THE OPTIMAL REFRESHMENT RATE OF THE BOUNCY PARTICLE SAMPLER: UNIQUENESS TO WITHIN 10^{-7}, A CERTIFIED CRITICAL POINT AT $1.423266366804207\ldots$, AND THE EVENT RATIO 0.7811

KOREA SUPERINTELLIGENCE LABS

ABSTRACT. For the Bouncy Particle Sampler on a standard Gaussian target in $\mathbb{R}^d$ with velocity refreshment rate ρ, Bierkens, Kamatani and Roberts proved that the log-density process, sped up by d, converges to an Ornstein–Uhlenbeck process whose diffusion coefficient $\sigma^2(\rho)$ is the Laplace transform at ρ of the autocovariance of an explicit one-dimensional piecewise deterministic process. They located the maximiser of σ^2 by Monte Carlo at "around 1.424" and derived from it the tuning rule that refreshments should make up 0.7812 of all events. Shiba and Kamatani have since obtained σ^2 in closed form, in terms of the Mills-ratio function $M(a) = e^{a^2/2}\int_a^\infty e^{-t^2/2}\,dt$, and located the maximiser numerically at $1.423\cdots$. Starting from that closed form we prove that σ^2 is strictly increasing on $[\frac{1}{2}, 1.4232663]$ and strictly decreasing on $[1.4232664, 2]$, that $\sigma^2(\rho) < \sigma^2(\rho^*)$ for $\rho \in [\frac{1}{10}, \frac{1}{2}]$ and for $\rho \geq 2$, and hence that σ^2 attains its maximum on $[\frac{1}{10}, \infty)$ and every maximiser lies in a window of width 10^{-7}; that $(\sigma^2)'$ changes sign inside $(1.423266366804207, 1.423266366804208)$; that $1.8383133642103960 < \sigma^2(1.423266366804207) < 1.8383133642103961$; and that the ratio of refreshment events to all events at every point of that bracket lies in $(0.781066629766458, 0.781066629766459)$. So the printed constants 1.424 and 0.7812 are both off in their fourth significant figure — to four figures the values are 1.423 and 0.7811 — and the tuning rule, named "the 78.12% rule" in the published version and targeted verbatim by an adaptive sampler of Bertazzi and Bierkens, should read 0.78107 rather than 0.7812. Literal uniqueness of the maximiser inside the 10^{-7} window and the range $(0, \frac{1}{10})$ are left open. Every statement is verified in Lean 4 against *Mathlib*: the numerical steps are exact rational interval enclosures (16,576 derivative boxes and 8,256 value boxes) checked by compiled evaluation, and the passage from an enclosure to a real inequality is a kernel proof.

1. INTRODUCTION

The Bouncy Particle Sampler (BPS) of Bouchard-Côté, Vollmer and Doucet [9] is a continuous-time, non-reversible Markov chain Monte Carlo method: a particle (x, v) moves in a straight line, bounces off the level sets of the target's log-density at a state-dependent rate, and at a constant rate $\rho > 0$ *refreshes* its velocity by drawing a fresh direction. Refreshment is needed for ergodicity but slows the exploration down when overused, so the choice of ρ is the tuning problem of the method, and the constant that answers it plays the role that the acceptance rate 0.234 of Roberts, Gelman and Gilks [8] plays for the random-walk Metropolis algorithm.

Bierkens, Kamatani and Roberts [1] settled the high-dimensional behaviour for the standard Gaussian target. For velocities on the sphere S^{d-1} and a fixed refreshment rate ρ, their Theorem 2.8 states that the normalised log-density process $Y_t^{B,d} = \sqrt{d}\,(\|\xi_{dt}^{B,d}\|^2/d - 1)$, sped up by the factor d, converges to the stationary Ornstein–Uhlenbeck process

$$dY_t = -\frac{\sigma(\rho)^2}{4}\, Y_t\, dt + \sigma(\rho)\, dW_t, \qquad \sigma(\rho)^2 := 8\int_0^\infty e^{-\rho s} K(s, 0)\, ds, \tag{1}$$

where $K(s,t) = \mathbb{E}[\mathcal{T}_s \mathcal{T}_t]$ (their equation (2.3)) is the autocovariance of the stationary process $\mathcal{T}$ on $\mathbb{R}$ with generator $Gf(x) = f'(x) + x^+(f(-x) - f(x))$ (their equation (2.2)) and invariant law $\mathcal{N}(0,1)$. Their Proposition 2.9 shows $\sigma(\rho)^2 \to 0$ as $\rho \downarrow 0$ and as $\rho \to \infty$, so a maximiser ρ^* exists; their Corollary 2.7 gives $T(\frac{1}{\sqrt{2\pi}} + \rho)$, independently of d, for the expected number of events of the process $\langle \xi^{B,d}, v^{B,d}\rangle$ on $[0, T]$, of which ρT are refreshments. The text between Proposition 2.9 and Theorem 2.10 then says, verbatim:

> "The covariance function $K(t,0)$ and the diffusion coefficients $\sigma(\rho)^2$ do not admit simple expressions. These functions can be written as infinite sums of convolutions, and numerical evaluation is difficult. On the other hand, simple Monte Carlo calculations yield good estimates of these functions (Figure 1). The Monte Carlo estimates also provide that the optimal choice of ρ^* is around 1.424. The ratio of the expected number of the refreshment jumps to that of overall jumps is $\rho^*/(\frac{1}{\sqrt{2\pi}}+\rho^*)\approx 0.7812$."

(Statement and equation numbers for [1] throughout are those of the arXiv e-print, version 1, compiled for this work and read from its `.aux` file; the published version renumbers, and the bibliography records the map.) Version 1 closes that paragraph with "However the above jump ratio does not depend on the scale."; version 2 and the published version continue it, the latter with "making the 78.12% rule a possible criterion for the choice of the refreshment rate", and the published Proposition 2.16 carries the same ratio to a class of non-i.i.d. targets. The criterion is therefore in print under a name built from the constant. The introduction of the same paper adds that the process "attains the optimal convergence rate when the ratio of the expected number of refreshment jumps to that of all jumps is approximately 0.7812" and that "this result may provide a practical criterion for selecting the refreshment rate." The criterion was taken up: Bertazzi and Bierkens [4] define an adaptive BPS whose refreshment rate is updated by the rule $\lambda_r^n/(\lambda_r^n+\bar\lambda_{\mathrm{refl}}(n))=0.7812$, i.e. $\lambda_r^n=(0.7812/0.2188)\,\bar\lambda_{\mathrm{refl}}(n)$, "using the optimality criterion in [1]"; and the 2026 survey [3] repeats both constants, with the figure caption "The maximum is attained at $\lambda_r=1.424$" and the sentence

> "The speed $\sigma^2(\lambda_r)$ admits a semi-analytical expression in terms of an integral, and depends crucially on the value of λ_r. It can be computed numerically that this speed is optimal for $\lambda_r=1.424$, and perhaps more importantly, the (scale invariant) ratio of expected number of refreshment jumps to overall jumps is 0.7812. The latter observation can be used more generally as a suitable criterion for tuning of the refreshment rate."

The closed form. The integral in (1) is not intractable. Shiba and Kamatani [2], Theorem 4.1, obtain both the BPS coefficient and that of the Forward Event-Chain Monte Carlo sampler in closed form by reading $\int_0^\infty e^{-\rho s}K(s,0)\,ds$ as a resolvent of the generator G and solving the resulting Poisson equations (their appendix attributes the BPS resolvent itself to Bierkens and Verduyn Lunel [5], Theorem 5.3). In the notation of this paper their formula reads

$$\sigma^2(\rho)=\frac{8}{\rho}\Big[1+\frac{1}{\rho^2}-\frac{4}{\rho\sqrt{2\pi}}-\frac{2\,N(\rho)^2}{\sqrt{2\pi}\,M(2\rho)}\Big],\qquad N(\rho)=1-\frac{\rho^2+1}{\rho}\,M(\rho),\qquad M(a)=e^{a^2/2}\int_a^\infty e^{-t^2/2}\,dt,\tag{2}$$

a rational expression in ρ, $\sqrt{2\pi}$ and the two values $M(\rho)$, $M(2\rho)$ of a Mills-ratio function (Section 2 records the translation). Their Remark 4.2 then reads: "Now that we have analytic expressions, we are able to numerically optimize σ_B^2, for example, using the algorithm of [Brent 1971]. This function attains its maximum value $\sigma_B^2(\rho^*)=1.838\cdots$ at $\rho^*=1.423\cdots$." No error bound and no uniqueness statement accompany this; the same paper does prove a monotonicity statement, its Corollary 4.4 "Monotonicity of σ_F^2", but for the Forward Event-Chain coefficient, whose maximum is at the boundary $\rho=0$, and it prints the refreshment ratio only through its complement, $\frac{1/\sqrt{2\pi}}{1/\sqrt{2\pi}+\rho^*}\approx 0.218\cdots$ (Remark 3.4). The survey [3] cites [2] twice, for the algorithm and for its scaling limit, while keeping the digits above.

