) = \frac{\exp\big(\psi(y)\big)}{\int_{\mathbb{R}^n}\exp\big(\psi(u)\big)q^x_{t,a}(\,\mathrm{d}u)}.$$

In particular,

$$\nabla\psi(y) = f(t+a,y) - b.$$

No value of Z_t will be queried.

D.2 A GAUSSIAN-PRESERVING ARC

Let $g = x + ab$. For a candidate $y \sim q^x_{t,a}$, independently sample

$$r \sim \mathrm{Unif}[0,1], \qquad \widehat{x} \sim \mathcal{N}\big(g, \tfrac{a}{2}I_n\big), \qquad z \sim \mathcal{N}\big(0, \tfrac{a}{2}I_n\big).$$

Define

$$\gamma_r(y) := A_r y + (1-A_r)\widehat{x} + B_r z, \tag{7}$$

$$A_r := \frac{1}{3}\left(1 + 2\cos\frac{2\pi(1-r)}{3}\right), \qquad B_r := \frac{2}{\sqrt{3}}\sin\frac{2\pi(1-r)}{3}. \tag{8}$$

Let $\dot\gamma_r$ denote the derivative with respect to r.

Lemma D.1 (Gaussian arc)**.** *The path satisfies $\gamma_0(y) = \widehat{x} + z$ and $\gamma_1(y) = y$. For every fixed $r \in [0,1]$, under the joint candidate and auxiliary law,*

$$\gamma_r(y) \sim \mathcal{N}(g, aI_n), \qquad \dot\gamma_r(y) \sim \mathcal{N}(0, c_{\mathrm{arc}}aI_n), \qquad \gamma_r(y) \perp \dot\gamma_r(y),$$

where

$$c_{\mathrm{arc}} := \frac{8\pi^2}{27} < 3.$$

Proof. This is Chen et al. (2026, Lemma F.1) under the rescaling $Y_u \mapsto Y_u/u$ used in the proof of Theorem D.4. The two half-variance auxiliary laws in the present normalization give the displayed covariances. □

For a real number w, write

$$\mathrm{clip}_{[-1/2,1/2]}(w) \coloneqq \max\{-1/2, \min\{w, 1/2\}\}.$$

Define the raw and clipped corrections

$$W^\circ \coloneqq \langle \dot\gamma_r(y), f(t+a, \gamma_r(y)) - b\rangle, \qquad \overline{W} \coloneqq \mathrm{clip}_{[-1/2,1/2]}(W^\circ). \tag{9}$$

The distinction is essential: the path-integral identity holds for W°, whereas the implemented factory uses $\overline{W}$. The following is the general path-integral identity of Chen et al. (2026, Equation (9)) in our notation.

Lemma D.2 (Random path integral)**.** *There is a constant $c_{t,a,x}$ independent of y such that*

$$\mathbb{E}_{r,\widehat{x},z}[W^\circ \mid y] = \psi(y) + c_{t,a,x}.$$

D.3 The Poisson–Bernoulli factory

The general task of simulating a prescribed Bernoulli probability from samples of another random quantity is known as a Bernoulli factory (Keane & O'Brien, 1994; Nacu & Peres, 2005). The elementary Poisson-product identity below is the particular factory used by FORS. The following is Chen et al. (2026, Algorithm 1 and Theorem 3.1) with factory envelope 1.

For a fixed candidate y, let $\overline{W}_1, \overline{W}_2, \ldots$ be conditionally independent copies of the clipped correction in (9). Draw $J^{\mathrm{Pois}} \sim \mathrm{Poi}(2)$ independently and, for $1 \le k \le J^{\mathrm{Pois}}$, draw

$$E_k \mid \overline{W}_k \sim \mathrm{Ber}\left(\frac{1+\overline{W}_k}{2}\right).$$

The clipping threshold is $1/2$, while the Bernoulli-factory scale is 1. Thus every factor probability belongs to $[1/4, 3/4]$, which will later give mutual absolute continuity of the stopped target and proposal laws on their common stopped support.

Lemma D.3 (Corrected endpoint law)**.** *Accept y if $E_1 = \cdots = E_{J^{\mathrm{Pois}}} = 1$, with the empty product accepted. The conditional acceptance probability is*

$$A_\infty(y) = \exp\big(-1 + \mathbb{E}[\overline{W} \mid y]\big) \in [\mathrm{e}^{-3/2}, \mathrm{e}^{-1/2}]. \tag{10}$$

Repeating independent candidates until acceptance returns the probability measure

$$\widehat{\mu}^x_{t,a}(\,\mathrm{d}y) \coloneqq \frac{\exp(\mathbb{E}[\overline{W} \mid y]) q^x_{t,a}(\,\mathrm{d}y)}{\int \exp(\mathbb{E}[\overline{W} \mid y']) q^x_{t,a}(\,\mathrm{d}y')}. \tag{11}$$

The expectation and denominator in (11) are never evaluated by the algorithm.

D.4 An unconditional exact-posterior one-step bound

The arc and clipped correction above follow the FORS construction of Chen et al. (2026); the proof below gives the coordinate identification used here. We make no low-intrinsic-dimensional assumption and instantiate its intrinsic-dimension bound by the ambient dimension n. Our score is exact, so all approximate-score terms vanish. For probability measures P, Q on a common measurable space with $P \ll Q$, define the conventional Pearson divergence by

$$D_{\chi^2}(P \,\|\, Q) \coloneqq \int \left(\frac{\mathrm{d}P}{\mathrm{d}Q} - 1\right)^2 \mathrm{d}Q = \int \frac{\mathrm{d}P}{\mathrm{d}Q}\,\mathrm{d}P - 1.$$

Theorem D.4 (Exact-posterior one-step bound)**.** *Under the standing finite-second-moment assumption on π, there is a universal constant $C_{\mathsf{FORS}} > 0$ such that, for every $t > 0$, $a > 0$, and $0 < \vartheta < 1/2$, if*

$$\frac{t}{a} \ge C_{\mathsf{FORS}} \left(n \log \frac{1}{\vartheta} + \log^2 \frac{1}{\vartheta}\right), \tag{12}$$

then the corrected law (11) satisfies

$$\mathbb{E}_{Y_t} D_{\chi^2}\left(\mu^{Y_t}_{t,a} \,\middle\|\, \widehat{\mu}^{Y_t}_{t,a}\right) \le \vartheta,$$

and consequently

$$\mathbb{E}_{Y_t} D_{\mathrm{KL}}\left(\mu^{Y_t}_{t,a} \,\middle\|\, \widehat{\mu}^{Y_t}_{t,a}\right) \le \vartheta. \tag{13}$$

Proof. In the variance-exploding coordinates of Chen et al. (2026, Proposition F.3), set $X_+ = Y_t/t$ and $X = Y_{t+a}/(t+a)$. Then their noise variances and forward increment are

$$\sigma_k^2 = (t+a)^{-1}, \qquad \eta_k = \frac{a}{t(t+a)}, \qquad \frac{\sigma_k^2}{\eta_k} = \frac{t}{a}.$$

Indeed, $X_+ - X$ is independent of X and has covariance $\eta_k I_n$. Their effective reverse-proposal variance is $a/(t+a)^2$; multiplying its proposal by $t+a$ recovers $q^x_{t,a} = \mathcal{N}(x + af(t,x), aI_n)$. For the arc, $\gamma_{\mathrm{Chen}} = \gamma/(t+a)$ and $\dot\gamma_{\mathrm{Chen}} = \dot\gamma/(t+a)$; the factor $\sigma_k^{-2} = t+a$ in their correction cancels this derivative scaling. Thus their Gaussian arc and clipped correction match (7) and (9), and their true and corrected transition laws become $\mu^x_{t,a}$ and $\widehat\mu^x_{t,a}$. Apply their proposition with exact posterior means, clipping threshold $1/2$, and intrinsic dimension $d_\star \le n$. Its implicit universal constant is absorbed into C_{FORS}, giving the stated χ^2 bound. The KL bound follows from $D_{\mathrm{KL}}(P \,\|\, Q) \le \log(1 + D_{\chi^2}(P \,\|\, Q)) \le D_{\chi^2}(P \,\|\, Q)$. □

The two terms in (12) come from the ambient-dimensional and clipping-tail terms in the cited one-step estimate.

E First-zero-stopped transition transcripts and path accuracy

Although the sampler ultimately outputs a path of states, recursive rejection must compare the probabilities of whole proposed blocks under the corrected path law and the parallel proposal. The law of a selected next state, $\widehat\mu^x_{t,a}$ in (11), hides a normalizing constant, so its probability cannot be scored directly. Instead, for each transition we record the random choices that produced the selected state: Gaussian candidate increments, Poisson counts, and the auxiliary draws and bits used in correction tests. We call this auxiliary record a *transition transcript*. Its probability is an explicit product of known factors, giving the likelihood ratios needed for recursive rejection. The transcript is bookkeeping for verifying a proposed path, not an additional diffusion state.

The attempt and factor caps make the transcript a finite object, but generating and scoring every slot would make the work depend on those caps. A candidate with zero Poisson factors succeeds automatically, so attempts after the first such candidate cannot affect the selected state. We erase those later records, leaving a transcript with an explicit likelihood and only constantly many attempts and active factors in expectation. The construction below makes these claims precise and shows that erasure does not change the selected-state law.

E.1 The finite ambient transcript

Fix one generic transition from localization time t to $t+a$, entered at state $x \in \mathbb{R}^n$. In the complete path, these local inputs become (t_i, a_i, X_i). The positive integers R and S are safety caps: R is the maximum number of candidate attempts, and S is the maximum number of Poisson factors recorded for one candidate. Their global values are specified later in Definition I.1; throughout this section they are fixed. Set

$$b = f(t,x), \qquad g = x + ab.$$

Thus b is the current drift value returned by the posterior-mean oracle and g is the mean of the Gaussian predictor. The index $r \in \{1,\dots,R\}$ labels candidate attempts, whereas $k \in \{1,\dots,S\}$ labels the possible Poisson-factor slots within one attempt.

The full transcript records the base coordinates needed to reconstruct every candidate and correction factor. Write the record for attempt r and the full transcript as

$$\mathcal{T}_r := \big(\xi_r, C_r, (u_{rk}, G^{(1)}_{rk}, G^{(2)}_{rk}, E_{rk})_{k=1}^S\big), \qquad \mathcal{T} := (\mathcal{T}_r)_{r=1}^R, \tag{14}$$

where

$$\xi_r, G^{(1)}_{rk}, G^{(2)}_{rk} \in \mathbb{R}^n, \quad C_r \in \{0,1,\dots,S,\infty\}, \quad u_{rk} \in [0,1], \quad E_{rk} \in \{0,1\}.$$

Here ξ_r is a candidate increment, C_r is its stored capped count, $(u_{rk}, G^{(1)}_{rk}, G^{(2)}_{rk})$ supplies the random arc used by factor k, and E_{rk} is the corresponding Bernoulli factor. The symbol ∞ marks count overflow and is not a numerical count. The candidate states and correction quantities

$y_r, \widehat{x}_{rk}, z_{rk}, \overline{W}_{rk}, p_{rk}$ introduced below are derived from these coordinates and oracle evaluations; they are not additional transcript coordinates. Conditional on the fixed inputs above, all Poisson, uniform, and Gaussian draws are mutually independent over their displayed indices; only the Bernoulli bits have the conditional laws specified below.

The transcript is generated attempt by attempt as follows.

1. Draw the candidate increment $\xi_r \sim \mathcal{N}(ab, aI_n)$ and set $y_r = x + \xi_r$. Thus y_r is the candidate next state and has the Gaussian-predictor law $q^x_{t,a}$.
2. Draw the uncapped factor count $J_r^{\mathrm{Pois}} \sim \mathrm{Poi}(2)$. Store $C_r = J_r^{\mathrm{Pois}}$ if $J_r^{\mathrm{Pois}} \le S$ and $C_r = \infty$ otherwise; the raw count J_r^{Pois} is not part of the transcript.
3. For every $1 \le k \le S$, draw $u_{rk} \sim \mathrm{Unif}[0,1]$ and independent $G^{(1)}_{rk}, G^{(2)}_{rk} \sim \mathcal{N}(0, I_n)$. Form the spatial auxiliaries
$$\widehat{x}_{rk} = g + \sqrt{a/2}\, G^{(1)}_{rk}, \qquad z_{rk} = \sqrt{a/2}\, G^{(2)}_{rk}.$$
Here u_{rk} is an independent copy of the arc parameter denoted by r in Equation (7); it is distinct from the candidate index r used here.
4. A slot is *active* when $C_r \neq \infty$ and $k \le C_r$. For an active slot, compute $\overline{W}_{rk}$ from (7) and (9) using the complete substitution
$$(r, y, \widehat{x}, z) = (u_{rk}, y_r, \widehat{x}_{rk}, z_{rk}).$$
Define, for every slot,
$$p_{rk} \coloneqq \begin{cases} (1 + \overline{W}_{rk})/2, & C_r \neq \infty \text{ and } k \le C_r, \\ 1/2, & \text{otherwise}, \end{cases} \qquad E_{rk} \sim \mathrm{Ber}(p_{rk}).$$
The bits are conditionally independent given the preceding coordinates. An inactive slot therefore receives a fair *ghost bit*; no correction or oracle evaluation is needed there. These ghost coordinates merely keep every transcript in the same finite product space.

Candidate r is successful when $C_r \neq \infty$ and $E_{rk} = 1$ for all $k \le C_r$. Let ι be the smallest successful candidate index; in particular, $C_r = 0$ gives success by the empty-product convention. If there is no success, set $\iota = 1$. Thus ι is a deterministic function of the transcript, and the selected next state is
$$\Phi_x(\mathcal{T}) \coloneqq x + \xi_\iota. \tag{15}$$
This is the full ambient transcript: before first-zero erasure, it contains all R attempts and all S auxiliary slots even when an early candidate succeeds.

For completeness, we write its normalized density. The tail probability $p_S \coloneqq \mathbb{P}(J_r^{\mathrm{Pois}} > S)$ is independent of r, and the law of the stored count C_r is the mass function
$$\lambda_S(c) \coloneqq \begin{cases} \mathrm{e}^{-2} 2^c / c!, & c \in \{0, \dots, S\}, \\ p_S, & c = \infty. \end{cases}$$
on $\{0, 1, \dots, S, \infty\}$. Let $\mathsf{K}^x_{t,a}$ denote the full-transcript law. Relative to Lebesgue measure on all Euclidean and uniform coordinates and counting measure on counts and bits, its density at $\mathcal{T}$ is
$$\prod_{r=1}^{R} \varphi_{aI_n}(\xi_r - ab) \lambda_S(C_r)$$
$$\times \prod_{r=1}^{R} \prod_{k=1}^{S} \mathbb{1}\{u_{rk} \in [0,1]\} \varphi_{I_n}(G^{(1)}_{rk}) \varphi_{I_n}(G^{(2)}_{rk}) p_{rk}^{E_{rk}} (1 - p_{rk})^{1 - E_{rk}},$$
Here φ_Σ is the centered Gaussian density introduced in Section D, and p_{rk} is the active or ghost probability defined above. Every line in the generative construction is normalized conditional on the preceding lines, so the displayed density integrates to one. We suppress the fixed caps R, S in $\mathsf{K}^x_{t,a}$.