What this paper proves. Everything below is about the function (2), which we take as the definition of σ^2; its identification with (1) is Theorem 4.1 of [2] and is not re-proved here (Remark 2.4 records an independent check of it). Write

$$\rho_{\mathrm{lo}}=1.423266366804207,\quad \rho_{\mathrm{hi}}=1.423266366804208,\quad w_{\mathrm{lo}}=1.4232663,\quad w_{\mathrm{hi}}=1.4232664.$$

- **The maximiser is unique to within** 10^{-7} (Theorems 6.1 and 6.2). σ^2 is strictly increasing on $[\frac12,w_{\mathrm{lo}}]$ and strictly decreasing on $[w_{\mathrm{hi}},2]$; $\sigma^2(\rho)<\sigma^2(\rho_{\mathrm{lo}})$ for $\rho\in[\frac{1}{10},\frac12]$ and for every $\rho\ge 2$. Hence σ^2 attains a maximum on $[\frac{1}{10},\infty)$, every maximiser lies in $[w_{\mathrm{lo}},w_{\mathrm{hi}}]$, and two maximisers differ by at most 10^{-7}. No source states a bound of this kind: [1] locate the optimum by Monte Carlo, [2] by Brent's method.

- **A certified bracket and certified digits** (Theorems 4.1–4.3). $(\sigma^2)'$ vanishes somewhere in the open interval $(\rho_{\mathrm{lo}}, \rho_{\mathrm{hi}})$ of width 10^{-15}; $1.8383133642103960 < \sigma^2(\rho_{\mathrm{lo}}) < 1.8383133642103961$; and for every $c \in [\rho_{\mathrm{lo}}, \rho_{\mathrm{hi}}]$ the event ratio $r(c) = c/(\frac{1}{\sqrt{2\pi}} + c)$ satisfies $0.781066629766458 < r(c) < 0.781066629766459$, so $0.7812 - r(c) > 10^{-4}$. These are certificates rather than new mathematics — the digits follow from (2) by ordinary numerics — but we found no source that proves digits beyond the four of [2], and none that prints the corrected ratio (Remark 7.4).
- **The printed constants** (Proposition 4.4). $\sigma^2(1.424) < \sigma^2(\rho_{\mathrm{lo}})$ and $(\sigma^2)'(1.424) < 0$: the value 1.424 is not a maximiser and lies strictly past the peak. To four significant figures the optimal rate is 1.423, not 1.424, and the ratio is 0.7811, not 0.7812; the printed 0.7812 is $r(1.424) = 0.78115\ldots$ rounded, so the second constant inherits the error of the first. We state plainly that the correction is small: the speed lost by running at 1.424 is $\sigma^2(\rho^*) - \sigma^2(1.424) = 1.61 \cdot 10^{-7}$, a relative loss of $8.8 \cdot 10^{-8}$, and the ratio targeted by the adaptive rule of [4] is off by $1.33 \cdot 10^{-4}$. What is corrected is the printed record and the tuning criterion, not the practice.
- **Elementary tail bounds** (Propositions 5.1 and 5.2). $\sigma^2(\rho) \le 8/\rho + 8/\rho^3 - 32/(\rho^2\sqrt{2\pi})$ for all $\rho > 0$, whence $\sigma^2(\rho) < \sigma^2(\rho_{\mathrm{lo}})$ for $\rho \ge 2$; these make the finite box sweeps sufficient.

Not proved, and stated as such in Section 7: that the maximiser is literally unique (the sweeps say nothing inside $[w_{\mathrm{lo}}, w_{\mathrm{hi}}]$; strict concavity there would settle it), and anything about $\rho \in (0, \frac{1}{10})$, where the closed form has a fourth-order cancellation at 0 that defeats interval arithmetic.

Method. All theorems are proved in Lean 4 against *Mathlib*; Section 8 gives the details and Appendix A the statements. The analytic input is small: $M'(a) = aM(a) - 1$, so σ^2 has an elementary derivative $(\sigma^2)'$, proved by the chain rule; $M > 0$; the ratio is increasing in ρ; Darboux's theorem; and the compactness of $[\frac{1}{10}, 2]$. The numerical input is a rational interval arithmetic with kernel-proved soundness: $\int_0^a e^{-t^2/2}dt$ by an alternating Taylor sandwich, e^r by Mathlib's `Real.exp_bound`, $\sqrt{2\pi}$ by a rational bracket of width 10^{-19}, and, for a box $[a_1, a_2]$ of the argument, $M(a) \in [e^{a_1^2/2}T(a_2), e^{a_2^2/2}T(a_1)]$ with $T(a) = \frac{\sqrt{2\pi}}{2} - \int_0^a e^{-t^2/2}dt$, because the first factor increases and the second decreases. The sign of $(\sigma^2)'$ is certified on 16,576 boxes covering $[\frac{1}{2}, w_{\mathrm{lo}}] \cup [w_{\mathrm{hi}}, 2]$ and the value $\sigma^2 < 1.8383133642103960$ on 8,256 boxes covering $[\frac{1}{10}, \frac{1}{2}]$; only rational comparisons are evaluated by compiled code, and every step from an enclosure to a statement about a real number is a kernel proof.

Literature closure. A full-text sweep of a local corpus of arXiv e-prints for "refreshment rate" and "bouncy particle", read for the digit strings 1.424, 0.7812 and for uniqueness or monotonicity language about the optimal rate, and the OpenAlex citation graph of [1], are reported in Remark 7.4. The nearest neighbours do not compete: Deligiannidis, Paulin, Bouchard-Côté and Doucet [6] send the refreshment rate to infinity with the dimension, obtaining randomised Hamiltonian Monte Carlo as the limit, so no finite optimal rate arises there, and Andrieu and Livingstone [7] order samplers by asymptotic variance rather than tuning ρ. We found no proof, in any source, that the maximiser of σ^2 is unique or that σ^2 is monotone on any interval.

2. Preliminaries

Throughout $\rho > 0$ is the refreshment rate (λ_r in [3]), and $\sqrt{2\pi}$ is written for $\sqrt{2\pi}$.

Definition 2.1. For $a \in \mathbb{R}$ put

$$M(a) = e^{a^2/2}\Big(\frac{\sqrt{2\pi}}{2} - \int_0^a e^{-t^2/2}\,dt\Big), \qquad \text{so that}$$
$$M(a) = e^{a^2/2}\int_a^\infty e^{-t^2/2}\,dt = \int_0^\infty e^{-ay-y^2/2}\,dy \qquad (a \ge 0),$$

the Mills ratio of the standard normal law at a (formal: `mills_eq_integral`); the elementary form is taken as the definition so that differentiation and the enclosures need no hypothesis on a. Then $M > 0$ on $[0, \infty)$ and

$$M'(a) = a\,M(a) - 1 \qquad (a \in \mathbb{R}) \tag{3}$$

(formal: `mills_pos`, `hasDerivAt_mills`). $M(0) = \sqrt{\pi/2}$ and $M(1) = 0.65567954\ldots$, the tabulated value $(1-\Phi(1))/\phi(1)$; both are among the controls of Section 8.

Definition 2.2. For $\rho > 0$ let $N(\rho) = 1 - \frac{\rho^2+1}{\rho} M(\rho)$,

$$G(\rho) = 1 + \frac{1}{\rho^2} - \frac{4}{\rho\sqrt{2\pi}} - \frac{2N(\rho)^2}{\sqrt{2\pi}\, M(2\rho)}, \qquad \sigma^2(\rho) = \frac{8}{\rho}\, G(\rho), \qquad r(\rho) = \frac{\rho}{\frac{1}{\sqrt{2\pi}} + \rho}.$$

σ^2 is (2); r is the ratio of the expected number of refreshment events to the expected number of all events of the process $\langle \xi^{B,d}, v^{B,d} \rangle$ on a time interval — a ratio that does not depend on d, by Corollary 2.7 of [1] — and is strictly increasing on $(0,\infty)$ (formal: `bpsRatio_strictMonoOn`).