The capped-count masses are positive, the candidate Gaussians are nondegenerate, and all bit probabilities lie in $[1/4, 3/4]$. The frozen-drift proposal introduced in Section G.2 uses the same coordinate space, nondegenerate candidate Gaussians, and fair bits. Hence its full law and the full target transcript law are strictly mutually absolutely continuous. The stopped laws below instead use their common stopped support and a stopped reference measure.

E.2 First-zero erasure

We now shorten the auxiliary record without changing the transition it produces. Stopping at the first zero Poisson count, rather than at a Bernoulli-dependent success, gives a simple bound because the counts are independent of the candidate states and have the same law at every attempt. Recall that $\mathcal{T}_r$ is the record for candidate r in (14), and define the retained indicator

$$\rho_r(\mathcal{T}) \coloneqq \mathbb{1}\{C_s \neq 0 \text{ for every } s < r\}, \qquad 1 \le r \le R. \tag{16}$$

Thus the first zero-count candidate itself is retained. To keep every stopped transcript in the same R-candidate product space, fix a deterministic failure record $\mathcal{T}^{\mathrm{fail}}$: its count is ∞, and all its other coordinates are fixed once and for all. It always fails. Define the measurable erasure map Stop by

$$\big(\mathsf{Stop}(\mathcal{T})\big)_r \coloneqq \begin{cases} \mathcal{T}_r, & \rho_r(\mathcal{T}) = 1, \\ \mathcal{T}^{\mathrm{fail}}, & \rho_r(\mathcal{T}) = 0. \end{cases}$$

If no count is zero, the entire transcript is retained. A zero-count candidate is successful by the empty-product convention, so no candidate after the first zero can affect first-success selection.

Lemma E.1 (First-zero erasure and occupation)**.** *For every full transcript $\mathcal{T}$,*

$$\Phi_x\big(\mathsf{Stop}(\mathcal{T})\big) = \Phi_x(\mathcal{T}). \tag{17}$$

If $K_R \coloneqq \sum_{r=1}^{R} \rho_r(\mathcal{T})$ is the number of retained candidates and A_R is the number of retained active Bernoulli factors, then

$$\mathbb{E}K_R \le \mathrm{e}^2 < 9, \qquad \mathbb{E}A_R \le 2\mathrm{e}^2 < 18. \tag{18}$$

Consequently, the stopped one-step implementation uses $O(1)$ full-vector oracle evaluations in expectation, uniformly in R, S.

Proof. Erasure can only turn candidates after the first zero into failures. If the full transcript has a successful candidate, every candidate up to and including the first success is retained: an earlier zero would itself have been an earlier success. If all candidates fail, no count can be zero and nothing is erased. This proves (17), including the total-failure convention $\iota = 1$.

Let $q = 1 - \mathrm{e}^{-2}$. Candidate r is reached precisely when the preceding $r-1$ independent Poisson counts are nonzero, so

$$\mathbb{E}K_R = \sum_{r=1}^{R} q^{r-1} \le \frac{1}{1-q} = \mathrm{e}^2 < 9.$$

The reach event is independent of the current count, and $\mathbb{E}[J^{\mathrm{Pois}}\mathbb{1}\{J^{\mathrm{Pois}} \le S\}] \le \mathbb{E}J^{\mathrm{Pois}} = 2$. Summing the active factor counts gives $\mathbb{E}A_R \le 2\mathbb{E}K_R \le 2\mathrm{e}^2 < 18$. A stopped step uses a constant number of current-state evaluations and one arc-point evaluation per retained active factor. Its expected query count is therefore $O(1 + \mathbb{E}A_R) = O(1)$. □

Let

$$\mathsf{K}_{t,a}^{x,\mathrm{stop}} \coloneqq (\mathsf{Stop})_{\#}\mathsf{K}_{t,a}^{x}$$

be the resulting fixed-length stopped law. It has an explicit normalized density, but not with respect to the full product reference: its erased suffix consists of deterministic copies of the failure record.

We need a common baseline for the densities of the stopped target and proposal. Construct it as follows. Generate the same capped counts, uniforms, and standard-Gaussian auxiliaries as in the full transcript. For this baseline, draw each candidate increment from $\mathcal{N}(0, aI_n)$ and each bit from $\mathrm{Ber}(1/2)$. Finally, apply the same first-zero erasure. Denote the resulting target-independent law by Λ^{stop}. Because all three laws use the same erasure rule, Λ^{stop}, the stopped target, and the stopped proposal are supported on the same set of transcripts.

The likelihood ratio only needs the retained prefix. Indeed, stopping depends only on the count coordinates, whose distribution is the same under all three laws. Every erased record is also the same deterministic failure record $\mathcal{T}^{\mathrm{fail}}$. The erased suffix therefore contributes no factor to the density, which gives

$$\frac{\mathrm{d}\mathsf{K}_{t,a}^{x,\mathrm{stop}}}{\mathrm{d}\Lambda^{\mathrm{stop}}}(\mathcal{T}) = \prod_{r:\,\rho_r(\mathcal{T})=1} \left[\frac{\varphi_{aI_n}(\xi_r - ab)}{\varphi_{aI_n}(\xi_r)} \prod_{k=1}^{S} 2p_{rk}^{E_{rk}}(1-p_{rk})^{1-E_{rk}} \right]. \tag{19}$$

For an inactive slot, $p_{rk} = 1/2$, so its factor in (19) is one. Replacing the target candidate mean and bit probabilities by their proposal values gives the proposal density relative to the same reference. After conditioning on an incoming history, the same formula applies with the history-measurable means and bit probabilities used below. Those probabilities remain in $[1/4, 3/4]$, so every retained factor is strictly positive. Thus the stopped target and proposal laws are mutually absolutely continuous, and their likelihood ratio reads only the retained prefix.

By (17), define the selected-state kernel equivalently from the stopped or full law:

$$Q^x_{R,S;t,a} \coloneqq (\Phi_x)_{\#}\mathsf{K}^{x,\mathrm{stop}}_{t,a} = (\Phi_x)_{\#}\mathsf{K}^x_{t,a}.$$

Here $(\Phi_x)_{\#}$ denotes pushforward, and we write $q^x_{R,S;t,a}$ for the Lebesgue density of this common selected-state law.

E.3 Effect of the two caps

Lemma E.2 (Capped-transition domination)**.** *Suppose*

$$\mathrm{e}^{3/2}p_S \le \frac{1}{2}, \qquad \exp(-\mathrm{e}^{-2}R) \le \frac{1}{2}. \tag{20}$$

Then, pointwise with respect to Lebesgue measure,

$$q^x_{R,S;t,a}(y) \ge \big(1 - \exp(-\mathrm{e}^{-2}R)\big)\big(1 - \mathrm{e}^{3/2}p_S\big)\,\frac{\mathrm{d}\widehat{\mu}^x_{t,a}}{\mathrm{d}y}(y). \tag{21}$$

Consequently, for every probability measure $\mathsf{P} \ll Q^x_{R,S;t,a}$,

$$D_{\mathrm{KL}}\big(\mathsf{P} \,\big\|\, Q^x_{R,S;t,a}\big) \le D_{\mathrm{KL}}\big(\mathsf{P} \,\big\|\, \widehat{\mu}^x_{t,a}\big) + 2\mathrm{e}^{3/2}p_S + 2\exp(-\mathrm{e}^{-2}R). \tag{22}$$

Proof. Let $A_S(y)$ be the probability that one capped attempt with candidate y succeeds. The infinite and capped experiments agree when the Poisson factor count is at most S, so

$$A_S(y) \ge A_\infty(y) - p_S \ge (1 - \mathrm{e}^{3/2}p_S)A_\infty(y),$$

where the second inequality uses $A_\infty(y) \ge \mathrm{e}^{-3/2}$ from (10). The capped success normalizer is at most the infinite success normalizer. Therefore the candidate law conditional on capped success pointwise dominates $(1 - \mathrm{e}^{3/2}p_S)\widehat{\mu}^x_{t,a}$.

One attempt succeeds whenever its Poisson count is zero, an event of probability e^{-2}. Thus the probability that none of the R attempts succeeds is at most

$$(1 - \mathrm{e}^{-2})^R \le \exp(-\mathrm{e}^{-2}R).$$

Conditional on at least one success, the selected candidate has exactly the one-attempt law conditional on capped success. On total failure the output has some nonnegative residual density; discarding it proves (21). If probability measures Q and $\widehat{\mu}$ satisfy $\mathrm{d}Q/\,\mathrm{d}y \ge c\,\mathrm{d}\widehat{\mu}/\,\mathrm{d}y$, then $D_{\mathrm{KL}}(\mathsf{P} \,\|\, Q) \le D_{\mathrm{KL}}(\mathsf{P} \,\|\, \widehat{\mu}) - \log c$. Apply $-\log(1-z) \le 2z$ for $0 \le z \le 1/2$ to the two factors in (21) to obtain (22). □

Remark E.3. On total failure, candidate one is distributed according to a rejection residual after conditioning on failure; it is generally *not* an unconditioned Gaussian proposal. The proof uses only its nonnegative density and never makes that identification.

E.4 The unguarded path law

We now distinguish the stopped transition-transcript space from the selected-state space. Let $\mathbf{Q}^{\mathrm{tr}}_0$ be the joint law of $X_0 \sim \mathcal{N}(0, t_0 I_n)$ and the stopped transition transcripts generated by, at grid step i, sampling $\mathcal{T}_i \sim \mathsf{K}^{X_i,\mathrm{stop}}_{t_i,a_i}$. The next state is $X_{i+1} = \Phi_{X_i}(\mathcal{T}_i)$. Let $\mathbf{Q}_0$ be the pushforward of $\mathbf{Q}^{\mathrm{tr}}_0$ to the state path $(X_0, \dots, X_M)$. Recall that $\mathbf{P}$ denotes the true grid-state law $(Y_{t_0}, \dots, Y_{t_M})$.

Proof of Proposition B.1. By Lemma C.2, the initial divergence is at most $\zeta/4$. At step i, apply Lemma E.2 with $\mathsf{P} = \mu^x_{t_i,a_i}$ and then average at $x = Y_{t_i}$. The conclusion (13) of Theorem D.4 gives

$$\mathbb{E}_{Y_{t_i}} D_{\mathrm{KL}}\left(\mu^{Y_{t_i}}_{t_i,a_i} \,\middle\|\, Q^{Y_{t_i}}_{R,S;t_i,a_i}\right) \le \theta + 2\mathrm{e}^{3/2} p_S + 2\exp(-\mathrm{e}^{-2}R). \tag{23}$$

The parameter verification in Lemma I.2 gives

$$p_S \le \frac{\zeta}{16\mathrm{e}^{3/2}M}, \qquad \exp(-\mathrm{e}^{-2}R) \le \frac{\zeta}{16M}, \qquad \theta = \frac{\zeta}{4M}.$$

Thus the right side of (23) is at most $\zeta/(2M)$. The chain rule for relative entropy on the selected-state path gives

$$D_{\mathrm{KL}}(\mathbf{P} \,\|\, \mathbf{Q}_0) \le \frac{\zeta}{4} + M\frac{\zeta}{2M} = \frac{3\zeta}{4} \le \zeta. \tag{24}$$

Pinsker's inequality then gives $d_{\mathrm{TV}}(\mathbf{P}, \mathbf{Q}_0) \le \sqrt{\zeta/2}$. The parameter check gives $\zeta \le \varepsilon^2/32$, so these bounds imply $\mathfrak{d}_{\mathrm{path}} \le \varepsilon^2/32$ and $d_{\mathrm{TV}}(\mathbf{P}, \mathbf{Q}_0) \le \varepsilon/8$. This proves Proposition B.1. □

F THE CAUSAL LARGE-DRIFT GUARD

The levelwise rejection proof needs control of the admitted drift array over every subblock in the nested hierarchy under the path actually sampled by the scheduler. The martingale estimate in Lemma C.1 supplies this control only under the true localization path $\mathbf{P}$. Although the unguarded corrected path $\mathbf{Q}_0$ is close to $\mathbf{P}$ in KL divergence, that fact alone cannot transfer the estimate: under $\mathbf{Q}_0$, the drift can fluctuate arbitrarily across scales within a block, so rare off-path histories could dominate the expected dyadic variation. A bounded-support argument could cap such excursions, but the present theorem assumes only a finite second moment.

The causal large-drift guard supplies exactly this missing boundedness. Before admitting each tentative posterior mean, it checks all relevant prefix-measurable dyadic budgets. If the new value would cause an overshoot, the value is discarded, the previous admitted drift is repeated from that step onward, and later FORS correction factors are made neutral. The sampler does not halt. The resulting variation functional is bounded deterministically, so the bounded change-of-measure lemma applies; meanwhile, martingale control shows that the guard triggers rarely on the true path. A common-seed coupling therefore gives both outputs needed later: a bounded drift-variation estimate under the guarded path $\mathbf{Q}_g$ and a small total-variation cost for replacing $\mathbf{Q}_0$ by $\mathbf{Q}_g$.

The guard is not the accept–reject correction. Recursive overlap/residual rejection still corrects the frozen proposal exactly to the guarded FORS target. The guard makes the corresponding rejection probability analyzable, and its rare-trigger coupling relates that guarded target back to the intended diffusion law.

F.1 DYADIC PREFIX BUDGETS

Within coarse interval j, index the N step drifts by $k \in \{0, \dots, N-1\}$. At dyadic level $d \in \{0, \dots, h\}$, partition these indices into 2^d consecutive blocks

$$\mathcal{B}_{d,s} = \left\{ s\frac{N}{2^d}, \dots, (s+1)\frac{N}{2^d} - 1 \right\}, \qquad 0 \le s < 2^d.$$

Thus $\min \mathcal{B}_{d,s} = sN/2^d$ is the first index of the block; we use $v_{\min \mathcal{B}_{d,s}}$ as its anchor drift. Using the interval-restriction notation from Definition I.1, for a local array $v = (v_0, \dots, v_{N-1}) \in (\mathbb{R}^n)^N$ define

$$\mathcal{G}_{j,d}(v) := \sum_{s=0}^{2^d-1} \max_{k \in \mathcal{B}_{d,s}} \left\| v_k - v_{\min \mathcal{B}_{d,s}} \right\|^2.$$

The subscript j identifies the coarse interval to which the array belongs; the formula itself is the same in every interval.