Remark 2.3 (the translation to the form of [2]). Theorem 4.1 of [2] prints

$$\sigma_B^2(\rho) = \frac{8}{\rho^4}\Big(\rho^3 - \rho^2\sqrt{\tfrac{8}{\pi}} + \rho - \sqrt{\tfrac{8}{\pi}}\, \frac{\big((1+\rho^2)\Omega(\rho) - \rho^2\big)^2}{\Omega(2\rho)}\Big),$$
$$\Omega(\rho) = \sqrt{\tfrac{\pi}{2}}\, \rho \operatorname{erfcx}\Big(\frac{\rho}{\sqrt{2}}\Big) = \rho\, e^{\rho^2/2} \int_\rho^\infty e^{-t^2/2} dt.$$

Since $\Omega(a) = aM(a)$ and $\sqrt{8/\pi} = 4/\sqrt{2\pi}$, the first three terms give $\frac{8}{\rho}(1 - \frac{4}{\rho\sqrt{2\pi}} + \frac{1}{\rho^2})$ and the last, using $(1+\rho^2)\Omega(\rho) - \rho^2 = \rho\big((1+\rho^2)M(\rho) - \rho\big) = -\rho^2 N(\rho)$, gives $\frac{8}{\rho} \cdot \frac{2N(\rho)^2}{\sqrt{2\pi}\, M(2\rho)}$; so $\sigma_B^2 = \sigma^2$ identically. We also checked the two expressions numerically to 10^{-128} at $\rho = \frac{1}{2}, 1, 1.4232663668, 2, 3$.

Remark 2.4 (the closed form; a computation outside the formal development). The identification of (2) with the integral (1) is Shiba and Kamatani's theorem and we claim nothing about it. For the record, it was rederived independently for this work, before [2] was found, from the generator G alone: by stationarity and Fubini, $\int_0^\infty e^{-\rho s} K(s,0)\, ds = \int_{\mathbb{R}} \phi(x)\, x\, U(x)\, dx$ with $U(x) = \int_0^\infty e^{-\rho t} \mathbb{E}_x[\mathcal{T}_t]\, dt$, and U solves the resolvent equation $\rho U - U' - x^+(U(-x) - U(x)) = x$, a first-order linear equation that is explicit on $x \le 0$ and, once $U(-x)$ is known there, on $x > 0$; the growth bound $|\mathbb{E}_x[\mathcal{T}_t]| \le |x| + t$ fixes the constants. The result agrees with [2] as above, and it passed four checks before any of the formal work was written: the large-ρ expansion $\sigma^2(\rho) = 8/\rho - 32/(\sqrt{2\pi}\rho^2) + 8/\rho^3 - 128/(\sqrt{2\pi}\rho^8) + \cdots$ reproduces $K(0) = 1$, $\partial_t K(t,0)|_0 = -2\sqrt{2/\pi}$, $\partial_t^2 K(t,0)|_0 = 1$ of Proposition 2.13 of [1], and the residual after three terms at $\rho = 100$ is $-5.1005 \cdot 10^{-15}$ against the predicted $-5.1065 \cdot 10^{-15}$; the limits at 0 and ∞ are those of Proposition 2.9; an exact simulation of $\mathcal{T}$ (the next jump from $x \ge 0$ is at position $\sqrt{x^2 + 2E}$ with $E \sim \operatorname{Exp}(1)$; $4 \cdot 10^7$ paths per point) agrees with (2) within 0.4 standard errors at $\rho = \frac{1}{2}, 1, \rho^*, 2, 5$; and two unrelated evaluations, 220-digit decimal arithmetic and the rational interval arithmetic of Section 3, agree to every digit the latter resolves. Every numerical value in this paper was recomputed for the manuscript with a third, independently written implementation (Machin's π, the Taylor series for $\int_0^a e^{-t^2/2} dt$, the Laplace continued fraction for M at large argument, bisection on $(\sigma^2)'$) at 130 digits.

Lemma 2.5 (the derivative)**.** *σ^2 is differentiable on $(0,\infty)$ with*

$$(\sigma^2)'(\rho) = -\frac{8}{\rho^2}\, G(\rho) + \frac{8}{\rho}\, G'(\rho), \qquad G'(\rho) = -\frac{2}{\rho^3} + \frac{4}{\rho^2\sqrt{2\pi}} - \frac{2}{\sqrt{2\pi}}\, Q'(\rho),$$
$$Q'(\rho) = \frac{2N(\rho)N'(\rho)M(2\rho) - N(\rho)^2 \cdot 2\big(2\rho M(2\rho) - 1\big)}{M(2\rho)^2},$$
$$N'(\rho) = -\Big(1 - \frac{1}{\rho^2}\Big)M(\rho) - \frac{\rho^2+1}{\rho}\big(\rho M(\rho) - 1\big).$$

Proof. The chain and quotient rules applied to Definition 2.2 with (3), and $\frac{d}{d\rho} M(2\rho) = 2(2\rho M(2\rho) - 1)$; $M(2\rho) > 0$ by `mills_pos`. (Formal: `hasDerivAt_bpsSpeed`, kernel-checked with no numerical content.) We write $D(\rho)$ for the right-hand side, which is the function `bpsSpeedD` of the formal development. $\square$

Lemma 2.6 (Darboux bracket)**.** *If $0 < a \le b$, $D(a) > 0$ and $D(b) < 0$, then $D(c) = 0$ for some $c \in (a,b)$.*

Proof. Darboux's theorem for the derivative D of σ^2 on $[a,b]$ (Mathlib's `exists_hasDerivWithinAt_eq_of_lt_of_gt`); no continuity of D is used. (Formal: `exists_critical_point`.) ☐

3. The rational enclosures

The certified numerical facts rest on one enclosure machine, described here once so that the theorems can say exactly what was computed. A rational interval is a pair $[\ell,u] \subset \mathbb{Q}$ with the operations $+,-,\times$, reciprocal (when $0 \notin [\ell,u]$), a sharp square, and *outward rounding* to dyadic endpoints with denominator 2^{220}, inserted after each composite step to keep denominators bounded; each operation carries a kernel-proved membership lemma. Three analytic sandwiches feed it.

- $\int_0^a e^{-t^2/2}dt = \sum_{k\ge 0}(-1)^k a^{2k+1}/(2^k k!(2k+1))$ for $a \ge 0$, and an even number of terms underestimates while an odd number over-estimates (Mathlib's alternating bound for e^{-s}, integrated term by term); we use 40 and 41 terms.
- For $0 \le r \le 8$, $e^r \in \left[(P-\varepsilon)^8, (P+\varepsilon)^8\right]$ with $P = \sum_{k<16}(r/8)^k/k!$ and $\varepsilon = (r/8)^{16}\cdot 17/(16!\cdot 16)$, from Mathlib's `Real.exp_bound`; we use it at $r = a^2/2 \le 8$, i.e. $a \le 4$, which is why the sweeps stop at $\rho = 2$ ($M(2\rho)$ is needed).
- $2.5066282746310005024 \le \sqrt{2\pi} \le 2.5066282746310005025$.

At a rational point $a \ge 0$ these give $M(a) \in [\ell,u]$ (formal: `mem_millsItv`), and over a box $a \in [a_1,a_2]$ with $0 \le a_1$,

$$e^{a_1^2/2}\,T(a_2) \;\le\; M(a) \;\le\; e^{a_2^2/2}\,T(a_1), \qquad T(a) = \tfrac{\sqrt{2\pi}}{2} - \textstyle\int_0^a e^{-t^2/2}dt, \tag{4}$$

because $a \mapsto e^{a^2/2}$ is increasing and T is decreasing on $[0,\infty)$ (formal: `mem_millsBox`, from `Real.exp_le_exp` and the monotonicity of the Gaussian integral). Substituting these into Definition 2.2 and Lemma 2.5, term by term, yields interval-valued functions speedItv, speedDItv at a rational point and speedBox, speedDBox over a rational box, each with a soundness theorem of the form "if ρ lies in the box and the box satisfies eight decidable side conditions (positivity, $a \le 4$, nonnegative tails, nonzero denominators), then $\sigma^2(\rho)$, resp. $D(\rho)$, lies in the computed interval" (formal: `mem_speedItv`, `mem_speedDItv`, `mem_speedBox`, `mem_speedDBox`; all four are kernel-checked and contain no compiled evaluation). With 40 terms the enclosures near ρ^* have width $6\cdot 10^{-18}$ for σ^2 and $8\cdot 10^{-17}$ for D at a point; over a box of width w the width of the D-enclosure is about $181w$ at $\rho = 1.4$, $620w$ at $\rho = 1$ and $10^4 w$ at $\rho = \frac{1}{2}$, the growth reflecting the near-cancellation of $1/\rho^2$ against $2N^2/(\sqrt{2\pi}M(2\rho))$ that interval arithmetic cannot see.

A *sweep* is a list of runs (step, k): starting from a rational a, the run checks the k consecutive boxes $[a + i\cdot\text{step}, a + (i+1)\cdot\text{step}]$ and the next run starts where it ends. The theorem `sweepOK_sound` (kernel-checked, generic in the box predicate) states that if every box passes, then every real point of the covered interval satisfies the conclusion; the only compiled evaluations are the Boolean sweeps themselves, i.e. comparisons of rationals.