For causality, we also need partial potentials. Given a local prefix ending at $q_{\mathrm{end}} \in \{0, \dots, N-1\}$, set

$$\mathcal{G}^{\le q_{\mathrm{end}}}_{j,d}(v) := \sum_{s:\, \min \mathcal{B}_{d,s} \le q_{\mathrm{end}}} \max_{\substack{k \in \mathcal{B}_{d,s} \\ k \le q_{\mathrm{end}}}} \left\| v_k - v_{\min \mathcal{B}_{d,s}} \right\|^2. \tag{25}$$

This quantity is measurable from the prefix and nondecreasing in q_{end}.

Let

$$\alpha \coloneqq \varepsilon/4, \qquad K \coloneqq J(h+1), \qquad w_j \coloneqq \frac{M_2^2}{1+\tau_j M_2^2/n}, \qquad \Gamma_j \coloneqq \frac{4Kw_j}{\alpha} = \frac{16Kw_j}{\varepsilon}. \tag{26}$$

F.2 Stopping rule, cache, and conditional guarded law

Causal stopping rule. The guarded chain carries a frozen flag and a currently admitted drift. If the flag is already set, it reuses the admitted drift and sets every correction $\overline{W}$ to zero, so all factor bits are fair. Otherwise, at a step $i \in \mathcal{I}_j$ it tentatively appends $f(t_i, x_i)$ to the current coarse-interval drift prefix. If every partial budget in (25) is at most Γ_j, the value is admitted. At the first tentative value that makes at least one budget exceed Γ_j, the value is discarded, the preceding admitted drift is used for the current step, and the flag is set permanently. (At the first drift index of any coarse interval no violation is possible, because every relevant difference is zero.)

This defines a deterministic causal map $\mathcal{S} : (\mathbb{R}^n)^M \to (\mathbb{R}^n)^M$ from any complete raw drift array to its admitted array. It also defines a stopping index ι_g: before ι_g, admitted and raw drifts agree; at and after ι_g, the admitted array repeats the value immediately preceding ι_g. If there is no violation, $\iota_g = \infty$.

Let $\mathbf{Q}_g^{\mathrm{tr}}$ be the joint law of the initial state $X_0^g \sim \mathcal{N}(0, t_0 I_n)$ and the guarded stopped transition transcripts $\mathsf{Z} = (\mathsf{Z}_i)_{i=0}^{M-1}$. Let $\mathbf{Q}_g$ be the selected-state pushforward of this joint law. Denote its admitted drifts by b_i and set

$$I_i \coloneqq \mathbb{1}\{i < \iota_g\}.$$

Thus $I_i = 0$ at the triggering step itself, and the correction there is already fair.

Incoming history and deterministic cache. For a contiguous block B of global step indices, set

$$\mathcal{H}_{<B} \coloneqq \left(X_0^g, (\mathsf{Z}_i)_{0 \le i < \min B}\right),$$

where the transcript prefix is empty when $\min B = 0$. Write $\mathfrak{h}_{<B}$ for a deterministic realization. Including X_0^g is essential at the first block: the empty transcript prefix alone does not determine its incoming state. For a leaf $B = \{i\}$, we use the abbreviations $\mathcal{H}_{<i}$ and $\mathfrak{h}_{<i}$.

The executable recursion also carries a guard cache $\mathfrak{c}_{<B}$. Its base fields are the decoded state at the start of B, the permanent frozen flag, the last admitted drift when one exists, and the local position in the coarse interval. For every dyadic level d, it also contains the anchor drift of the current block $\mathcal{B}_{d,s}$ once that block has begun, the running maximum of its squared deviations, and the sum of the maxima of the completed blocks at that level. Their sum is the current partial potential in (25). When a new drift arrives, the known local position determines whether to open a new dyadic block or update its running maximum, so these fields are sufficient to perform every admission test. The cache is therefore previously computed deterministic data, not additional randomness. We suppress it from probability-law superscripts but pass it explicitly to executable routines, avoiding any replay of earlier oracle calls. Its threading through the recursive sampler is described in Section H.

Conditional guarded kernel. Condition on the full history entering step i. The state x_i, admitted drift b_i, and indicator I_i are then deterministic. Give the transcript coordinates in (14) the additional step index i. The conditional step law is obtained by first-zero erasure of the corresponding full guarded transcript. Relative to the stopped reference in Section E.2, it has the retained-prefix density (19) with candidate-increment mean $a_i b_i$ and arc center $x_i + a_i b_i$. For every retained candidate r, its conditional factor-bit probabilities are

$$\mathbb{P}(E_{irk} = 1 \mid \mathcal{H}_{<i},\ \text{preceding coordinates}) = \begin{cases} (1 + I_i \overline{W}_{irk}^{(b_i)})/2, & C_{ir} \neq \infty \text{ and } k \le C_{ir}, \\ 1/2, & \text{otherwise.} \end{cases} \tag{27}$$

Here $\overline{W}_{irk}^{(b_i)}$ is obtained from (7) and (9) by using (t_i, a_i, x_i, b_i) and the stored arc variables. Here $r \in \{1, \dots, R\}$ indexes candidate attempts and $k \in \{1, \dots, S\}$ indexes their factors. Candidates outside the retained prefix in (16) are fixed failure records, not collections of fair ghost coordinates.

Because the stopped reference applies the same erasure map, the displayed formula defines a normalized guarded step kernel without assigning a product density to the erased suffix.

For a deterministic incoming history $\mathfrak{h}_{<B}$, let $\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}$ denote the resulting conditional block law. Its normalized density with respect to the product of the stepwise stopped references is the product of the step densities above, with the decoded state and deterministic guard updated after each step.

F.3 Variation under the true path

Lemma F.1 (True dyadic variation)**.** *For every coarse interval j and every dyadic level d,*

$$\mathbb{E}_{\mathbf{P}}\mathcal{G}_{j,d}\big((\widehat{x}_{\star,t_{j,k}})_{k=0}^{N-1}\big) \le 4\big(m(\tau_j) - m(\tau_{j+1})\big) \le 4w_j. \tag{28}$$

Consequently, when the guard is applied hypothetically to the true path,

$$\mathbb{P}_{\mathbf{P}}(\iota_g < \infty) \le \alpha. \tag{29}$$

Proof. Fix one block $\mathcal{B}_{d,s}$ and let s_- and s_+ be the localization times at its left and right boundaries. Doob's L^2 maximal inequality for the $\mathbb{R}^n$-valued martingale $\widehat{x}_{\star,t} - \widehat{x}_{\star,s_-}$ gives

$$\mathbb{E}\max_{k\in\mathcal{B}_{d,s}} \big\|\widehat{x}_{\star,t_{j,k}} - \widehat{x}_{\star,s_-}\big\|^2 \le 4\mathbb{E}\big\|\widehat{x}_{\star,s_+} - \widehat{x}_{\star,s_-}\big\|^2 = 4(m(s_-) - m(s_+)),$$

where using the boundary after the last drift index only enlarges the maximum-time interval. Summing over the blocks telescopes the last differences, proving the first inequality in (28). The second follows from $m(\tau_j) \le w_j$ in (5).

There are exactly $K = J(h+1)$ interval-level budgets. By Markov's inequality, the probability that a fixed one exceeds $\Gamma_j = 4Kw_j/\alpha$ is at most α/K. A union bound proves (29). The same argument applies to partial budgets because each is bounded by its completed budget. □

Remark F.2. Only the standard L^2 maximal inequality is used; bounded support is not needed. See Acciaio et al. (2013) for a deterministic and probabilistic perspective on Doob-type maximal inequalities.

F.4 Deterministic contraction and coupling

Lemma F.3 (Pre-overshoot contraction)**.** *For every deterministic global raw drift array $v \in (\mathbb{R}^n)^M$, every interval j, and every level d,*

$$\mathcal{G}_{j,d}\big((\mathcal{S}v)^{(j)}\big) \le \mathcal{G}_{j,d}\big(v^{(j)}\big), \qquad \mathcal{G}_{j,d}\big((\mathcal{S}v)^{(j)}\big) \le \Gamma_j.$$

Proof. Before the stopping index, the two arrays coincide. A block strictly after the stopping index is constant under $\mathcal{S}$ and contributes zero. In a block crossing the stopping index, if the stop occurs after the block's first coordinate, every post-stop value repeats a raw prefix value from that block; if it occurs at the first coordinate, the entire guarded block is constant. In either case the block maximum cannot increase. This proves the first inequality. Immediately before the tentative overshoot, every partial budget is at most Γ_j; replacing the overshooting value and every later value as above cannot increase any partial budget. This proves the second inequality. □

Couple $\mathbf{Q}_0^{\mathrm{tr}}$ and $\mathbf{Q}_g^{\mathrm{tr}}$ using the same initial Gaussian, candidate standard normals, Poisson counts, arc variables, and bit uniforms, followed by the same first-zero erasure and failure-record replacement. Candidate locations are formed from the appropriate drift, and bits from the appropriate inverse Bernoulli transform. The selected states and all retained transcript coordinates agree strictly before the first guard violation. Although the state paths may differ afterward, the admitted drift array satisfies the pathwise identity

$$(b_i)_{i=0}^{M-1} = \mathcal{S}\big((f(t_i, X_i^0))_{i=0}^{M-1}\big), \tag{30}$$

where (X_i^0) is the unguarded selected-state path in this coupling. The identity holds because $\mathcal{S}$ ignores all raw values after its first overshoot.

Proof of Proposition B.2. Recall that A_g is the event that the deterministic guard map triggers. Equation (29) gives $\mathbb{P}_{\mathbf{P}}(A_g) \le \alpha$. Pinsker's inequality then implies

$$\mathbb{P}_{\mathbf{Q}_0}(A_g) \le \mathbb{P}_{\mathbf{P}}(A_g) + d_{\mathrm{TV}}(\mathbf{P}, \mathbf{Q}_0) \le \alpha + \sqrt{\mathfrak{d}_{\mathrm{path}}/2}.$$

In the common-seed coupling, the guarded and unguarded state paths agree on A_g^c. Hence

$$d_{\mathrm{TV}}(\mathbf{Q}_g, \mathbf{Q}_0) \le \mathbb{P}_{\mathbf{Q}_0}(A_g).$$

A second application of Pinsker's inequality together with the triangle inequality gives

$$d_{\mathrm{TV}}(\mathrm{Law}(X_M^g), \mathrm{Law}(Y_T)) \le d_{\mathrm{TV}}(\mathbf{Q}_g, \mathbf{Q}_0) + d_{\mathrm{TV}}(\mathbf{Q}_0, \mathbf{P}) \le \alpha + 2\sqrt{\mathfrak{d}_{\mathrm{path}}/2}.$$

By Proposition B.1, the right side is at most $\varepsilon/2$, and in particular at most ε. This proves Proposition B.2. □

Proof of Proposition B.3. Fix j, d. For $x_{0:M} \in (\mathbb{R}^n)^{M+1}$, define the selected-state path functional

$$F_{j,d}(x_{0:M}) := \mathcal{G}_{j,d}\Big(\big[\mathcal{S}\big((f(t_i, x_i))_{i=0}^{M-1}\big)\big]^{(j)} \Big).$$

By Lemma F.3, $0 \le F_{j,d} \le \Gamma_j$ and, under the true law, $F_{j,d}$ is bounded by the unguarded true-path potential. Therefore Lemmas C.4 and F.1 and $\mathfrak{d}_{\mathrm{path}} = D_{\mathrm{KL}}(\mathbf{P} \,\|\, \mathbf{Q}_0)$ give

$$\mathbb{E}_{\mathbf{Q}_0} F_{j,d} \le 2\mathbb{E}_{\mathbf{P}} F_{j,d} + 2\Gamma_j \mathfrak{d}_{\mathrm{path}} \le 8w_j + 2\Gamma_j \mathfrak{d}_{\mathrm{path}}.$$

By (24) and Lemma I.2, $\mathfrak{d}_{\mathrm{path}} \le \varepsilon/(4K)$, so the last term is at most $8w_j$ by (26). The coupling identity (30) now yields

$$\mathbb{E}_{\mathbf{Q}_g} \mathcal{G}_{j,d}\big(b^{(j)}\big) = \mathbb{E}_{\mathbf{Q}_0} F_{j,d} \le 16 w_j, \tag{31}$$

Since the interval length is $H_j \le \tau_j$ and $w_j \le n/\tau_j$, multiplying (31) by the interval length gives the $O(n)$ statement in Proposition B.3. This proves Proposition B.3. □

G Correction energy and the rejection budget

The guard controls the Gaussian mean shifts in a frozen block proposal. It does not by itself control the transcript's Bernoulli factors. A per-step bound $\overline{W}^2 \le 1/4$ would accumulate to order N before Pinsker and would destroy the desired depth bound. The next lemma supplies the missing factor of the step size.

G.1 Correction energy

For fixed $t, a > 0$ and $x \in \mathbb{R}^n$, set $b = f(t, x)$. In the expectation below, draw $y \sim q_{t,a}^x$ and, independently, draw $r \sim \mathrm{Unif}[0, 1]$, $\widehat{x} \sim \mathcal{N}(x + ab, (a/2)I_n)$, and $z \sim \mathcal{N}(0, (a/2)I_n)$, as in the Gaussian-arc construction. Define

$$\kappa_{t,a}(x) := \mathbb{E}\overline{W}^2, \tag{32}$$

where $\overline{W}$ is the clipped correction in (9).