4. The critical point and the constants

Theorem 4.1 (a bracket of width 10^{-15})**.** *$D(c) = 0$ for some c in the open interval $(\rho_{\mathrm{lo}}, \rho_{\mathrm{hi}}) =$ (1.423266366804207, 1.423266366804208).*

Proof. The point enclosures give $D(\rho_{\mathrm{lo}}) \in [2.7824637865\cdot 10^{-16}, 3.5795597307\cdot 10^{-16}]$ and $D(\rho_{\mathrm{hi}}) \in [-3.2012557130\cdot 10^{-16}, -2.4041597688\cdot 10^{-16}]$ (the 130-digit values are $3.1904\cdot 10^{-16}$ and $-2.7933\cdot 10^{-16}$); Lemma 2.6. (Formal: `exists_critical_point_bracket`.) The bracket is the limit of the present settings: at width 10^{-15} the certified $|D|$ at the endpoints is only 3.5 times the enclosure width. A control (`control_wide_bracket`) runs the same argument at 1.4 and 1.5, where $D(1.4) = 1.43\cdot 10^{-2}$ and $D(1.5) = -4.24\cdot 10^{-2}$, so that the hypotheses of Lemma 2.6 are satisfiable far from the tight bracket. ☐

Theorem 4.2 (the optimal speed)**.** $1.8383133642103960 < \sigma^2(\rho_{\mathrm{lo}}) < 1.8383133642103961.$

Proof. The certified interval is $[1.83831336421039605211,\ 1.83831336421039605812]$, of width $6 \cdot 10^{-18}$; the 130-digit value is $1.8383133642103960557583 5\ldots$ (formal: `bpsSpeed_rhoLo_enclosure`). Two controls refute the neighbouring claims $\sigma^2(\rho_{\mathrm{lo}}) > 1.8384$ and $\sigma^2(\rho_{\mathrm{lo}}) < 1.8383$. □

The theorem is stated at ρ_{lo}, the endpoint of the bracket, because that is what the enclosure evaluates. Numerically $\sigma^2(\rho^*) - \sigma^2(\rho_{\mathrm{lo}}) = 8.5 \cdot 10^{-32}$ and

$$\sigma^2(\rho^*) = 1.8383133642103960557583524865051794719 2\ldots;$$

what is *proved* about the maximum value $\sigma^2_{\max}$ of σ^2 on $[\frac{1}{10}, \infty)$ is the lower bound $\sigma^2_{\max} \geq \sigma^2(\rho_{\mathrm{lo}}) > 1.8383133642103960$ (from Theorem 6.2), not an upper bound. Shiba and Kamatani print $1.838\cdots$; [1] print no value of σ^2 (their Figure 1 is a Monte Carlo plot).

Theorem 4.3 (the event ratio)**.** *For every $c \in [\rho_{\mathrm{lo}}, \rho_{\mathrm{hi}}]$,*

$$0.781066629766458 < r(c) < 0.781066629766459 \qquad \textit{and} \qquad 0.7812 - r(c) > 10^{-4}.$$

Proof. r is strictly increasing, so it suffices to enclose the two endpoints, and the enclosures are

$$r(\rho_{\mathrm{lo}}) \in [0.78106662976645874 17596,\ 0.78106662976645874 17666],$$
$$r(\rho_{\mathrm{hi}}) \in [0.78106662976645886 19069,\ 0.78106662976645886 19138],$$

both of width $7 \cdot 10^{-21}$ (formal: `bpsRatio_bracket`, `bpsRatio_printed_wrong`). The 130-digit value at ρ^* is

$$r(\rho^*) = 0.781066629766458805820622030057913387 09\ldots;$$

its complement $0.21893337\ldots$ is the $0.218\cdots$ of Remark 3.4 of [2]. □

Proposition 4.4 (the printed rate is not optimal)**.** *$\sigma^2(1.424) < \sigma^2(\rho_{\mathrm{lo}})$ and $D(1.424) < 0$.*

Proof. $\sigma^2(1.424) \in [1.83831320326551432167,\ 1.83831320326551432772]$ against Theorem 4.2: the gap $\sigma^2(\rho_{\mathrm{lo}}) - \sigma^2(1.424) = 1.6094 \cdot 10^{-7}$ is seven orders of magnitude above the enclosure widths. $D(1.424) \in [-4.38649039 \cdot 10^{-4}, -4.38649038 \cdot 10^{-4}]$. (Formal: `bpsSpeed_printed_lt_bpsSpeed_rhoLo`, `bpsSpeedD_printed_neg`.) That 1.424 is not a maximiser also follows from Theorem 6.1: it is outside the window $[w_{\mathrm{lo}}, w_{\mathrm{hi}}]$ (formal: `control_printed_not_in_window`). □

The proposition certifies what Remark 4.2 of [2] already implies by printing $\rho^* = 1.423\cdots$; we include it because the two printed constants are the objects the literature actually uses. For the record, $\rho^* = 1.4232663668042075331779479072979007101 7\ldots$ (bisection on D at 130 digits; the bracket of Theorem 4.1 contains it), $1.424 - \rho^* = 7.34 \cdot 10^{-4}$, $r(1.424) = 0.78115473830933306430\ldots$, which rounds to 0.7812, while $r(\rho^*)$ rounds to 0.7811; $0.7812 - r(\rho^*) = 1.334 \cdot 10^{-4}$, of which $0.88 \cdot 10^{-4}$ is $r(1.424) - r(\rho^*)$ and $0.45 \cdot 10^{-4}$ is the rounding. These decimal expansions are outside the formal development; the inequalities of Theorems 4.1–4.3 are what is proved.

5. Elementary tail bounds

Proposition 5.1. *For every $\rho > 0$, $\sigma^2(\rho) \leq 8/\rho + 8/\rho^3$. Consequently $\sigma^2(\rho) < \sigma^2(\rho_{\mathrm{lo}})$ for every $\rho \geq 5$.*

Proof. In Definition 2.2 the two subtracted terms of G are nonnegative, the second because $M(2\rho) > 0$; drop both. For $\rho \geq 5$ the bound is at most $8/5 + 8/125 = 1.664 < 1.8383133642103960 < \sigma^2(\rho_{\mathrm{lo}})$ by Theorem 4.2. (Formal: `bpsSpeed_le_tail_bound`, kernel-checked with no numerical content, and `bpsSpeed_lt_of_large`.) The bound first drops below $\sigma^2(\rho^*)$ at $\rho = 4.561\ldots$, so 5 is a convenient integer cut, not the sharp one. □

Proposition 5.2. *For every $\rho > 0$, $\sigma^2(\rho) \leq B(\rho) := 8/\rho + 8/\rho^3 - 32/(\rho^2\sqrt{2\pi})$. Consequently $\sigma^2(\rho) < \sigma^2(\rho_{\mathrm{lo}})$ for every $\rho \geq 2$.*

Proof. Drop only the last term of G. B is strictly decreasing on $(0, \infty)$: $B'(\rho) = \rho^{-4}(-8\rho^2 + 32\rho/\sqrt{2\pi} - 24)$, a quadratic of discriminant $(32/\sqrt{2\pi})^2 - 768 = -605.0\ldots < 0$; and $B(2) = 5 - 8/\sqrt{2\pi} = 1.80846\ldots < 1.8383133642103960$. In the formal proof the surd is removed by $32/\sqrt{2\pi} \geq 12.7661$ (from the rational upper bound for $\sqrt{2\pi}$), and the remaining inequality $1.8383133642103960\,\rho^3 - 8\rho^2 +$

$12.7661\rho - 8 > 0$ for $\rho \geq 2$ is a polynomial inequality in $u = \rho - 2 \geq 0$ with positive coefficients, closed by `nlinarith`. (Formal: `bpsSpeed_le_tail_bound2`, kernel-checked; `bpsSpeed_lt_of_ge_two`, whose only numerical input is Theorem 4.2.) B crosses $\sigma^2(\rho^*)$ at $\rho = 1.90668\ldots$, so 2 leaves a margin of 5%; the cut at 2 also keeps every box of Section 6 inside the range $a \leq 4$ of the exponential sandwich. ☐

6. Uniqueness to within 10^{-7}

Theorem 6.1 (monotonicity on the two flanks, and the window)**.** (i) *σ^2 is strictly increasing on $[\frac{1}{2}, w_{\text{lo}}] = [\frac{1}{2}, 1.4232663]$.*

(ii) *σ^2 is strictly decreasing on $[w_{\text{hi}}, 2] = [1.4232664, 2]$.*
(iii) *σ^2 attains a maximum on $[\frac{1}{2}, \infty)$.*
(iv) *Every maximiser of σ^2 on $[\frac{1}{2}, \infty)$ lies in $[w_{\text{lo}}, w_{\text{hi}}]$.*
(v) *Any two maximisers of σ^2 on $[\frac{1}{2}, \infty)$ differ by at most 10^{-7}.*