Lemma G.1 (Correction energy)**.** *For every $t, a > 0$ and $x \in \mathbb{R}^n$,*

$$0 \le \kappa_{t,a}(x) \le \frac{1}{4}, \qquad \kappa_{t,a}(x) \le \frac{25}{4} a \Delta_{t,a}(x). \tag{33}$$

Proof. The first bound is clipping. Under the joint law in (32), abbreviate $\gamma = \gamma_r(y)$ and $\dot{\gamma} = \dot{\gamma}_r(y)$. Then Lemma D.1 says that this pair has independent marginals

$$q_{t,a}^x \quad \text{and} \quad \mathcal{N}(0, c_{\mathrm{arc}} a I_n)\,.$$

Let Q_γ denote this product law. Let P_γ be the product law with the same second marginal but first marginal $\mu_{t,a}^x$. Then

$$D_{\mathrm{KL}}(P_\gamma \,\|\, Q_\gamma) = D_{\mathrm{KL}}\big(\mu_{t,a}^x \,\big\|\, q_{t,a}^x\big) \le \frac{a}{2} \Delta_{t,a}(x) \tag{34}$$

by Lemma C.3. For the observable

$$F(\gamma,\dot\gamma)=\Big[\mathrm{clip}_{[-1/2,1/2]}(\langle\dot\gamma, f(t+a,\gamma)-b\rangle)\Big]^2,$$

we have $0\le F\le 1/4$ and, under P_γ,

$$\begin{aligned}\mathbb{E}_{P_\gamma}F &\le \mathbb{E}_{P_\gamma}\left|\langle\dot\gamma, f(t+a,\gamma)-b\rangle\right|^2\\ &= c_{\mathrm{arc}}a\Delta_{t,a}(x).\end{aligned}$$

Apply Lemma C.4 with $P=P_\gamma$, $Q=Q_\gamma$, and $C=1/4$. Using (34) and $c_{\mathrm{arc}}<3$ gives

$$\kappa_{t,a}(x)\le 2c_{\mathrm{arc}}a\Delta_{t,a}(x)+\frac{1}{2}\cdot\frac{a}{2}\Delta_{t,a}(x)\le\frac{25}{4}a\Delta_{t,a}(x).$$

□

Recall the fine-step index set $\mathcal{I}_j$ from Definition I.1. For $x_{0:M}\in(\mathbb{R}^n)^{M+1}$, define the path functional

$$A_j(x_{0:M}):=\sum_{i\in\mathcal{I}_j}\kappa_{t_i,a_j}(x_i).$$

Since there are N summands, $0\le A_j\le N/4$.

Lemma G.2 (Accumulated correction)**.** *For every coarse interval j,*

$$\mathbb{E}_{\mathbf{P}}A_j\le\frac{25H_jw_j}{4N},\tag{35}$$

$$\mathbb{E}_{\mathbf{Q}_0}A_j\le\frac{25H_jw_j}{2N}+\frac{N}{2}\mathfrak{d}_{\mathrm{path}},\tag{36}$$

$$\mathbb{E}_{\mathbf{Q}_g^{\mathrm{tr}}}\sum_{i\in\mathcal{I}_j}I_i\kappa_{t_i,a_j}(X_i^g)\le\frac{25H_jw_j}{2N}+\frac{N}{2}\mathfrak{d}_{\mathrm{path}}.\tag{37}$$

Proof. Average (33) at $x=Y_{t_i}$. By the martingale identity (4),

$$\mathbb{E}\Delta_{t_i,a_j}(Y_{t_i})=m(t_i)-m(t_{i+1}).$$

All fine steps in the interval have size $a_j=H_j/N$, so summing telescopes:

$$\mathbb{E}_{\mathbf{P}}A_j\le\frac{25H_j}{4N}\big(m(\tau_j)-m(\tau_{j+1})\big)\le\frac{25H_jw_j}{4N}.$$

This proves (35). Apply Lemma C.4 to the bounded path functional $A_j\le N/4$ and the pair $(\mathbf{P},\mathbf{Q}_0)$ to obtain (36). Finally, in the common-seed coupling of Section F, guarded and unguarded states agree at every active step; after the stopping index the left side of (37) has no contribution. It is therefore bounded pathwise by the unguarded A_j, which proves (37). □

G.2 The speculative FORS trajectory

Fix a contiguous non-leaf block $B=\{s,s+1,\dots,s+m-1\}$ of m single-step transitions in one coarse interval. Condition on a realization $\mathfrak{h}_{<B}$ of the full incoming history $\mathcal{H}_{<B}$ from Section F, and let $\mathfrak{c}_{<B}$ be its carried guard cache. Together these fix the incoming selected state x_s, guard flag, last admitted drift, and partial guard budgets. The following is the detailed version of the informal Algorithm 2. It first calls the pure Euler primitive and then augments the resulting increments into the stopped proposal $\nu_B^{\mathfrak{h}_{<B}}$ on the same fixed-length stopped transcript space as the guarded conditional target $\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}$.

Algorithm 4: PARALLELFORSBLOCK: complete implementation of Algorithm 2.

Input: A non-leaf block $B = \{s, \dots, s+m-1\}$, incoming history $\mathfrak{h}_{<B}$, guard cache $\mathfrak{c}_{<B}$, and parameters a_j, R, S
Access: The exact posterior-mean oracle and deterministic guard
Output: A stopped proposal transcript, its speculative states, both likelihood ratios, and a provisional outgoing cache

On a provisional copy of $\mathfrak{c}_{<B}$, process the block head: if the guard is active, query $f(t_s, x_s)$ and apply its admission rule; if frozen, reuse its last admitted drift. Obtain $(b_s, I_s, \mathfrak{c}_s^+)$ and set $b_B^\circ \leftarrow b_s$;
Call Algorithm 3 with (B, x_s, b_B°, a_j), obtaining $(\Delta_i)_{i\in B}$ and $(x_i^{\mathrm{sp}})_{i=s}^{s+m}$;
foreach $i \in B$ in parallel **do**
  Independently of the increments returned by Algorithm 3, draw $R-1$ fresh fillers from $\mathcal{N}(a_j b_B^\circ, a_j I_n)$ and, for every $(r,k) \in [R] \times [S]$, independently draw $u_{irk} \sim \mathrm{Unif}[0,1]$ and $G_{irk}^{(1)}, G_{irk}^{(2)} \sim \mathcal{N}(0, I_n)$;
  For each $r \in [R]$, draw $J_{ir} \sim \mathrm{Poi}(2)$, set $C_{ir} = J_{ir}$ when $J_{ir} \le S$ and $C_{ir} = \infty$ otherwise, and draw $(E_{irk})_{k=1}^S$ independently from $\mathrm{Ber}(1/2)$;
  Let ι_i be the first r with $C_{ir} \neq \infty$ and $E_{irk} = 1$ for all $k \le C_{ir}$, or set $\iota_i = 1$ if there is no such r. Put Δ_i in candidate slot ι_i and assign the fillers to the other slots in increasing slot order;
  Replace every attempt strictly after the first zero count by the fixed failure record, obtaining the stopped step transcript Z_i;
Set $\mathsf{Z}_B \leftarrow (\mathsf{Z}_i)_{i\in B}$ and call Algorithm 5 on it, reusing the cached head update $(b_s, I_s, \mathfrak{c}_s^+)$; obtain L_B, L_B^{-1}, and $\mathfrak{c}_{\le B}$;
return Z_B, $(\Delta_i)_{i\in B}$, $(x_i^{\mathrm{sp}})_{i=s}^{s+m}$, L_B, L_B^{-1}, and the provisional $\mathfrak{c}_{\le B}$;

Lemma G.3 (Selected-coordinate disintegration)**.** *The augmentation step in Algorithm 4 produces exactly the original stopped frozen-proposal law $\nu_B^{\mathfrak{h}_{<B}}$, and its decoded increments and states are precisely the output of Algorithm 3.*

Proof. Fix the count and fair-bit variables for one step, hence its selected index r. The frozen candidate density is $\prod_{h=1}^R \varphi_{a_j I_n}(\xi_h - a_j b_B^\circ)$. Writing $\delta = \xi_r$ factors this density as $\varphi_{a_j I_n}(\delta - a_j b_B^\circ) \prod_{h\neq r} \varphi_{a_j I_n}(\xi_h - a_j b_B^\circ)$, independently of r. Thus one may draw δ first, place it in the subsequently revealed selected slot, and fill every other slot independently. The uniforms and bridge Gaussians are independent proposal variables and may also be drawn before the counts and bits. Finally, stopping depends only on the counts and never erases the selected slot by Lemma E.1. The fixed failure-record replacement therefore gives exactly the original stopped proposal law. The argument is independent across steps. □

This reverse ordering is used only for the frozen proposal. For a target-law fallback or leaf sample, the corrected bits depend on the candidates, bridge variables, and oracle values, so the original forward sampling order is kept.

Write $p_B^{\mathfrak{h}_{<B}}$ and $q_B^{\mathfrak{h}_{<B}}$ for the normalized densities of the guarded target $\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}$ and proposal $\nu_B^{\mathfrak{h}_{<B}}$, respectively, relative to the common stopped reference. The procedure below evaluates both densities.

Proposition G.4 (Conditional block likelihood ratio)**.** *Fix an admissible incoming history $\mathfrak{h}_{<B}$ with its cache, and let z_B lie in the common stopped support. Decode its states and replay the guard, obtaining $(b_i, I_i)_{i\in B}$, and set $b_B^\circ = b_s$. For $i \in B$ and $r \in [R]$, define*

$$\rho_{ir} := \mathbb{1}\{C_{i\ell} \neq 0 \text{ for every } \ell < r\}, \qquad \mathcal{A}_{ir} := \{k \in [S] : C_{ir} \neq \infty,\ k \le C_{ir}\}.$$

For $k \in \mathcal{A}_{ir}$ put

$$\omega_{irk} := \begin{cases} \overline{W}_{irk}^{(b_i)}, & I_i = 1, \\ 0, & I_i = 0. \end{cases}$$

Define the one-candidate Gaussian and active-bit likelihood ratios by

$$G_{ir} := \frac{\varphi_{a_j I_n}(\xi_{ir} - a_j b_i)}{\varphi_{a_j I_n}(\xi_{ir} - a_j b_B^\circ)},$$

$$B_{irk} := 2\left(\frac{1+\omega_{irk}}{2}\right)^{E_{irk}} \left(\frac{1-\omega_{irk}}{2}\right)^{1-E_{irk}}.$$

Then

$$\frac{\mathrm{d}\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}}{\mathrm{d}\nu_B^{\mathfrak{h}_{<B}}}(\mathsf{z}_B) = \frac{p_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)}{q_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)} = \prod_{i\in B}\prod_{r:\,\rho_{ir}=1}\left[G_{ir}\prod_{k\in\mathcal{A}_{ir}}B_{irk}\right]. \tag{38}$$

Equivalently, the Gaussian and bit factors in (38) are respectively

$$G_{ir} = \exp\Big\{\langle \xi_{ir}, b_i - b_B^\circ\rangle - \frac{a_j}{2}\left(\|b_i\|^2 - \|b_B^\circ\|^2\right)\Big\}, \qquad B_{irk} = 1 + (2E_{irk}-1)\omega_{irk}.$$

All factors are strictly positive; hence the reverse derivative is their reciprocal.

Proof. Apply (19) conditionally at each step. On a retained attempt, the target and proposal candidate means are $a_j b_i$ and $a_j b_B^\circ$, while on an active factor slot their bit probabilities are $(1+\omega_{irk})/2$ and $1/2$. The capped-count, uniform, and auxiliary standard-Gaussian laws agree; ghost-bit factors are fair under both laws; and the suffix after the first zero consists of the same fixed failure records. Dividing the conditional one-step densities and multiplying them along the decoded block gives (38). The two equivalent factors follow by expanding the Gaussian square and using $E_{irk} \in \{0,1\}$. Finally, $|\omega_{irk}| \le 1/2$, so the ratio is finite and strictly positive. □

The scorer is a separate callable operation. It handles both a transcript from Algorithm 4, for the initial overlap test, and a target-generated fallback transcript, for the residual test. It never resamples its input.

Algorithm 5: Scoring a supplied speculative-FORS block transcript

Input: A non-leaf block $B = \{s, s+1, \dots, s+m-1\}$, its incoming history $\mathfrak{h}_{<B}$, guard cache $\mathfrak{c}_{<B}$, and a stopped transcript z_B in the common support; optionally, the cached head update from Algorithm 4

Access: The exact posterior-mean oracle, the deterministic guard, and the step parameters a_j, R, S

Output: The two reciprocal conditional likelihood ratios at z_B and the outgoing guard cache $\mathfrak{c}_{\le B}$

Decode the selected increments and all states $x_s, \dots, x_{s+m}$ from $(\mathfrak{h}_{<B}, \mathsf{z}_B)$ using (15);

Oracle stage 1: if the cached head update is present, reuse $(b_s, I_s, \mathfrak{c}_s^+)$, query $f(t_i, x_i)$ for every $s < i \le s+m-1$ in one batch, and continue the guard from $\mathfrak{c}_s^+$; otherwise query all $f(t_i, x_i)$, $i \in B$, in one batch and start from $\mathfrak{c}_{<B}$. Obtain every admitted drift b_i, active indicator I_i, and the outgoing cache $\mathfrak{c}_{\le B}$, and set $b_B^\circ \leftarrow b_s$;

For every retained active slot with $I_i = 1$, reconstruct $y_{ir}, \widehat{x}_{irk}, z_{irk}, \gamma_{irk}, \dot\gamma_{irk}$ from the stored variables, with

$$y_{ir} = x_i + \xi_{ir}, \qquad \widehat{x}_{irk} = x_i + a_j b_i + \sqrt{a_j/2}\, G_{irk}^{(1)}, \qquad z_{irk} = \sqrt{a_j/2}\, G_{irk}^{(2)},$$

and with $(\gamma_{irk}, \dot\gamma_{irk})$ given by (7) and (8);

Oracle stage 2: query all resulting values $f(t_i + a_j, \gamma_{irk})$ in one batch, and set the correction to zero in slots whose $I_i = 0$;

Compute the clipped corrections and the two normalized densities $p_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)$ and $q_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)$ from the retained one-step factors, using the admitted b_i for the target Gaussian factors, the frozen b_B° for the proposal Gaussian factors, and (38) for their ratio;

return $p_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)/q_B^{\mathfrak{h}_{<B}}(\mathsf{z}_B)$, its reciprocal, and $\mathfrak{c}_{\le B}$;

By Lemma G.3, the selected increments of the stopped proposal are the independent $\mathcal{N}(a_j b_B^\circ, a_j I_n)$ increments drawn by Algorithm 3; their prefix scan recovers all $m+1$ speculative states. The Euler primitive itself uses no oracle call. Algorithm 4 adds the head-guard query in at most one oracle stage, and Algorithm 5 adds the current-state and arc-point batches. Thus proposal generation and the overlap test use at most three oracle stages. There is at most one head query, at most one further current-state query per fine step, and one query per retained active factor. By (18), the conditional expected query count is at most

$$1 + 19|B|. \tag{39}$$

On a supplied transcript, ScoreFORSBlock uses one current-state query per fine step and one query per retained active factor. Under either the target or proposal law the count coordinates have the same stopped capped-Poisson distribution, so (18) bounds its conditional expected query count by $19|B|$, uniformly in R, S. It uses two adaptive oracle stages pointwise. In particular, this is the residual scorer required for a fresh target-distributed fallback candidate.