Proof. (i), (ii): by Mathlib's `strictMonoOn_of_deriv_pos` and `strictAntiOn_of_deriv_neg` it suffices that $D > 0$ on $[\frac{1}{2}, w_{\text{lo}}]$ and $D < 0$ on $[w_{\text{hi}}, 2]$, and that is the box sweep: $[\frac{1}{2}, 1.42]$ in 11 runs and 3,328 boxes, $[1.42, w_{\text{lo}}]$ in 14 runs and 4,224 boxes forming a dyadic ladder toward the optimum (the last run has 1,024 boxes of width $3.9 \cdot 10^{-10}$), $[w_{\text{hi}}, 1.43]$ in 14 runs and 7,424 boxes (the first run, $3.3 \cdot 10^{-8}$ from the optimum, needs 4,096 boxes of width $2.0 \cdot 10^{-10}$), and $[1.43, 2]$ in 5 runs and 1,600 boxes; 44 runs and 16,576 boxes in all, on each of which the lower (resp. upper) end of the D-enclosure is a positive (resp. negative) rational. The ladder is dyadic because near the optimum $|D(\rho)| \approx 0.60\,|\rho - \rho^*|$ while the enclosure width is $\approx 181w$, so a box at distance δ must have width $\lesssim 0.0033\,\delta$: about 256 boxes per octave, 14 octaves per side. (iii): σ^2 is continuous on $[\frac{1}{2}, 2]$, so it attains a maximum there, and by Proposition 5.2 no point $\rho > 2$ exceeds $\sigma^2(\rho_{\text{lo}})$, which is a value taken on $[\frac{1}{2}, 2]$. (iv): if a maximiser x had $x < w_{\text{lo}}$ then $\sigma^2(x) < \sigma^2(w_{\text{lo}})$ by (i); if $w_{\text{hi}} < x \leq 2$ then $\sigma^2(x) < \sigma^2(w_{\text{hi}})$ by (ii); if $x > 2$ then $\sigma^2(x) < \sigma^2(\rho_{\text{lo}})$ by Proposition 5.2. (v) is (iv) twice. (Formal: `bpsSpeed_strictMonoOn_left`, `bpsSpeed_strictAntiOn_right`, `exists_isMaxOn_Ici`, `maximiser_mem_window`, `maximiser_unique_to_1e7`.) Two controls show the box test is not vacuous: the single box $[1.4, 1.5]$, which straddles the optimum, fails the positivity test (`control_box_test_fails`), and the box $[1, 1.1]$ satisfies the side conditions (`control_boxOK_nonvacuous`). ☐

Below $\rho = \frac{1}{2}$ the derivative enclosure degrades like ρ^{-4} and a sign sweep would cost of the order of 10^5 boxes. The value enclosure has room instead: $\sigma^2(\frac{1}{2}) = 1.3215\ldots$ and $\sigma^2(\frac{1}{10}) = 0.3689\ldots$ against the optimum $1.8383\ldots$.

Theorem 6.2 (the range $[\frac{1}{10}, \infty)$)**.** *$\sigma^2(\rho) < \sigma^2(\rho_{\text{lo}})$ for every $\rho \in [\frac{1}{10}, \frac{1}{2}]$. Consequently:*

(i) *σ^2 attains a maximum on $[\frac{1}{10}, \infty)$;*
(ii) *every maximiser of σ^2 on $[\frac{1}{10}, \infty)$ lies in $[w_{\text{lo}}, w_{\text{hi}}]$;*
(iii) *any two maximisers of σ^2 on $[\frac{1}{10}, \infty)$ differ by at most 10^{-7}.*

Proof. The first sentence is a sweep of $[\frac{1}{10}, \frac{1}{2}]$ in 15 runs and 8,256 boxes (widths from $4.9 \cdot 10^{-6}$ at $\frac{1}{10}$ to $7.8 \cdot 10^{-4}$ at $\frac{1}{2}$), each certifying that the upper end of the σ^2-enclosure is below the rational 1.8383133642103960 of Theorem 4.2 (formal: `bpsSpeed_lt_on_low`); the single box $[\frac{1}{10}, \frac{1}{2}]$ fails this test (`control_low_box_fails`), and $\sigma^2(\frac{1}{4}) < \sigma^2(\rho_{\text{lo}})$ is recorded as an instance. (i)–(iii) then follow as in Theorem 6.1, with the compact interval $[\frac{1}{10}, 2]$ and the new sweep disposing of $[\frac{1}{10}, \frac{1}{2})$. (Formal: `exists_isMaxOn_Ici_low`, `maximiser_mem_window_low`, `maximiser_unique_to_1e7_low`.) ☐

Corollary 6.3 (on paper; not formalised)**.** *Let $\rho^\dagger$ be any maximiser of σ^2 on $[\frac{1}{10}, \infty)$. Then $1.4232663 \leq \rho^\dagger \leq 1.4232664$, so $\rho^\dagger$ rounds to 1.423 at four significant figures and to 1.4233 at five; $\sigma^2(\rho^\dagger) \geq \sigma^2(\rho_{\text{lo}}) > 1.8383133642103960$; and, r being increasing, $0.78106662174\ldots = r(w_{\text{lo}}) \leq r(\rho^\dagger) \leq r(w_{\text{hi}}) = 0.78106663375\ldots$, so the event ratio at the optimum rounds to 0.7811 at four decimals and to 0.781067 at six. The critical point of Theorem 4.1 lies in the same window (formal: `control_window_contains_bracket`).*

Proof. Theorem 6.2(ii), Theorem 4.2, and the monotonicity of r; the two values $r(w_{\mathrm{lo}})$, $r(w_{\mathrm{hi}})$ are decimal evaluations (also rational enclosures of width 10^{-20} from the machine of Section 3, run outside Lean). □

7. What is not proved, and what the sources print

Remark 7.1 (literal uniqueness). The sweeps certify $D \neq 0$ outside $[w_{\mathrm{lo}}, w_{\mathrm{hi}}]$ and say nothing inside it. Numerically D is strictly decreasing there — $D(w_{\mathrm{lo}}) = 3.997 \cdot 10^{-8}$, $D(w_{\mathrm{hi}}) = -1.986 \cdot 10^{-8}$, $(\sigma^2)''(\rho^*) = -0.598$ — so the critical point of Theorem 4.1 is the unique maximiser, but this is not proved. A certified enclosure of $(\sigma^2)''$ on $[1.42, 1.4234]$, of the same shape as the enclosure of D, would give strict concavity there and hence literal uniqueness together with the 10^{-15} localisation; alternatively the dyadic ladder could be continued for another 26 octaves per side ($\approx$ 13,000 boxes, if the point enclosures held their present margin) to reach the bracket, which would still not give literal uniqueness. Neither is done here.

Remark 7.2 (the range $(0, \frac{1}{10})$). On $(0, \frac{1}{10})$ the value test of Theorem 6.2 already fails on $[0.02, 0.05]$ and $[0.05, 0.1]$ at 8,192 boxes per segment. The reason is structural: $\sigma^2(\rho) = 8P(\rho)/\rho^3$ with P vanishing to fourth order at 0 ($\sigma^2(\rho)/\rho \to 4$; at 130 digits $\sigma^2(10^{-3})/10^{-3} = 3.99681\ldots$), and interval arithmetic loses the cancellation. A polynomial sandwich for M near 0 with $\sqrt{2\pi}$ kept symbolic would reduce the range to a polynomial inequality; we have not carried it out. Numerically D has exactly one sign change on 401 logarithmically spaced points of $[10^{-3}, 2\cdot 10^2]$, at the grid step containing ρ^*, with $D(10^{-3}) = 3.9936$ and $D(200) = -1.968 \cdot 10^{-4}$; together with Proposition 2.9 of [1] this is the evidence, not a proof, that the maximiser is unique on all of $(0, \infty)$.

Remark 7.3 (the identification with the sampler). Every theorem above is about the closed form (2). Its reading as the diffusion coefficient of the BPS log-density limit rests on Theorem 4.1 of [2] together with Theorem 2.8 of [1], neither of which is formalised: Mathlib has no theory of piecewise deterministic Markov processes, and formalising the resolvent argument of Remark 2.4 would require the process $\mathcal{T}$, its generator, the invariance of $\mathcal{N}(0,1)$ and Fubini's theorem for the Laplace transform.

Remark 7.4 (what the sources print, and what follows). We summarise, without adjectives, the state of the printed record.