The word "depth" in these statements means adaptive *oracle* depth. All indexed random draws are simultaneous. A coordinatewise bounded-fan-in prefix scan has $O(\log m)$ arithmetic depth;

selector and density reductions, guard evaluation, and scalar products introduce further reductions over R, S, the guard budgets, and the ambient dimension. In a unit-cost vector-operation model the final polylogarithmic caps give $\widetilde{O}(m)$ local work, but we do not state a scalar PRAM bound. These local arithmetic costs are outside the oracle-round theorem; in particular, we make no claim of constant total PRAM depth.

Proof of Proposition B.4. The disintegration lemma proves that Algorithm 4 generates the stated proposal law and that its decoded path is the output of the parallel Euler primitive. The stopped target and proposal densities are normalized and mutually absolutely continuous by Section E.2, and Proposition G.4 gives their Radon–Nikodym derivative explicitly. The scorer decodes its states before querying the oracle, after which the current-state and arc-point calls form exactly the two displayed batches. The occupation calculation proves both expected-cost bounds. The leaf claim follows from Definition G.5 below. □

Definition G.5 (Exact guarded leaf sampler)**.** For a leaf $B = \{i\}$ and an incoming full-history realization $\mathfrak{h}_{<i}$ with guard cache $\mathfrak{c}_{<i}$, ExactGuardedStep$(i, \mathfrak{h}_{<i}, \mathfrak{c}_{<i})$ generates one guarded stopped transition transcript as follows. It queries the current raw drift and applies the deterministic admission rule, samples the finite pre-bit transcript, erases after the first zero count, queries the retained active arc points in one batch, samples the retained factor bits from their conditional probabilities, and returns the outgoing guard cache. If $\mathsf{Z}_i^{\mathrm{leaf}}$ denotes its transcript output, then its law is the guarded conditional target, which we also use as the leaf proposal:

$$\mathrm{Law}(\mathsf{Z}_i^{\mathrm{leaf}}) = \mathbf{Q}_{g,\{i\}}^{\mathfrak{h}_{<i}} = \nu_{\{i\}}^{\mathfrak{h}_{<i}}.$$

Thus the sampler uses $O(1)$ expected queries and $O(1)$ pointwise rounds, uniformly in R, S, and a leaf never rejects.

All of these statements use the exact-real oracle model: Gaussian, Poisson, uniform, and Bernoulli variables are sampled exactly; returned reals and density ratios are manipulated exactly; and an unbounded number of queries may be issued within one oracle round.

G.3 Conditional transcript KL

Lemma G.6 (Block-transcript divergence)**.** *For every non-leaf block B and every incoming full-history realization $\mathfrak{h}_{<B}$,*

$$D_{\mathrm{KL}}\left(\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}} \,\middle\|\, \nu_B^{\mathfrak{h}_{<B}}\right) \le 9\mathbb{E}_{\mathbf{Q}_{g,B}^{\mathfrak{h}_{<B}}} \sum_{i\in B}\left[\frac{a_j}{2}\|b_i - b_B^\circ\|^2 + 2I_i\kappa_{t_i,a_j}(X_i^g)\right]. \tag{40}$$

We call the first summand in brackets the Girsanov-type error *and the second the* FORS correction-energy term.

Proof. First fix one step and condition on its incoming history. Let D_m be the KL divergence between the target and proposal stopped transcripts with attempt cap m, and let d_1 be the corresponding one-candidate divergence. Both laws have the same capped-$\mathrm{Poi}(2)$ count marginal. If the head count is zero, its probability is e^{-2} under both laws and the remaining transcript consists of the same deterministic failure records. On its complement the stopped tail has divergence D_m. The chain rule therefore gives, with $q = 1 - \mathrm{e}^{-2}$,

$$D_0 = 0, \qquad D_{m+1} = d_1 + qD_m, \qquad D_R \le \mathrm{e}^2 d_1 \le 9d_1. \tag{41}$$

For one candidate at step i, the target and proposal Gaussian increments have common covariance a_jI_n and means a_jb_i and $a_jb_B^\circ$. Their KL divergence is $a_j\|b_i - b_B^\circ\|^2/2$. The capped-count, uniform, and standard-Gaussian factors agree exactly and contribute zero.

For an active bit with clipped correction w, the elementary bound by chi-squared divergence gives

$$D_{\mathrm{KL}}(\mathrm{Ber}((1+w)/2) \,\|\, \mathrm{Ber}(1/2)) \le w^2, \qquad |w| \le 1/2.$$

If $J^{\mathrm{Pois}} \sim \mathrm{Poi}(2)$, then the expected number of active factors is $\mathbb{E}[J^{\mathrm{Pois}}\mathbb{1}\{J^{\mathrm{Pois}} \le S\}] \le \mathbb{E}J^{\mathrm{Pois}} = 2$. Conditional independence of the factor copies therefore bounds its expected bit contribution by $2\kappa_{t_i,a_j}(X_i^g)$. If $I_i = 0$, every target bit is fair and the contribution is zero. Thus

$$d_1 \le \frac{a_j}{2}\|b_i - b_B^\circ\|^2 + 2I_i\kappa_{t_i,a_j}(X_i^g).$$

Apply (41) conditionally at every step and then the chain rule along the block. Averaging the history-dependent terms proves (40). □

G.4 Aggregation over a tree level

Proof of Proposition B.5. Fix interval j and an internal tree level $d < h$. There are $n_{d,\mathrm{blk}} := 2^d$ blocks, each containing $N/n_{d,\mathrm{blk}}$ steps and spanning time $H_j/n_{d,\mathrm{blk}}$. Under $\mathbf{Q}_g^{\mathrm{tr}}$, abbreviate $b_B^\circ = b_B^\circ(\mathcal{H}_{<B}) = b_{\min B}$. For each block, average (40) over its incoming history under the global guarded transcript law, then sum over all blocks at the level. Since b_B° is the admitted drift at the block's first index,

$$\sum_{B \text{ at level } d} \sum_{i \in B} a_j \|b_i - b_B^\circ\|^2 \le \frac{H_j}{n_{d,\mathrm{blk}}} \mathcal{G}_{j,d}\big(b^{(j)}\big).$$

Using (31) and (37) gives

$$\sum_{B \text{ at level } d} \mathbb{E}_{\mathbf{Q}_g^{\mathrm{tr}}} D_{\mathrm{KL}}\Big(\mathbf{Q}_{g,B}^{\mathcal{H}_{<B}} \,\Big\|\, \nu_B^{\mathcal{H}_{<B}}\Big) \le \frac{72 H_j w_j}{n_{d,\mathrm{blk}}} + \frac{225 H_j w_j}{N} + 9N\mathfrak{d}_{\mathrm{path}}. \tag{42}$$

Pinsker's inequality, Jensen's inequality, and Cauchy–Schwarz now imply

$$\begin{aligned}
\overline{\mathfrak{R}}_{j,d} &:= \mathbb{E}_{\mathbf{Q}_g^{\mathrm{tr}}} \sum_{B \text{ at level } d} d_{\mathrm{TV}}\Big(\mathbf{Q}_{g,B}^{\mathcal{H}_{<B}}, \nu_B^{\mathcal{H}_{<B}}\Big) \\
&\le \left(\frac{n_{d,\mathrm{blk}}}{2} \sum_{B \text{ at level } d} \mathbb{E}_{\mathbf{Q}_g^{\mathrm{tr}}} D_{\mathrm{KL}}\Big(\mathbf{Q}_{g,B}^{\mathcal{H}_{<B}} \,\Big\|\, \nu_B^{\mathcal{H}_{<B}}\Big)\right)^{1/2} \\
&\le \left(\frac{297}{2} H_j w_j + \frac{9}{2} N^2 \mathfrak{d}_{\mathrm{path}}\right)^{1/2}.
\end{aligned} \tag{43}$$

Write $\lambda_j := \mathrm{Law}(\mathsf{Entry}_j)$. For λ_j-almost every cache-consistent entry e, let $\mathfrak{R}_{j,d}(e)$ be the conditional expectation of the sum in the first line of this display given $\mathsf{Entry}_j = e$. Disintegration with respect to this entry law gives

$$\overline{\mathfrak{R}}_{j,d} = \int \mathfrak{R}_{j,d}(e)\, \lambda_j(\,\mathrm{d}e).$$

Thus the levelwise certificate is averaged over the actual guarded entering law; it is not a uniform estimate for every deterministic entry. Indeed, $n_{d,\mathrm{blk}} \le N$, so the first two terms in (42), after multiplication by $n_{d,\mathrm{blk}}/2$, are at most $36 H_j w_j$ and $225 H_j w_j/2$, respectively.

Finally, $H_j \le \tau_j$ and $w_j \le n/\tau_j$ by (5), hence $H_j w_j \le n$. Equation (24) and Lemma I.2 give $\mathfrak{d}_{\mathrm{path}} \le N^{-2}$, so (43) yields

$$\overline{\mathfrak{R}}_{j,d} \le \left(\frac{297}{2} n + \frac{9}{2}\right)^{1/2} \le 15\sqrt{n+1}.$$

At level h, Definition G.5 makes every TV distance zero. The local query and depth bounds were verified in Section G.2. This proves Proposition B.5. □

H Recursive autospeculation

We restate the scheduler algorithm and theorem because the target coordinates in this work are transition transcripts rather than ordinary state increments. The result is the recursive speculative-rejection framework of Anari et al. (2026, Algorithms 5–6, Propositions 17–18 and 25, and Lemmas 23–24), with the conditional-prefix extension verified after the references. Section J reorganizes the cost proof into two scalar recurrences for fresh calls. This avoids tracking how often nodes are visited in the random recursion tree; the sampler and its asymptotic guarantees are unchanged.

H.1 CONDITIONAL TARGET AND PROPOSAL KERNELS

For $m \in \mathbb{Z}_{\geq 1}$, write $[m] = \{1, \dots, m\}$ and set $[0] = \varnothing$; for integers $1 \leq s \leq t$, write $[s : t] = \{s, \dots, t\}$. Let $h \in \mathbb{Z}_{\geq 0}$, set $N = 2^h$, let every Ω_i be a standard Borel space, and let μ be a probability law on $\Omega_1 \times \cdots \times \Omega_N$. Let $\mathscr{T}$ be the ordered perfect binary tree whose nodes are contiguous blocks $B = [s : t] \subseteq [N]$, write $|B|$ for the number of coordinates in B, and write $\mathscr{T}_r$ for its nodes at depth r. For $B = [s : t]$, define $< B = [s - 1]$, set $\Omega_B \coloneqq \prod_{i \in B} \Omega_i$ and $\Omega_{<B} \coloneqq \prod_{i \in <B} \Omega_i$, and denote the corresponding prefix marginal of μ by $\mu_{<B}$. At the root, $\Omega_{<B}$ is the singleton containing the empty prefix and $\mu_{<B}$ is its point mass. Thus $\mathscr{T}_r$ contains 2^r blocks and each of them has $N/2^r$ coordinates; this is a tree census, not a claim that an entire level is executed synchronously.

For $z \in \Omega_{<B}$, fix measurable regular conditional kernels μ_B^z for the law of Z_B given $Z_{<B} = z$ under $Z \sim \mu$, together with measurable proposal kernels ν_B^z. We use versions satisfying the usual tree-consistency identities, which hold outside a $\mu_{<B}$-null set. Assume that

$$\nu_B^z = \mu_B^z \qquad \text{at every leaf, for } \mu_{<B}\text{-almost every } z. \tag{44}$$

For every B, fix a σ-finite measure λ_B on Ω_B and jointly measurable target and proposal densities p_B^z, q_B^z with respect to λ_B. Call a prefix admissible when it belongs to these fixed, tree-consistent full-measure versions. Set

$$d_B(z) \coloneqq d_{\mathrm{TV}}(\mu_B^z, \nu_B^z)\,.$$

Fix $\rho \in (0, 1]$, set $c = 1 + \rho$, and define geometric batches by

$$\beta_m \coloneqq \lceil c^m \rceil - 1, \qquad \mathcal{J}_m \coloneqq \{\beta_m + 1, \dots, \beta_{m+1}\}, \qquad m = 0, 1, \dots. \tag{45}$$

For every node and prefix, define the overlap acceptance probability at a deterministic block value $x \in \Omega_B$ by

$$a_B^z(x) = \min\left\{1, \frac{p_B^z(x)}{q_B^z(x)}\right\}. \tag{46}$$

Here $p/q = +\infty$ when $q = 0 < p$ and $0/0 = 0$; the convention is immaterial under a draw from q. For a target-distributed fallback candidate, define the residual acceptance probability at a deterministic block value $y \in \Omega_B$ by

$$r_B^z(y) = \frac{(p_B^z(y) - q_B^z(y))_+}{p_B^z(y)}, \tag{47}$$

with value zero where $p_B^z = 0$.

The tree, kernel families, densities, and dominating measures are shared data for the recursion. Only the current node, its realized prefix, and the batching parameter are displayed as arguments. All draws use fresh randomness. Conditional on the input prefix, different candidate branches k are independent; within a branch, the right call is conditioned on the left output.

In this application, $\nu_C^{z'}$ is a law on block transcripts. The prefix z', together with the fixed interval-entering history, gives the full history $\mathfrak{h}_{<C}$ supplied to the generator and scorer with its carried guard cache.