- [1] print ρ^* "around 1.424" and the ratio "$\approx$ 0.7812", both from Monte Carlo, and say the coefficient has no simple expression. This is so in all three texts we read — arXiv versions 1 and 2 and the published one, which in that passage says that "the value of ρ maximising $\sigma(\rho)^2$" is around 1.424 where the e-prints say "the optimal choice of ρ^* is around 1.424". By Theorems 4.1 and 6.2, $\rho^* = 1.4232663\ldots$, which rounds to 1.423; by Theorem 4.3 and Corollary 6.3 the ratio is $0.78107\ldots$, which rounds to 0.7811. The printed 0.7812 is $r(1.424)$ rounded. Version 2 and the published version, but not version 1, name the criterion "the 78.12% rule"; the published version also adds a remark on the computational-effort target (Remark 2.12) and a proposition carrying the ratio to non-i.i.d. targets (Proposition 2.16). Neither addition bears on the value of ρ^*: both take the maximiser of σ^2 as given.
- [2] print the closed form, $\rho^* = 1.423\cdots$ and $\sigma_B^2(\rho^*) = 1.838\cdots$ from Brent's method, and the complement $0.218\cdots$ of the ratio; no bound, no uniqueness, and a monotonicity result for the other sampler only (Corollary 4.4). Everything in Sections 4–6 is consistent with their digits.
- [3] repeats 1.424 without the hedge "around" and 0.7812, describes the coefficient as admitting "a semi-analytical expression in terms of an integral" that "can be computed numerically", and cites [2] at two other places; no such expression is printed in [1], whose estimate is Monte Carlo, and the closed form is Theorem 4.1 of the paper the survey cites. Its "Open Problems" section, in the announced version, is an unwritten placeholder, so it gives no evidence either way on whether the refreshment-rate question was regarded as closed.
- [4] build their adaptive refreshment scheme on the target ratio 0.7812 (in their section "The adaptive schemes", update rule $\lambda_r^n = (0.7812/0.2188)\bar{\lambda}_{\mathrm{refl}}(n)$; they also note that the criterion "assumes a standard Gaussian target"). The corrected target is 0.78107, i.e. $\lambda_r^n =$

$(0.78107/0.21893)\bar{\lambda}_{\text{refl}}(n)$; the change in the multiplier is from 3.5704 to 3.5677, one part in 1300.

- The full-text sweep of Section 1, run for this manuscript over 376 corpus shards (116 e-prints contain "bouncy particle", 39 contain "refreshment rate", 124 in all): the digit string 1.424 occurs among them only in [1] and [3], and 0.7812 only in those two and [4]; the strings 1.4232 and .7810 occur in none; "Mills" occurs only in [2]. Of the 39 papers containing "refreshment rate", nine contain uniqueness or monotonicity vocabulary, none about the BPS optimum: "unimodal" describes the target ([5], [4] and two others), one concerns the optimal integration time of a different sampler, and the rest are monotonicity of unrelated auxiliary functions. The OpenAlex citation graph of [1] (20 citing works at the time of writing) contains, apart from [4] and [6], no work on tuning the refreshment rate, and a search for "refreshment rate" with "bouncy particle" and "optimal" returns nothing further.

Remark 7.5 (size of the correction). Running the BPS at $\rho = 1.424$ instead of ρ^* costs $1.6 \cdot 10^{-7}$ in σ^2, a relative loss of $8.8 \cdot 10^{-8}$, far below any practical resolution, and the adaptive target of [4] moves by $1.3 \cdot 10^{-4}$. What changes is that the two constants of the tuning rule are now proved rather than estimated, and that the maximiser is known to be unique to within 10^{-7} on $[\frac{1}{10}, \infty)$.

8. Verification

Theorems 4.1, 4.2, 4.3, 6.1, 6.2, Propositions 4.4, 5.1, 5.2 and Lemmas 2.5, 2.6 rest on declarations formally verified in Lean 4 against *Mathlib* [10, 11] (Lean 4.32.0, Mathlib at its `v4.32.0` tag), whose statements are reproduced verbatim in Appendix A. Where a printed statement says more than its Lean counterpart, the text names the extra step: the decimal expansions of ρ^*, $\sigma^2(\rho^*)$, $r(\rho^*)$ and the gaps quoted after Proposition 4.4, the crossing points $4.561\ldots$ and $1.90668\ldots$ in Section 5, the enclosure-width and box-count heuristics in the proof of Theorem 6.1, Corollary 6.3, and the whole of Section 7 except the two cited controls. The development is six modules, 1,923 lines in all. *Mills* (197 lines) defines M, proves `mills_eq_integral`, `hasDerivAt_mills`, `mills_pos` and the point enclosures; *Speed* (200) defines σ^2, N, r, the enclosures speedItv, ratioItv with their soundness, and `bpsRatio_strictMonoOn`; *Deriv* (322) defines D, proves `hasDerivAt_bpsSpeed` and `exists_critical_point` (Darboux) and the enclosure of D at a point; *Facts* (305) holds Theorems 4.1–4.3, Propositions 4.4 and 5.1 and the controls; *Unique* (727) holds the box enclosures (4), the generic sweep with `sweepOK_sound`, the 44 run lists, Theorem 6.1 and Proposition 5.2; *Low* (172) holds the value sweep and Theorem 6.2. The run lists were found by an adaptive march in a Python mirror of the interval arithmetic, in exact fractions, and are re-verified in Lean; for this manuscript the mirror was rewritten from the Lean definitions and a sample of boxes from every run, the five run totals, the thin-box values at ρ_{lo} and ρ_{hi} (which reproduce the point enclosures digit for digit) and the two failing controls were re-evaluated.

The first three modules contain no compiled evaluation at all; every step from an interval to a statement about a real number is a kernel proof. Compiled evaluation (`native_decide`) is used only on decidable rational arithmetic: in *Facts* on the side conditions and the endpoint comparisons of the point enclosures, in *Unique* and *Low* on the Boolean sweeps. Accordingly the axiom print (`#print axioms`) is `propext`, `Classical.choice`, `Quot.sound` for `hasDerivAt_mills`, `mills_pos`, `mills_eq_integral`, `hasDerivAt_bpsSpeed`, `exists_critical_point`, `bpsRatio_strictMonoOn`, `mem_speedItv`, `mem_speedDItv`, `mem_millsBox`, `mem_speedDBox`, `mem_speedBox`, `sweepOK_sound`, `bpsSpeed_le_tail_bound`, `bpsSpeed_le_tail_bound2`, `control_mills_zero`, `control_window_contains_bracket` and `control_printed_not_in_window`; and, for each theorem carrying numerical content, those three plus the compiled-evaluation axioms it inherits, named after the declaration that evaluates: `bpsSpeed_printed_lt_bpsSpeed_rhoLo` and `bpsSpeedD_printed_neg` carry 9 and 6; `exists_critical_point_bracket` 12; `bpsSpeed_rhoLo_enclosure` 6; `bpsRatio_bracket` and `bpsRatio_printed_wrong` 6; `bpsSpeed_lt_of_large`, `bpsSpeed_lt_of_ge_two`, `exists_isMaxOn_Ici` and `exists_isMaxOn_Ici_low` 6 (all inherited from the enclosure of $\sigma^2(\rho_{\text{lo}})$); `bpsSpeed_strictMonoOn_left` and `bpsSpeed_strictAntiOn_right` 4 each (two per sweep block); `maximiser_mem_window` and `maximiser_unique_to_1e7` 14; `bpsSpeed_lt_on_low` 8; `maximiser_mem_window_low` and `maximiser_unique_to_1e7_low` 16. No declaration depends on `sorryAx`, and no axiom is declared by hand. With *Mathlib* loaded, *Mills*, *Speed*, *Deriv* and *Facts* elaborate in about

10–12 seconds each, *Unique* in 3 minutes 51 seconds with a peak resident set of 8.9 GB, and *Low* in 1 minute 43 seconds; the whole of the work, including the exploratory marches, the 130-digit and 220-digit evaluations and the Monte Carlo check of Remark 2.4, took about 90 processor-minutes. The development is currently in a private repository: repository reference to be added at submission.

Appendix A. The formal statements

The Lean statements corresponding to the results above, verbatim (statements only; proofs, docstrings and attributes omitted), in the namespace `LeanProblemSpec.BpsLambdaOpt`. `gaussHalf u` is $\int_0^u e^{-y^2/2}dy$, `Itv` a rational interval with ends `lo`, `hi`, `x ∈ᵢ I` the membership `I.lo` $\leq x \leq$ `I.hi` as reals, `(q : ℚ) : ℝ` the cast of a rational, `Set.Icc`/`Ioo`/`Ici` the closed, open and right-unbounded intervals, `IsMaxOn f s x` "x maximises f on s", and `StrictMonoOn`/`StrictAntiOn` strict monotonicity on a set. `mills` is M, `bpsSpeed` is σ^2, `bpsSpeedD` is D, `bpsRatio` is r, and `rhoLo`, `rhoHi`, `rhoPrinted` are ρ_{lo}, ρ_{hi}, 1.424.