Algorithm 6: Recursive speculative rejection $\mathsf{RS}_\rho(B, z)$

Input: A node $B = [s : t] \in \mathscr{T}$, a realized prefix $z \in \Omega_{<B}$, and $\rho \in (0, 1]$

Access: For every descendant node C and realized prefix $z' \in \Omega_{<C}$, sampling from $\nu_C^{z'}$ and evaluating $p_C^{z'}, q_C^{z'}$. In the guarded-FORS application, non-leaf proposals and their overlap ratios are produced by Algorithm 4; residual ratios for fallback transcripts are evaluated by Algorithm 5; leaves use ExactGuardedStep from Definition G.5

Output: A block $Z_B \in \Omega_B$ with law μ_B^z

if $|B| = 1$ **then**
  draw $Z_B \sim \nu_B^z$ and **return** Z_B;
Draw $X \sim \nu_B^z$ and $u \sim \mathrm{Unif}[0, 1]$;
if $u \le a_B^z(X)$, *with* a_B^z *from* (46) **then**
  return X;
Write $B = B_{\mathrm{L}} \dot\cup B_{\mathrm{R}}$ for its ordered pair of children, and set $m \leftarrow 0$;
while true **do**
  foreach $k \in \mathcal{J}_m$ *from* (45) in parallel **do**
    $Z_{\mathrm{L}}^{(k)} \leftarrow \mathsf{RS}_\rho(B_{\mathrm{L}}, z)$;
    $Z_{\mathrm{R}}^{(k)} \leftarrow \mathsf{RS}_\rho\Big(B_{\mathrm{R}}, (z, Z_{\mathrm{L}}^{(k)})\Big)$;
    set $Z^{(k)} \leftarrow (Z_{\mathrm{L}}^{(k)}, Z_{\mathrm{R}}^{(k)})$ and draw $u_k \sim \mathrm{Unif}[0, 1]$;
    set $A_k \leftarrow \mathbb{1}\{u_k \le r_B^z(Z^{(k)})\}$ using (47);
  wait until every candidate and residual test in the batch is complete, and set $\mathcal{A}_m \leftarrow \{k \in \mathcal{J}_m : A_k = 1\}$;
  if $\mathcal{A}_m \neq \varnothing$ **then**
    set $k_\star \leftarrow \min \mathcal{A}_m$ and **return** $Z^{(k_\star)}$;
  $m \leftarrow m + 1$;

Deterministic cache-threaded lift. The generic display returns only the probabilistic transcript coordinate. In this paper, $\mathsf{RS}_\rho(B, z \mid h, c)$ denotes the same recursion with an external history h and deterministic guard cache c threaded alongside it; when $e = (h, c)$ is the bundled interval entry, we abbreviate this by $\mathsf{RS}_\rho(B, z \mid e)$. Every proposal scorer and exact leaf returns an outgoing cache. An accepted proposal returns its cache; a rejected proposal discards it. In a fallback candidate, the accepted left-child cache enters the right-child call, and the cache of the accepted full candidate is returned to the parent; caches belonging to all rejected candidates are discarded. Consequently the cache returned at the root is a deterministic measurable function of the interval entry and accepted transcript, while projection onto transcript coordinates is exactly Algorithm 6. This lift changes neither the target law nor the scheduler cost analysis.

Within one candidate, the two recursive calls are sequential; different k's in a batch run in parallel. The full-batch barrier and the smallest-index rule make the output independent of completion order; an empty $\mathcal{J}_m$ is simply skipped. In particular, "levelwise" below refers to an analysis of rejection budgets. It does not mean that the sampler runs all nodes at a fixed tree depth in one execution stage.

For the query bound, fix a scale $q_{\mathrm{loc}} \ge 1$ and impose the following conditional expected-cost hypotheses, uniformly over nodes B and admissible prefixes z. Generating $X \sim \nu_B^z$ and performing its overlap test costs $O(q_{\mathrm{loc}}|B|)$ expected queries. Given an independent fresh $\widetilde{Z}_B \sim \mu_B^z$, its parent residual test costs $O(q_{\mathrm{loc}}|B|)$ expected queries, even when the cost is correlated with $\widetilde{Z}_B$ or its acceptance. An exact leaf costs $O(q_{\mathrm{loc}})$ expected queries. Each of these local operations uses $O(1)$ oracle rounds pointwise. Define

$$\mathfrak{R}_r := \sum_{B \in \mathscr{T}_r} \mathbb{E}_{Z \sim \mu} d_B(Z_{<B}), \qquad \mathfrak{R}_\star := \max_{0 \le r < h} \mathfrak{R}_r, \tag{48}$$

with $\mathfrak{R}_\star = 0$ when $h = 0$.

Theorem H.1 (Recursive autospeculation scheduler)**.** *Under the leaf condition* (44)*, the sampler in Algorithm 6 terminates almost surely and, for every node B and $\mu_{<B}$-almost every prefix $z \in \Omega_{<B}$, returns an exact sample from μ_B^z. At the root its output therefore has law μ.*

If Q *is the total number of oracle queries, then*

$$\mathbb{E}[\mathsf{Q}] = O\left(q_{\mathrm{loc}} N \sum_{r=0}^{h} (1+\rho)^r\right). \tag{49}$$

For the level budgets in (48)*, if* D *is the number of adaptive oracle rounds, then*

$$\mathbb{E}[\mathsf{D}] = O\left(1 + \sum_{r=0}^{h-1} c^r \left[2\mathfrak{R}_r + \frac{\mathfrak{R}_r \log N + \mathrm{e}^{-1}}{\log c}\right]\right). \tag{50}$$

Consequently, for $\rho = 1/\max\{1, h\}$*,*

$$\mathbb{E}[\mathsf{Q}] = O(q_{\mathrm{loc}} N (h+1)), \qquad \mathbb{E}[\mathsf{D}] = O\big((1+\mathfrak{R}_\star)(h+1)^3\big). \tag{51}$$

H.2 Source and proof location

The exactness argument and geometric-batching estimates are the recursive speculative-rejection analysis of Anari et al. (2026, Algorithms 5–6, Propositions 17–18 and 25, and Lemmas 23–24). Passing from ordinary state coordinates to the standard-Borel transcript coordinates above changes neither the overlap decomposition nor the recursion. The only point that must be retained explicitly is the conditioning on the incoming transcript prefix; Section J verifies that extension. Its fresh-call recurrences replace the random-tree occupation bookkeeping of the original cost analysis, while retaining the same work and depth bounds.

Proof of Theorem H.1. Apply the cited recursive overlap/residual construction to the conditional kernels μ_B^z, ν_B^z. Its exactness proof is measure-theoretic and therefore applies to the transcript spaces without change. The fresh-prefix recurrence in Theorem J.1 gives the two sharp displayed cost bounds (49) and (50), and their specialization at $\rho = 1/\max\{1, h\}$. □

Proof of Proposition B.7. Apply Theorem H.1 with $N = m$ and its simplified bounds (51) when the incoming prefix is deterministic. In the random-prefix case, condition on that prefix, apply the sharp bound (50), and then average term by term. The assumed fully averaged bound controls every resulting level budget by $\mathfrak{R}_\star$. The estimates in the final paragraph of the theorem's proof then give (2); in particular, no expectation is interchanged with a maximum over levels. □

H.3 Proof of the one-interval theorem

Proof of Theorem B.6. Fix an interval j and condition on a cache-consistent entry $\mathsf{Entry}_j = e$. Write μ_j^e for the conditional guarded law of the stopped transition transcripts in $\mathcal{I}_j$; this is the target law μ for the recursive scheduler. Operationally, each node call also receives the guard cache from Section F. Scoring a proposed or fallback block returns its outgoing cache. A rejected proposal's cache is discarded; an accepted left child supplies the cache for its right sibling, and an accepted node returns its cache to its parent. This is deterministic side state and is not an extra coordinate of Ω_B, so the generic transcript law is unchanged and no prior oracle answer is recomputed.

The proposal kernels are the laws generated by ParallelFORSBlock in Algorithm 4 at internal nodes and the laws of ExactGuardedStep from Definition G.5 at leaves. Their common transcript space and normalized densities are explicit in (19) and (27). The shared capped-count, uniform, and standard-Gaussian head factors are positive; strict bit probabilities and nondegenerate candidate Gaussians give mutual absolute continuity on the common stopped support. The retained-prefix formulas define the conditional kernels for every admissible prefix.

At an internal node, ParallelFORSBlock obtains the admitted anchor and invokes Algorithm 3 for all $|B|$ frozen proposal steps in parallel. Its call to ScoreFORSBlock evaluates the overlap score in the two subsequent oracle stages. The same scorer, called separately, evaluates the residual score on a fresh target-law candidate in two stages. Their bounds in (39) and the paragraph following it, together with the leaf bound in Lemma E.1, verify the three conditional expected-cost hypotheses with $q_{\mathrm{loc}} = O(1)$, uniformly in the transcript caps; each local operation has pointwise $O(1)$ adaptive oracle depth. Hence Theorem H.1 gives exactness for every cache-consistent e and the uniform conditional query bound asserted in Theorem B.6. By the boundary update in Section B.3, the

committed outgoing cache followed by the reset is precisely the cache component of $\mathsf{Next}_j(e, \mathsf{Z}^{(j)})$. Exactness therefore holds jointly with the complete entry to interval $j+1$.

It remains to justify the prefix-averaged depth bound. Recall from the proof of Proposition B.5 that $\lambda_j = \mathrm{Law}(\mathsf{Entry}_j)$ is the actual entry law. For a realization e, let $\mathfrak{R}_{j,r}(e)$ denote the conditional level budget introduced in the proof of Proposition B.5. Under the two numerical conditions verified in Lemma I.2, that proposition, specifically (1), gives at every internal level

$$\int \mathfrak{R}_{j,r}(e)\, \lambda_j(\,\mathrm{d}e) = \overline{\mathfrak{R}}_{j,r} = O(\sqrt{n+1}).$$

Condition on e, apply the sharp depth bound (50), and only then integrate it term by term against λ_j. The displayed estimate and the prescribed geometric batching parameter give

$$\mathbb{E}[\mathsf{D}_j] = O\big((1+\sqrt{n+1})\log^3(N+1)\big)\,.$$

This order of operations does not interchange expectation with a maximum over levels. It proves the averaged depth conclusion and completes the theorem. □

I Parameter specification and verification

The high-level algorithm needs a starting precision, a coarse time partition with fine steps inside each interval, and finite transcript safety caps. We now specify them, give the complete first-zero-stopped sampler using the components developed in the preceding sections, and then verify in one place every numerical condition used earlier.

Definition I.1 (Deterministic parameter setup)**.** Fix once and for all one universal numerical constant $A_\star \geq 1$ satisfying Lemma I.2; that lemma proves that such a choice exists. This is a compile-time constant, not an algorithmic input or tuning parameter. For inputs $n \geq 1$, $\delta > 0$, $0 < \varepsilon \leq \varepsilon_0$ as in Theorem 2.1, and $M_2^2 \geq 0$, the routine ParameterSetup first sets

$$L := \log\left(\frac{\mathrm{e}(n+1)(1+M_2^2/\delta)}{\varepsilon}\right), \qquad \ell := 1+\log(n+L). \tag{52}$$

Define the internal path accuracy and terminal precision by

$$\zeta := \frac{\varepsilon^2}{A_\star^4 (n+1)^4 (1+M_2^2/\delta)^4 L^8}, \qquad T := \delta^{-1}, \tag{53}$$

so, under the standing small-accuracy convention, $L > 1$ and $\zeta \leq \varepsilon^2/16$. Define the starting precision by

$$t_0 := \min\left\{T, \frac{\zeta}{2M_2^2}\right\}. \tag{54}$$

Here $\zeta/(2M_2^2)$ is interpreted as $+\infty$ when $M_2^2 = 0$. If $t_0 = T$, return the tag $\mathsf{Gaussian}(T, t_0)$ and stop. Otherwise we are in the nontrivial branch

$$M_2^2 > 0, \qquad 0 < t_0 < T.$$

In this branch, let

$$\tau_j := \min\{2^j t_0, T\}, \qquad j = 0, 1, \ldots, J,$$

where J is the first index for which $\tau_J = T$. Write $H_j = \tau_{j+1} - \tau_j$ for $0 \leq j < J$; then $H_j \leq \tau_j$. Choose

$$N := 2^{\lceil \log_2(A_\star(n+L)L)\rceil}, \qquad h := \log_2 N, \tag{55}$$

and subdivide every $[\tau_j, \tau_{j+1}]$ into N equal steps of size $a_j = H_j/N$. Set $M := JN$ and write

$$t_{j,k} := \tau_j + k a_j, \qquad 0 \leq k \leq N,$$

and flatten the grid by setting $t_{jN+k} = t_{j,k}$ and $a_{jN+k} = a_j$ for $0 \leq k < N$, and set $t_M = T$. For any flattened state path, write $x_{j,k} = x_{jN+k}$. Thus

$$t_0 < t_1 < \cdots < t_M = T,$$

with $a_i = t_{i+1} - t_i$; we retain (j, k) notation when the coarse interval matters, and thus $a_i = a_j = H_j/N$ whenever step i lies in interval j. We call a_i the *time-step size* at step i. Because the grid is not uniform across coarse intervals, the dimensionless quantity used by the one-step analysis is the *relative time-step size* a_i/t_i, or equivalently its inverse ratio t_i/a_i. For any length-M array v, write $v^{(j)} = (v_{jN}, \dots, v_{jN+N-1})$ for its interval-j restriction. Recall the interval index set introduced in Section B:

$$\mathcal{I}_j = \{jN, \dots, (j+1)N - 1\}.$$

Finally set

$$\theta := \frac{\zeta}{4M}, \qquad R := \left\lceil \mathrm{e}^2 \log \frac{16M}{\zeta} \right\rceil, \qquad S := \left\lceil 16 + \log \frac{16\mathrm{e}^{3/2}M}{\zeta} \right\rceil. \tag{56}$$

Thus R is the finite candidate-attempt safety cap and S is the cap on the Poisson factor count. They control accuracy and finite transcript size; by first-zero stopping they do not enter the expected oracle-cost coefficients. Set $\rho = 1/\max\{1, h\}$ and define α, K, $(w_j)_{j=0}^{J-1}$, and the guard thresholds $(\Gamma_j)_{j=0}^{J-1}$ together by (26). Return

$$\mathsf{Recursive}\big(T, t_0, J, N, M, ((t_{j,k})_{k=0}^{N}, a_j, \mathcal{I}_j)_{j=0}^{J-1}, R, S, \rho, (\Gamma_j)_{j=0}^{J-1}\big).$$

This routine is deterministic and uses neither oracle evaluations nor random bits. The symbols $A_\star, \zeta, (\tau_j, H_j)_j, h, \theta, \alpha, K$, and $(w_j)_j$ name deterministic quantities internal to the setup whose notation is retained for the analysis; they are not additional inputs or outputs.

I.1 Complete sampler

All ingredients used below have now been defined. Recall from Section H that the complete execution history entering coarse interval j is

$$\mathcal{H}_0 = (X_0^g), \qquad \mathcal{H}_j = (X_0^g, \mathsf{Z}_0, \dots, \mathsf{Z}_{jN-1}) \quad (1 \le j \le J).$$

The corresponding guard cache $\mathfrak{c}_j$ stores the current state, permanent frozen flag, last admitted drift, and partial budgets. Although these are mathematically determined by the history and prior oracle answers, the algorithm carries them explicitly rather than re-querying the oracle. Together they form the interval entry $\mathsf{Entry}_j = (\mathcal{H}_j, \mathfrak{c}_j)$ from Section B.3. For $0 \le j < J$, identify local position $r \in [N]$ with global step $jN + r - 1 \in \mathcal{I}_j$ and transport the node kernels along this bijection. Conditional on Entry_j, the notation $\mathsf{RS}_\rho([N], \varnothing \mid \mathsf{Entry}_j)$ means that Algorithm 6 is run on this relabeled tree. Here $\mathcal{H}_j$ is an external conditioning parameter, $\mathfrak{c}_j$ is threaded deterministic side state, and $\varnothing$ is the empty local transcript prefix at the root.