Definitions.

```
noncomputable def mills (a : ℝ) : ℝ :=
  Real.exp (a ^ 2 / 2) * (Real.sqrt (2 * Real.pi) / 2 - gaussHalf a)

noncomputable def bpsSpeed (r : ℝ) : ℝ :=
  8 / r * (1 + 1 / (r * r) - 4 / r / Real.sqrt (2 * Real.pi)
    - 2 * ((1 - (r * r + 1) / r * mills r) * (1 - (r * r + 1) / r * mills r))
        / (Real.sqrt (2 * Real.pi) * mills (2 * r)))

noncomputable def bpsN (r : ℝ) : ℝ := 1 - (r * r + 1) / r * mills r

noncomputable def bpsRatio (r : ℝ) : ℝ := r / (1 / Real.sqrt (2 * Real.pi) + r)

noncomputable def bpsND (r : ℝ) : ℝ :=
  -((1 - 1 / (r * r)) * mills r) - (r * r + 1) / r * (r * mills r - 1)

noncomputable def bpsG (r : ℝ) : ℝ :=
  1 + 1 / (r * r) - 4 / r / Real.sqrt (2 * Real.pi)
    - 2 * (bpsN r * bpsN r) / (Real.sqrt (2 * Real.pi) * mills (2 * r))

noncomputable def bpsQD (r : ℝ) : ℝ :=
  (2 * bpsN r * bpsND r * mills (2 * r)
      - bpsN r * bpsN r * (2 * (2 * r * mills (2 * r) - 1)))
    / (mills (2 * r) * mills (2 * r))

noncomputable def bpsGD (r : ℝ) : ℝ :=
  -(2 / (r * r * r)) + 4 / (r * r) / Real.sqrt (2 * Real.pi)
    - 2 * bpsQD r / Real.sqrt (2 * Real.pi)

noncomputable def bpsSpeedD (r : ℝ) : ℝ := -(8 / (r * r)) * bpsG r + 8 / r * bpsGD r

def rhoLo : ℚ := 1423266366804207 / 1000000000000000
def rhoHi : ℚ := 1423266366804208 / 1000000000000000
def rhoPrinted : ℚ := 1424 / 1000
```

Analytic lemmas (Definition 2.1, Lemmas 2.5, 2.6).

```
lemma mills_eq_integral {a : ℝ} (ha : 0 ≤ a) :
    mills a = Real.exp (a ^ 2 / 2) * ∫ t in Ioi a, Real.exp (-t ^ 2 / 2)

lemma mills_pos {a : ℝ} (ha : 0 ≤ a) : 0 < mills a

lemma hasDerivAt_mills (a : ℝ) : HasDerivAt mills (a * mills a - 1) a
```

```
lemma hasDerivAt_bpsSpeed {r : ℝ} (hr : 0 < r) : HasDerivAt bpsSpeed (bpsSpeedD r) r

theorem exists_critical_point {a b : ℝ} (ha : 0 < a) (hab : a ≤ b)
    (hpos : 0 < bpsSpeedD a) (hneg : bpsSpeedD b < 0) :
    ∃ c ∈ Set.Ioo a b, bpsSpeedD c = 0

lemma bpsRatio_strictMonoOn : StrictMonoOn bpsRatio (Set.Ioi 0)
```

The enclosures (Section 3).

```
lemma mem_millsBox {n : ℕ} (hn : Even n) {A : Itv} (h0 : 0 ≤ A.lo)
    (h8 : A.hi * A.hi / 2 ≤ 8) (hlo : 0 ≤ tailLoQ (n + 1) A.hi)
    {x : ℝ} (hx : x ∈ᵢ A) : mills x ∈ᵢ millsBox n A

theorem mem_speedDBox {n : ℕ} (hn : Even n) {A : Itv} (hA : BoxOK n A)
    {x : ℝ} (hx : x ∈ᵢ A) : bpsSpeedD x ∈ᵢ speedDBox n A

theorem mem_speedBox {n : ℕ} (hn : Even n) {A : Itv} (hA : BoxOK n A)
    {x : ℝ} (hx : x ∈ᵢ A) : bpsSpeed x ∈ᵢ speedBox n A

lemma sweepOK_sound {n : ℕ} {P : Itv → Bool} {Q : ℝ → Prop}
    (hP : ∀ A : Itv, BoxOK n A → P A = true → ∀ x : ℝ, x ∈ᵢ A → Q x) :
    ∀ (L : List (ℚ × ℕ)) (a : ℚ), sweepOK n P a L = true →
      ∀ x : ℝ, (a : ℝ) ≤ x → x ≤ ((endPt a L : ℚ) : ℝ) → Q x

def posP (n : ℕ) (A : Itv) : Bool := decide (0 < (speedDBox n A).lo)
def negP (n : ℕ) (A : Itv) : Bool := decide ((speedDBox n A).hi < 0)
def ltP (n : ℕ) (A : Itv) : Bool :=
  decide ((speedBox n A).hi < 18383133642103960/10000000000000000)
```

Theorems 4.1–4.3 and Proposition 4.4.

```
theorem exists_critical_point_bracket :
    ∃ c ∈ Set.Ioo ((rhoLo : ℚ) : ℝ) ((rhoHi : ℚ) : ℝ), bpsSpeedD c = 0

theorem bpsSpeed_rhoLo_enclosure :
    (18383133642103960 / 10000000000000000 : ℝ) < bpsSpeed ((rhoLo : ℚ) : ℝ) ∧
      bpsSpeed ((rhoLo : ℚ) : ℝ) < (18383133642103961 / 10000000000000000 : ℝ)

theorem bpsRatio_bracket {c : ℝ} (hc : c ∈ Set.Icc ((rhoLo : ℚ) : ℝ) ((rhoHi : ℚ) : ℝ)) :
    (781066629766458 / 1000000000000000 : ℝ) < bpsRatio c ∧
      bpsRatio c < (781066629766459 / 1000000000000000 : ℝ)

theorem bpsRatio_printed_wrong {c : ℝ}
    (hc : c ∈ Set.Icc ((rhoLo : ℚ) : ℝ) ((rhoHi : ℚ) : ℝ)) :
    (1 / 10000 : ℝ) < (7812 / 10000 : ℝ) - bpsRatio c

theorem bpsSpeed_printed_lt_bpsSpeed_rhoLo :
    bpsSpeed ((rhoPrinted : ℚ) : ℝ) < bpsSpeed ((rhoLo : ℚ) : ℝ)

theorem bpsSpeedD_printed_neg : bpsSpeedD ((rhoPrinted : ℚ) : ℝ) < 0
```

Propositions 5.1 and 5.2.

```
theorem bpsSpeed_le_tail_bound {r : ℝ} (hr : 0 < r) :
    bpsSpeed r ≤ 8 / r + 8 / (r * r * r)

theorem bpsSpeed_lt_of_large {r : ℝ} (hr : 5 ≤ r) : bpsSpeed r < bpsSpeed ((rhoLo : ℚ) : ℝ)

theorem bpsSpeed_le_tail_bound2 {r : ℝ} (hr : 0 < r) :
    bpsSpeed r ≤ 8 / r + 8 / (r * r * r) - 32 / (r * r * Real.sqrt (2 * Real.pi))
```

```
theorem bpsSpeed_lt_of_ge_two {r : ℝ} (hr : 2 ≤ r) : bpsSpeed r < bpsSpeed ((rhoLo : ℚ) : ℝ)
```

Theorem 6.1.

```
theorem speedD_pos_left {x : ℝ} (h1 : (1/2 : ℝ) ≤ x) (h2 : x ≤ (14232663/10000000 : ℝ)) :
    0 < bpsSpeedD x

theorem speedD_neg_right {x : ℝ} (h1 : (14232664/10000000 : ℝ) ≤ x) (h2 : x ≤ (2 : ℝ)) :
    bpsSpeedD x < 0

theorem bpsSpeed_strictMonoOn_left :
    StrictMonoOn bpsSpeed (Set.Icc (1/2 : ℝ) (14232663/10000000))

theorem bpsSpeed_strictAntiOn_right :
    StrictAntiOn bpsSpeed (Set.Icc (14232664/10000000 : ℝ) 2)

theorem exists_isMaxOn_Ici :
    ∃ x ∈ Set.Ici (1/2 : ℝ), IsMaxOn bpsSpeed (Set.Ici (1/2 : ℝ)) x

theorem maximiser_mem_window {x : ℝ} (hx : x ∈ Set.Ici (1/2 : ℝ))
    (hmax : IsMaxOn bpsSpeed (Set.Ici (1/2 : ℝ)) x) :
    x ∈ Set.Icc (14232663/10000000 : ℝ) (14232664/10000000 : ℝ)

theorem maximiser_unique_to_1e7 {x y : ℝ} (hx : x ∈ Set.Ici (1/2 : ℝ))
    (hy : y ∈ Set.Ici (1/2 : ℝ)) (hmx : IsMaxOn bpsSpeed (Set.Ici (1/2 : ℝ)) x)
    (hmy : IsMaxOn bpsSpeed (Set.Ici (1/2 : ℝ)) y) : |x - y| ≤ (1/10000000 : ℝ)
```

Theorem 6.2.

```
theorem bpsSpeed_lt_on_low {x : ℝ} (h1 : (1/10 : ℝ) ≤ x) (h2 : x ≤ (1/2 : ℝ)) :
    bpsSpeed x < bpsSpeed ((rhoLo : ℚ) : ℝ)

theorem exists_isMaxOn_Ici_low :
    ∃ x ∈ Set.Ici (1/10 : ℝ), IsMaxOn bpsSpeed (Set.Ici (1/10 : ℝ)) x

theorem maximiser_mem_window_low {x : ℝ} (hx : x ∈ Set.Ici (1/10 : ℝ))
    (hmax : IsMaxOn bpsSpeed (Set.Ici (1/10 : ℝ)) x) :
    x ∈ Set.Icc (14232663/10000000 : ℝ) (14232664/10000000 : ℝ)

theorem maximiser_unique_to_1e7_low {x y : ℝ} (hx : x ∈ Set.Ici (1/10 : ℝ))
    (hy : y ∈ Set.Ici (1/10 : ℝ)) (hmx : IsMaxOn bpsSpeed (Set.Ici (1/10 : ℝ)) x)
    (hmy : IsMaxOn bpsSpeed (Set.Ici (1/10 : ℝ)) y) : |x - y| ≤ (1/10000000 : ℝ)
```

Controls.

```
theorem control_speed_not_above : ¬ ((1.8384 : ℝ) < bpsSpeed ((rhoLo : ℚ) : ℝ))

theorem control_speed_not_below : ¬ (bpsSpeed ((rhoLo : ℚ) : ℝ) < (1.8383 : ℝ))

theorem control_mills_zero : mills 0 = Real.sqrt (2 * Real.pi) / 2

theorem control_mills_one :
    (65567 / 100000 : ℝ) < mills 1 ∧ mills 1 < (65569 / 100000 : ℝ)

theorem control_bracket_nonempty : ((rhoLo : ℚ) : ℝ) < ((rhoHi : ℚ) : ℝ)

theorem control_wide_bracket :
    ∃ c ∈ Set.Ioo (1.4 : ℝ) (1.5 : ℝ), bpsSpeedD c = 0

theorem control_window_contains_bracket :
```

```
    (14232663/10000000 : ℝ) < ((rhoLo : ℚ) : ℝ) ∧
      ((rhoHi : ℚ) : ℝ) < (14232664/10000000 : ℝ)

theorem control_printed_not_in_window :
    ((rhoPrinted : ℚ) : ℝ) ∉ Set.Icc (14232663/10000000 : ℝ) (14232664/10000000 : ℝ)

theorem control_speedD_neg_at_1_5 : bpsSpeedD (3/2 : ℝ) < 0

theorem control_speedD_pos_at_1 : 0 < bpsSpeedD (1 : ℝ)

theorem control_box_test_fails : posP 40 (Itv.ofEnds (14/10) (15/10)) = false

theorem control_boxOK_nonvacuous : BoxOK 40 (Itv.ofEnds (1 : ℚ) (11/10))

theorem control_low_at_quarter : bpsSpeed (1/4 : ℝ) < bpsSpeed ((rhoLo : ℚ) : ℝ)

theorem control_low_box_fails : ltP 40 (Itv.ofEnds (1/10 : ℚ) (1/2)) = false
```

References

[1] J. Bierkens, K. Kamatani and G. O. Roberts, *High-dimensional scaling limits of piecewise deterministic sampling algorithms*, Ann. Appl. Probab. **32** (2022), no. 5, 3361–3407, DOI 10.1214/21-AAP1762; arXiv:1807.11358 [math.PR] (v1 30 July 2018, v2 30 July 2019; no later version). Journal, volume and issue checked against Crossref; the page range is read off the first page of the published PDF. Three texts were read for this work. (i) The e-print source of the local mirror, compiled here; all statement and equation numbers quoted above — equations (2.2), (2.3), Corollary 2.7, Theorem 2.8, Proposition 2.9, Theorem 2.10, Proposition 2.13, Figure 1 — are read from its `.aux` file, and the quotations of Section 1 are verbatim from it. That source is *version 1*: it lacks both the phrase "78.12% rule" and the proposition on non-i.i.d. targets that version 2 contains, although the corpus sidecar carries the 2019 date. (ii) The version 2 PDF from arXiv, which has the same numbering for the four statements above except that the covariance proposition is 2.14 there, and which adds the "78.12% rule" phrase and Proposition 2.11. (iii) The published PDF from Project Euclid, whose header carries the volume, issue and page range above. Its numbering differs from both e-prints: the switch-count corollary is Corollary 2.9, the Ornstein–Uhlenbeck limit Theorem 2.10, the vanishing of σ^2 at both ends Proposition 2.11, the covariance properties Proposition C.3, and the displayed ratio equation (2.8) (equations (2.2), (2.3) of the e-prints are (2.4), (2.5) there). This is why [2] cite the switch count as "Corollary 2.9" and the limit as "Theorem 2.10": they cite the published version. We keep the e-print numbers above because that is the text whose `.aux` we read.

[2] H. Shiba and K. Kamatani, *Diffusive scaling limits of forward event-chain Monte Carlo: provably efficient exploration with partial refreshment*, arXiv:2602.17087v1 [math.PR] (submitted 19 February 2026; "43 pages, 5 figures", one version, no journal reference). Read in full from the e-print source; the numbers Theorem 4.1 (equation (4.5)), Remark 4.2, Corollary 4.4 and Remark 3.4 are read from the `.aux` file of that source compiled for this work; the Brent 1971 reference in Remark 4.2 is R. P. Brent, Comput. J. **14** (1971), no. 4, 422–425 (Crossref). Its Appendix D contains the sentence "For L^B, its resolvent operator $(\rho - L^B)^{-1}$ is determined in [[5], Theorem 5.3]."

[3] J. Bierkens, *Scaling results for piecewise deterministic Monte Carlo: a survey*, arXiv:2607.22449v1 [stat.CO] (submitted 24 July 2026; one version, no journal reference, no comments field, primary category stat.CO, all from the arXiv abstract page). Read in full from the e-print source, compiled here; the two quotations in Section 1 are verbatim from its scaling section and from the caption of its Figure 1, the plot of $\sigma^2(\lambda_r)$. Its theorem on the BPS limits carries no label in the source; in the compiled document it is Theorem 5.1, whose part (i)(b) is the BPS log-density limit.

[4] A. Bertazzi and J. Bierkens, *Adaptive schemes for piecewise deterministic Monte Carlo algorithms*, Bernoulli **28** (2022), no. 4, 2404–2430, DOI 10.3150/21-BEJ1423; arXiv:2012.13924. Journal, volume, issue and DOI checked against Crossref, the page range against the Project Euclid article page and the zbMATH record (Zbl 1501.65001). The update rule and the two quoted phrases are verbatim from the arXiv e-print.

[5] J. Bierkens and S. M. Verduyn Lunel, *Spectral analysis of the zigzag process*, Ann. Inst. Henri Poincaré Probab. Stat. **58** (2022), no. 2, 827–860, DOI 10.1214/21-AIHP1188. Journal, volume, issue and DOI checked against Crossref, the page range against the Project Euclid article page and zbMATH (Zbl 1519.37005); not read for this work, cited because [2] attributes the BPS resolvent to its Theorem 5.3.

[6] G. Deligiannidis, D. Paulin, A. Bouchard-Côté and A. Doucet, *Randomized Hamiltonian Monte Carlo as scaling limit of the bouncy particle sampler and dimension-free convergence rates*, Ann. Appl. Probab. **31** (2021), no. 6, 2612–2662, DOI 10.1214/20-AAP1659; arXiv:1808.04299. Journal, volume, issue and DOI checked against Crossref, the page range against the Project Euclid article page and zbMATH (Zbl 1484.65004); not read in full.

[7] C. Andrieu and S. Livingstone, *Peskun–Tierney ordering for Markovian Monte Carlo: beyond the reversible scenario*, Ann. Statist. **49** (2021), no. 4, 1958–1981, DOI 10.1214/20-AOS2008. Checked against Crossref, the page range against the Project Euclid article page and zbMATH (Zbl 1489.65005); not read in full.

[8] G. O. Roberts, A. Gelman and W. R. Gilks, *Weak convergence and optimal scaling of random walk Metropolis algorithms*, Ann. Appl. Probab. **7** (1997), no. 1, 110–120, DOI 10.1214/aoap/1034625254. Checked against Crossref, the page range against the Project Euclid article page and zbMATH (Zbl 0876.60015). Cited by [1] as the model for a tuning criterion.

[9] A. Bouchard-Côté, S. J. Vollmer and A. Doucet, *The bouncy particle sampler: a nonreversible rejection-free Markov chain Monte Carlo method*, J. Amer. Statist. Assoc. **113** (2018), no. 522, 855–867, DOI 10.1080/01621459.2017.1294075. Checked against Crossref; not read for this work.

[10] L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, in: Automated Deduction – CADE 28, Lecture Notes in Computer Science **12699**, Springer, 2021, pp. 625–635.

[11] The mathlib Community, *The Lean mathematical library*, in: Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, pp. 367–381.