Algorithm 7: HIGH-ACCURACY AUTOSPEC: complete specification

Input: $n \geq 1$, $\delta > 0$, a sufficiently small target accuracy $\varepsilon > 0$, and a second-moment bound $M_2^2 \geq 0$
Oracle: Exact evaluations of $f(t, y)$ for every $t > 0$ and $y \in \mathbb{R}^n$
Output: A random vector $X_{\mathrm{out}} \in \mathbb{R}^n$

Set $\mathsf{mode} \leftarrow \textsc{ParameterSetup}(n, \delta, \varepsilon, M_2^2)$ according to Definition I.1;
if $\mathsf{mode} = \mathsf{Gaussian}(T, t_0)$ **then**
  draw $X_{t_0} \sim \mathcal{N}(0, t_0 I_n)$ and **return** X_{t_0}/T;
Unpack $\mathsf{mode} = \mathsf{Recursive}(T, t_0, J, N, M, ((t_{j,k})_{k=0}^{N}, a_j, \mathcal{I}_j)_{j=0}^{J-1}, R, S, \rho, (\Gamma_j)_{j=0}^{J-1})$;
Draw $X_0^g \sim \mathcal{N}(0, t_0 I_n)$, set $\mathcal{H}_0 \leftarrow (X_0^g)$, and initialize $\mathfrak{c}_0$ with current state X_0^g, an active flag, no last admitted drift, empty per-level anchors, and zero running and completed-block totals; set $\mathsf{Entry}_0 \leftarrow (\mathcal{H}_0, \mathfrak{c}_0)$;
for $j = 0, \ldots, J-1$ **do**
  let $\mathscr{T}_j$ be the ordered perfect binary tree on local positions $[N]$, with r carrying global transition step $jN + r - 1$;
  unpack $\mathsf{Entry}_j = (\mathcal{H}_j, \mathfrak{c}_j)$ and equip every node $B \in \mathscr{T}_j$ with the guarded conditional target obtained by multiplying the guarded stopped single-step densities specified by (19) and (27), using threshold Γ_j;
  equip each non-leaf B with the proposal-and-overlap interface PARALLELFORSBLOCK and the residual scorer SCOREFORSBLOCK from Algorithms 4 and 5, and equip each leaf with EXACTGUARDEDSTEP from Definition G.5, always passing and returning the current guard cache;
  sample $((\widetilde{\mathsf{Z}}_r^{(j)})_{r\in[N]}, \mathfrak{c}_j^{\mathrm{out}}) \leftarrow \mathsf{RS}_\rho([N], \varnothing \mid \mathsf{Entry}_j)$, threading the cache as in Section H;
  set $\mathsf{Z}_{jN+r-1} \leftarrow \widetilde{\mathsf{Z}}_r^{(j)}$ for every $r \in [N]$;
  decode all selected increments and recover the states through the interval by a prefix scan using (15); set

$$\mathsf{Entry}_{j+1} \leftarrow \mathsf{Next}_j\big(\mathsf{Entry}_j, (\mathsf{Z}_i)_{i\in\mathcal{I}_j}\big),$$

  reusing $\mathfrak{c}_j^{\mathrm{out}}$ rather than replaying its guard updates, and unpack it as $(\mathcal{H}_{j+1}, \mathfrak{c}_{j+1})$;
return $X_{\mathrm{out}} \leftarrow X_M^g/T$;

Thus Algorithm 7 is a complete top-level specification. At the first call inside each coarse interval, the scheduler is at the root $B = [N]$, so PARALLELFORSBLOCK invokes Algorithm 3 for all N fine steps at once. There is no sequential loop of N proposal-oracle calls. Sequential dependence appears only in the ordered child calls used to repair a rejected block. The root call in each loop is precisely the sampler analyzed in Theorem B.6. It therefore returns the exact conditional guarded transcript law for that interval together with its corresponding outgoing cache. The map Next_j in Section B.3 appends the accepted transcript and applies the deterministic boundary update, giving exactly the cache-consistent entry to the next interval. In particular, the permanent frozen flag and last admitted drift are retained; only interval-local dyadic fields are reset. All normalized densities and randomization are explicit, and no endpoint normalizing constant is supplied to the algorithm.

I.2 VERIFICATION

Direct-Gaussian branch. If $t_0 = T$, then $TM_2^2 \leq \zeta/2$. By Lemma C.2,

$$D_{\mathrm{KL}}(\mathrm{Law}(Y_T) \,\|\, \mathcal{N}(0, TI_n)) \leq \frac{TM_2^2}{2} \leq \frac{\zeta}{4}.$$

Pinsker's inequality and invariance under rescaling therefore give

$$d_{\mathrm{TV}}\big(\mathrm{Law}(Y_T/T), \mathcal{N}\big(0, T^{-1}I_n\big)\big) \leq \sqrt{\zeta/8} \leq \varepsilon.$$

These are respectively the target and the output laws in the Gaussian branch of Algorithm 7; that branch uses zero oracle queries and zero adaptive oracle rounds.

We now verify every numerical condition used earlier. Throughout this subsection we are in the nontrivial branch $M_2^2 > 0$ and $t_0 < T$.

Lemma I.2 (Parameter check)**.** *There is a universal A_0, depending only on the universal constant C_{FORS} in Theorem D.4, such that every fixed $A_\star \geq A_0$ has the following properties.*

1. *At every fine step,*

$$\frac{t_i}{a_i} \geq C_{\mathsf{FORS}} \left(n \log \frac{1}{\theta} + \log^2 \frac{1}{\theta} \right). \tag{57}$$

2. *The two truncation tails satisfy*

$$p_S \le \frac{\zeta}{16\mathrm{e}^{3/2}M}, \qquad \exp(-\mathrm{e}^{-2}R) \le \frac{\zeta}{16M}, \tag{58}$$

and hence the smallness assumptions (20) hold.

3. *The three global error conditions hold:*

$$\zeta \le \frac{\varepsilon^2}{32}, \qquad \zeta \le \frac{\varepsilon}{4J(h+1)}, \qquad N^2\zeta \le 1. \tag{59}$$

4. *With constants allowed to depend on the fixed universal choice of $A_\star$,*

$$J = O(L), \quad N = O((n+L)L), \quad R, S = O(L), \quad h+1 = O(\ell). \tag{60}$$

Proof. We proceed in the order needed by the main proof.

Elementary size bounds. The definition (52) and the standing small-accuracy convention together with $n \ge 1$ imply $L > 1$. In the nontrivial branch, $t_0 = \zeta/(2M_2^2)$ and therefore

$$J = \left\lceil \log_2 \frac{T}{t_0} \right\rceil = \left\lceil \log_2 \frac{2M_2^2}{\delta\zeta} \right\rceil.$$

Writing $u = 1 + M_2^2/\delta$, we have

$$\log \frac{1}{\zeta} = 2\log\frac{1}{\varepsilon} + 4\log A_\star + 4\log(n+1) + 4\log u + 8\log L. \tag{61}$$

Every term except $\log A_\star$ is $O(L)$, so

$$J \le C(1 + \log A_\star)L \tag{62}$$

for a numerical C. Also, from (55),

$$A_\star(n+L)L \le N < 2A_\star(n+L)L \tag{63}$$

and

$$h + 1 \le C\big(1 + \log A_\star + \log(n+L)\big) \le C(1 + \log A_\star)\ell. \tag{64}$$

Here $\log L \le \log(n+L)$. Since $n \le \mathrm{e}^L$ and $L \le \mathrm{e}^L$, we also have $\ell = O(L)$. These estimates already prove the asserted bounds for J, N, h.

Because $M = JN$, (61)–(64) give

$$\log \frac{1}{\theta} = \log \frac{4M}{\zeta} \le C(1 + \log A_\star)L. \tag{65}$$

It follows from (56) that $R, S = O(L)$, completing (60).

The FORS relative time-step condition. Every fine step in interval j has size $a_j = H_j/N$. Since $t_i \ge \tau_j$ and $H_j \le \tau_j$,

$$\frac{t_i}{a_i} \ge N. \tag{66}$$

By (65), the right side of (57) is, for a numerical constant C', at most

$$C'C_{\mathsf{FORS}}(1 + \log A_\star)^2(n+L)L.$$

Since $A_\star/(1+\log A_\star)^2 \to \infty$, enlarge A_0 so that every $A_\star \ge A_0$ satisfies

$$A_\star \ge C'C_{\mathsf{FORS}}(1 + \log A_\star)^2.$$

Then (63) and (66) prove (57). The same bounds give $0 < \theta < 1/2$ after enlarging A_0 if needed.

Poisson and attempt tails. The definition of R directly gives

$$\exp(-\mathrm{e}^{-2}R) \le \frac{\zeta}{16M}.$$

For $X \sim \mathrm{Poi}(2)$ and $S+1 \ge 2\mathrm{e}^2$, the Chernoff bound gives

$$\mathbb{P}(X > S) \le \left(\frac{2\mathrm{e}}{S+1}\right)^{S+1} \le \mathrm{e}^{-(S+1)}.$$

Our choice has $S \ge 16$, so the premise holds, and

$$\mathrm{e}^{-(S+1)} \le \frac{\zeta}{16\mathrm{e}^{3/2}M}.$$

This proves (58). Since $\zeta < 1$ and $M \ge 1$, both quantities in (20) are in fact at most $1/16$.

Global smallness. The first inequality in (59) follows by taking A_0 large, since every other denominator factor in (53) is at least one. For the third, combine (63) with $n + L \le (n+1)L$:

$$\begin{aligned} N^2\zeta &\le \frac{4A_\star^2(n+L)^2L^2\varepsilon^2}{A_\star^4(n+1)^4u^4L^8} \\ &\le \frac{4\varepsilon^2}{A_\star^2(n+1)^2u^4L^4} \le 1. \end{aligned}$$

The last inequality uses the standing small-accuracy convention and the fixed lower bound on $A_\star$.

Finally, (62) and (64) give, for a numerical constant C,

$$J(h+1) \le C(1+\log A_\star)^2L^2.$$

On the other hand, the standing small-accuracy convention gives

$$\frac{\zeta}{\varepsilon} = \frac{\varepsilon}{A_\star^4(n+1)^4u^4L^8} \le \frac{1}{A_\star^4L^8}.$$

Consequently,

$$\frac{4J(h+1)\zeta}{\varepsilon} \le \frac{4C(1+\log A_\star)^2}{A_\star^4L^6} \le 1$$

after enlarging A_0 once more, uniformly for every $A_\star \ge A_0$. This is the remaining inequality in (59). □

Corollary I.3 (Global runtime bookkeeping)**.** *In the nontrivial branch, the sequential composition of the J interval samplers from Theorem B.6 satisfies*

$$\mathbb{E}[\mathsf{Q}] = O\big((n+L)L^2\ell\big), \qquad \mathbb{E}[\mathsf{D}] = O\big(\sqrt{n}\,L\ell^3\big).$$

Proof. The intervals are run sequentially, so both their query counts and their adaptive depths add. Since $h = \log_2 N$, Theorem B.6 gives

$$\mathbb{E}[\mathsf{Q}] = O(M(h+1)), \qquad \mathbb{E}[\mathsf{D}] = O\big(J(1+\sqrt{n+1})(h+1)^3\big).$$

Linearity of expectation is sufficient here; independence between intervals is not required. Substituting (60) yields the claim, using $n \ge 1$ to absorb $1+\sqrt{n+1}$ into $O(\sqrt{n})$. □

J A SCALAR-RECURRENCE PROOF FOR RECURSIVE SPECULATIVE REJECTION

We verify the expected work and depth bounds for the recursive speculative rejection sampler of Anari et al. (2026, Algorithms 5–6 and Section 4). Their analysis tracks calls made throughout the random recursion tree. Here we instead condition on a fixed incoming prefix, analyze one fresh call, and only then average over target prefixes. The resulting query and excess-depth bounds each reduce to a scalar recurrence. This shorter proof supplies the conditional-prefix version needed for transcript-valued blocks in Theorem H.1; it changes neither the sampler nor its asymptotic guarantees.

J.1 Setup and statement

Use the notation of Section H. Let $h \in \mathbb{Z}_{\geq 0}$ and $N = 2^h$, and let $\mathscr{T}$ be the perfect binary tree of contiguous blocks of $[N]$. For a block $B = [s : t]$, write $< B = [s - 1]$ and let μ_B^z and ν_B^z be the target and proposal laws conditional on $Z_{<B} = z$, where $z \in \Omega_{<B}$. Their densities are p_B^z and q_B^z, and

$$d_B(z) = d_{\mathrm{TV}}(\mu_B^z, \nu_B^z)\,, \qquad \mathfrak{R}_r = \sum_{B \in \mathscr{T}_r} \mathbb{E}_{Z \sim \mu} d_B(Z_{<B}), \qquad \mathfrak{R}_\star = \max_{0 \leq r < h} \mathfrak{R}_r.$$

When $h = 0$, the maximum is over the empty set and we set $\mathfrak{R}_\star = 0$. Assume the following uniformly over blocks and admissible prefixes. The proposal together with its overlap test has conditional expected query cost $O(q_{\mathrm{loc}}|B|)$ under ν_B^z; a residual test on a fresh μ_B^z candidate has conditional expected cost $O(q_{\mathrm{loc}}|B|)$; and an exact leaf has expected cost $O(q_{\mathrm{loc}})$. These local operations use $O(1)$ oracle rounds pointwise. Here a prefix is *admissible* if it belongs to the fixed, tree-consistent full-measure versions of the conditional kernels and densities from Section H. At a leaf the proposal is exact. All prefix identities below are understood on these sets, hence target-almost surely.

For completeness, the sampler first draws $X \sim \nu_B^z$ and accepts with probability $\min\{1, p_B^z(X)/q_B^z(X)\}$. On rejection, it generates fallback candidates $Z^{(k)}$ by sequentially sampling the left child and then the right child, using the recursive sampler. Each candidate passes its residual test with probability

$$r_B^z(Z^{(k)}) = \frac{(p_B^z(Z^{(k)}) - q_B^z(Z^{(k)}))_+}{p_B^z(Z^{(k)})},$$

with value zero when the denominator is zero. Candidate branches use independent randomness and are processed in geometric batches

$$c = 1 + \rho, \qquad \beta_m = \lceil c^m \rceil - 1, \qquad \mathcal{J}_m = \{\beta_m + 1, \ldots, \beta_{m+1}\}, \qquad m \geq 0,$$

where $0 < \rho \leq 1$. Empty batches are skipped. A batch is completed before proceeding, and the smallest accepted index is returned from the first successful batch. Thus selection does not depend on completion order. This is Algorithm 6.

Theorem J.1 (Expected bounds from fresh-call recurrences)**.** *The sampler terminates almost surely and returns an exact sample from μ. Its total queries* Q *and adaptive oracle rounds* D *satisfy*

$$\mathbb{E}[\mathsf{Q}] = O\left(q_{\mathrm{loc}} N \sum_{r=0}^{h} c^r\right), \tag{67}$$

$$\mathbb{E}[\mathsf{D}] = O\left(1 + \sum_{r=0}^{h-1} c^r \left[\mathfrak{R}_r + \frac{\mathfrak{R}_r \log N + \mathrm{e}^{-1}}{\log c}\right]\right). \tag{68}$$

In particular, when $h \geq 1$, $\rho = 1/h$ gives

$$\mathbb{E}[\mathsf{Q}] = O(q_{\mathrm{loc}} N h), \qquad \mathbb{E}[\mathsf{D}] = O((1 + \mathfrak{R}_\star) h^3).$$

The constants depend only on the uniform local cost bounds.

J.2 Correctness, batching, and unbiased candidate costs

Correctness. Fix a block and prefix and abbreviate its densities by p, q and its rejection probability by d. Induct on block height. Exact child calls produce an exact target candidate, so the accepted part of a fallback trial has subprobability density $(p - q)_+$ and mass d. The initially accepted proposal has subprobability density $\min\{p, q\}$. For $d > 0$, the first accepted fallback candidate therefore has density $(p - q)_+/d$, and the output density is

$$\min\{p, q\} + d\frac{(p - q)_+}{d} = p.$$

The fallback succeeds almost surely since its independent trials each succeed with probability d. If $d = 0$, the initial proposal is accepted almost surely. Together with exact leaves and finite batches, this proves exactness and almost-sure termination by induction.

The local variables. For a fixed block B and admissible deterministic prefix z, let $\mathbb{P}_{B,z}$ and $\mathbb{E}_{B,z}$ denote probability and expectation over the fresh randomness of one call to the recursive sampler. When B is clear, we write $\mathbb{P}(\cdot \mid z)$ and $\mathbb{E}[\cdot \mid z]$ as shorthand for this fresh-call law; adding "fallback" means further conditioning on rejection of the initial proposal. For one fixed non-leaf call at (B, z), let

$$I = \mathbb{1}\{\text{this call rejects its initial proposal}\}, \qquad \mathbb{E}[I \mid z] = d_B(z).$$

Conditional on entering fallback, let $J_\star$ be the first accepted candidate index, $L_\star$ the total number of candidates launched, and $M_\star$ the number of nonempty batches processed. These variables refer only to the current call. Set $L_\star = M_\star = 0$ if $I = 0$.

Two batching estimates. Write $d = d_B(z) > 0$. Conditional on fallback, $J_\star \sim \mathrm{Geom}(d)$ on $\{1, 2, \ldots\}$. If $J_\star \in \mathcal{J}_m$, then

$$J_\star \geq \lceil c^m \rceil, \qquad L_\star = \lceil c^{m+1} \rceil - 1 < cJ_\star, \qquad M_\star \leq m + 1 \leq 1 + \log_c J_\star.$$

Using $\mathbb{E}[J_\star \mid \text{fallback}, z] = 1/d$ and Jensen's inequality gives

$$d\mathbb{E}[L_\star \mid \text{fallback}, z] \leq c, \qquad d\mathbb{E}[M_\star \mid \text{fallback}, z] \leq \psi_\rho(d), \tag{69}$$

where $\psi_\rho : [0, 1] \to \mathbb{R}_{\geq 0}$ is

$$\psi_\rho(x) = x\left(1 + \frac{\log(1/x)}{\log c}\right) \quad (0 < x \leq 1), \qquad \psi_\rho(0) = 0.$$

In particular, $\mathbb{E}[IM_\star \mid z] \leq \psi_\rho(d_B(z))$ for every admissible prefix, including those with $d_B(z) = 0$.

Launching a batch does not bias its candidate costs. At a fixed prefix, imagine an infinite independent sequence of complete fallback candidate computations. Let Cost_k be the same nonnegative cost functional applied to candidate k, possibly including its child-call costs. The decision to launch candidate k depends only on the initial rejection and batches strictly preceding the batch containing k. It is independent of candidate k's entire computation. For $d > 0$, Tonelli's theorem and (69) imply

$$\begin{aligned}\mathbb{E}\left[I \sum_{k=1}^{L_\star} \mathsf{Cost}_k \,\middle|\, z\right] &= d \sum_{k \geq 1} \mathbb{P}(L_\star \geq k \mid \text{fallback}, z)\, \mathbb{E}[\mathsf{Cost}_1 \mid z] \\ &= d\mathbb{E}[L_\star \mid \text{fallback}, z]\, \mathbb{E}[\mathsf{Cost}_1 \mid z] \leq c\mathbb{E}[\mathsf{Cost}_1 \mid z]. \end{aligned} \tag{70}$$

For $d = 0$ the left side is zero, so the inequality still holds. A candidate's cost may be correlated with its output or acceptance; neither independence is needed here.

J.3 A SCALAR RECURRENCE FOR EXPECTED WORK

Let $\mathsf{w}_B(z)$ be the expected total queries of a fresh call at (B, z). The initial local operations have the assumed conditional expected cost. For residual tests, the decision to launch candidate k is independent of that candidate's cost, while the fresh candidate has law μ_B^z. Therefore (70) bounds the combined initial and residual local work by $C_{\mathrm{work}} q_{\mathrm{loc}} |B|$ in expectation for a universal local-cost constant C_{work}. Apply (70) to the sum of the two child query costs:

$$\mathsf{w}_B(z) \leq C_{\mathrm{work}} q_{\mathrm{loc}} |B| + c\left[\mathsf{w}_{B_{\mathrm{L}}}(z) + \mathbb{E}_{Z_{\mathrm{L}} \sim \mu^z_{B_{\mathrm{L}}}} \mathsf{w}_{B_{\mathrm{R}}}\big((z, Z_{\mathrm{L}})\big)\right]. \tag{71}$$

Here (z, Z_{L}) is the concatenated prefix entering the right child. At a leaf, $\mathsf{w}_B(z) \leq C_{\mathrm{work}} q_{\mathrm{loc}}$. Backward induction in (71) bounds the expectations uniformly over admissible prefixes and proves their finiteness.

Average these fresh-call costs under the target prefix laws:

$$\mathsf{W}_r := \sum_{B \in \mathcal{J}_r} \mathbb{E}_{Z \sim \mu} \mathsf{w}_B(Z_{<B}).$$

The conditional chain rule identifies the right-child average in (71) with its target-prefix average. Since $\sum_{B \in \mathcal{J}_r} |B| = N$, we obtain the scalar recurrence

$$\mathsf{W}_r \leq C_{\mathrm{work}} q_{\mathrm{loc}} N + c\mathsf{W}_{r+1}, \qquad \mathsf{W}_h \leq C_{\mathrm{work}} q_{\mathrm{loc}} N.$$

Unrolling it gives

$$\mathbb{E}[\mathsf{Q}] = \mathsf{W}_0 \leq C_{\mathrm{work}} q_{\mathrm{loc}} N \sum_{r=0}^{h} c^r,$$

which proves (67).

J.4 A SCALAR RECURRENCE FOR EXPECTED EXCESS DEPTH

Choosing the constant baseline. Let r_{prop} and r_{test} be uniform upper bounds on the rounds for proposal generation and its initial acceptance test, and let r_{leaf} bound an exact leaf call. Set

$$d_0 = \max\{r_{\mathrm{prop}} + r_{\mathrm{test}}, r_{\mathrm{leaf}}\}.$$

The local cost assumptions make $d_0 = O(1)$, independent of N, h and the rejection probabilities. For example, if all three bounds are one round, one may take $d_0 = 2$. This bounds the initial part of a call, before fallback, as well as an entire leaf call. It does not bound a whole non-leaf recursive call.

For the random depth D_B of a fresh call, define

$$e_B(z) = \mathbb{E}[(\mathsf{D}_B - d_0)_+ \mid z].$$

The positive part is nonnegative, and $e_B = 0$ at a leaf.

Deriving the pathwise inequality. Fix a finite uniform bound r_0 on the rounds needed to evaluate the residual tests for a batch in parallel; the local cost assumptions give $r_0 = O(1)$. Let $\mathcal{B}_{\mathrm{run}}$ be the collection of nonempty batches processed by the current call. If $I = 0$, this collection is empty and $\mathsf{D}_B \le d_0$. Otherwise the two children of a candidate run sequentially, while different candidates run in parallel. It follows, for every execution, that

$$\mathsf{D}_B \le d_0 + I \sum_{\mathcal{J} \in \mathcal{B}_{\mathrm{run}}} \left[r_0 + \max_{k \in \mathcal{J}} \big(\mathsf{D}^{(k)}_{B_{\mathrm{L}}} + \mathsf{D}^{(k)}_{B_{\mathrm{R}}} \big) \right]. \tag{72}$$

Write $\mathsf{Exc}^{(k)}_{\mathrm{L}} = (\mathsf{D}^{(k)}_{B_{\mathrm{L}}} - d_0)_+$ and $\mathsf{Exc}^{(k)}_{\mathrm{R}} = (\mathsf{D}^{(k)}_{B_{\mathrm{R}}} - d_0)_+$. Each child's depth is at most d_0 plus its excess, so one batch contributes at most

$$\begin{aligned} r_0 + \max_{k \in \mathcal{J}} \big(\mathsf{D}^{(k)}_{B_{\mathrm{L}}} + \mathsf{D}^{(k)}_{B_{\mathrm{R}}} \big) &\le r_0 + 2d_0 + \max_{k \in \mathcal{J}} (\mathsf{Exc}^{(k)}_{\mathrm{L}} + \mathsf{Exc}^{(k)}_{\mathrm{R}}) \\ &\le C_0 + \sum_{k \in \mathcal{J}} (\mathsf{Exc}^{(k)}_{\mathrm{L}} + \mathsf{Exc}^{(k)}_{\mathrm{R}}), \end{aligned}$$

where $C_0 = r_0 + 2d_0$. The second inequality is simply maximum at most sum for nonnegative quantities. The processed batches partition the $L_\star$ launched candidates. Substituting into (72), subtracting d_0, and taking the positive part therefore gives

$$(\mathsf{D}_B - d_0)_+ \le C_0 I M_\star + I \sum_{k=1}^{L_\star} \left[(\mathsf{D}^{(k)}_{B_{\mathrm{L}}} - d_0)_+ + (\mathsf{D}^{(k)}_{B_{\mathrm{R}}} - d_0)_+ \right]. \tag{73}$$

The constant child baselines are charged once per batch. Only excess depth is bounded by a sum over every candidate. This is the reason for subtracting d_0 before applying maximum at most sum.

Taking expectations and averaging prefixes. Apply (69) and (70) to (73):

$$e_B(z) \le C_0 \psi_\rho(d_B(z)) + c \left[e_{B_{\mathrm{L}}}(z) + \mathbb{E}_{Z_{\mathrm{L}} \sim \mu^z_{B_{\mathrm{L}}}} e_{B_{\mathrm{R}}}\big((z, Z_{\mathrm{L}})\big) \right].$$

Define deterministic level sums

$$\mathcal{E}_r := \sum_{B \in \mathscr{T}_r} \mathbb{E}_\mu e_B(Z_{<B}), \qquad \mathcal{P}_r := \sum_{B \in \mathscr{T}_r} \mathbb{E}_\mu \psi_\rho(d_B(Z_{<B})).$$

The conditional chain rule gives

$$\mathcal{E}_r \le C_0 \mathcal{P}_r + c\mathcal{E}_{r+1}, \qquad \mathcal{E}_h = 0.$$

Unrolling yields

$$\mathbb{E}[\mathsf{D}] \le d_0 + \mathcal{E}_0 \le d_0 + C_0 \sum_{r=0}^{h-1} c^r \mathcal{P}_r. \tag{74}$$

All quantities can first be treated as nonnegative extended expectations; the finite upper bounds below establish integrability.

Finishing with the rejection budget. For $x \in [0,1]$, with the continuous convention at zero,

$$x \log(1/x) \le x \log N + \frac{1}{\mathrm{e}N}.$$

To see this, subtract $x \log N$, set $u = Nx$, and use $u \log(1/u) \le 1/\mathrm{e}$. Since $|\mathscr{T}_r| \le N$, it follows that

$$\mathcal{P}_r \le \mathfrak{R}_r + \frac{\mathfrak{R}_r \log N + \mathrm{e}^{-1}}{\log c}.$$

Together with (74), this proves (68). Taking $c = 1 + 1/h$ gives $c^h \le \mathrm{e}$, $\sum_{r=0}^{h} c^r = O(h)$, $1/\log c = O(h)$, and $\log N = \Theta(h)$. Hence

$$\mathbb{E}[\mathsf{Q}] = O(q_{\mathrm{loc}} N h), \qquad \mathbb{E}[\mathsf{D}] = O(1 + \mathfrak{R}_\star h^3 + h^2) = O((1 + \mathfrak{R}_\star) h^3).$$

If $h = 0$, the root is an exact leaf and the bounds in Theorem H.1 follow directly from the local costs.

J.5 Why the averaging is valid, and the scope of the proof

The functions $\mathsf{w}_B(z)$ and $e_B(z)$ are costs of hypothetical fresh calls with deterministic input prefix z. Their recurrences hold before any averaging. Integrating them against the target prefix laws uses only conditional consistency. It does not assert that prefixes visited by the actual random recursion have their target distribution.

For a random incoming prefix, condition on it first and then average (67) and (68) term by term. Thus fully averaged bounds on each $\mathfrak{R}_r$ suffice, exactly as in the transcript application in Section H; no expectation is interchanged with the maximum defining $\mathfrak{R}_\star$.

These arguments preserve the expected work and depth guarantees and the exactness of the scheduler. In particular, when $\mathfrak{R}_\star = \widetilde{O}(\sqrt{n})$ and h is logarithmic in the problem parameters, they give $\widetilde{O}(\sqrt{n})$ expected depth. Approximation error in the construction of the target path law is unaffected. Stronger high-probability complexity bounds require a separate tail argument.