Stochastic Bouncy Particle Sampler

Supplementary Material

A. Proof Sketch of Invariance under Noisy Gradients

In this section we start with a simple reformulation of the proof in (Bouchard-Côté et al., 2015) that the BPS Markov process leaves invariant the distribution $p(\mathbf{w}, \mathbf{v}) = p(\mathbf{w})p(\mathbf{v})$ where

$$p(\mathbf{w}) \propto e^{-U(\mathbf{w})}, \qquad \mathbf{w} \in \mathbb{R}^D, \tag{A.1}$$

$$p(\mathbf{v}) = \text{Unif}[S^{D-1}], \tag{A.2}$$

where S^{D-1} is the D-dimensional one-sphere. This will set the stage for the noisy case considered next. For a more formal and detailed treatment of the BPS algorithm, including ergodicity, see (Bouchard-Côté et al., 2015). For simplicity, we do not include here the velocity refreshments, which do not change the proof.

The proof sketches below are presented using a discrete-time approach followed by letting $\Delta t \to 0$. We have found this approach more accessible for a machine learning audience. After submitting a preliminary version of this work to the arXiv, the preprint (Fearnhead et al., 2016) was submitted to the arXiv, which presents similar proofs of invariance by first deriving a general Fokker-Planck equation and then showing that the equation is satisfied both in noiseless and noisy cases.

A.1. Exact Gradient

To understand why the algorithm is correct, consider first the transition rule

$$(\mathbf{w}, \mathbf{v})_{t+\Delta t} = \begin{cases} (\mathbf{w} + \mathbf{v}\Delta t, \mathbf{v}) & \text{with probability } 1 - \Delta t[G]_+ \\ (\mathbf{w} + \mathbf{v}\Delta t, \mathbf{v}_r) & \text{with probability } \Delta t[G]_+ \end{cases} \tag{A.3}$$

where

$$[x]_+ = \max(x, 0), \tag{A.4}$$

$$G = \mathbf{v} \cdot \nabla U(\mathbf{w}), \tag{A.5}$$

and

$$\mathbf{v}_r = \mathbf{v} - 2\frac{(\mathbf{v} \cdot \nabla U(\mathbf{w}))\nabla U(\mathbf{w})}{||\nabla U(\mathbf{w})||^2}. \tag{A.6}$$

This rule acts on the probability density $p(\mathbf{w}, \mathbf{v})$ as,

$$p_{t+\Delta t}(\mathbf{w}, \mathbf{v}) = [p_{t+\Delta t}(\mathbf{w}, \mathbf{v})]_d + [p_{t+\Delta t}(\mathbf{w}, \mathbf{v})]_r. \tag{A.7}$$

The two terms in (A.7) correspond to the two ways to reach $(\mathbf{w}, \mathbf{v})$ at time $t + \Delta t$. First, we can start at $(\mathbf{w} - \mathbf{v}\Delta t, \mathbf{v})$ at time t and move a distance $\mathbf{v}\Delta t$ without bouncing. This occurs with probability $1 - \Delta t[\mathbf{v} \cdot \nabla U]_+$, so we have

$$[p_{t+\Delta t}(\mathbf{w}, \mathbf{v})]_d = (1 - \Delta t[\mathbf{v} \cdot \nabla U]_+)p_t(\mathbf{v})p_t(\mathbf{w} - \mathbf{v}\Delta t), \tag{A.8}$$

$$= (1 - \Delta t[\mathbf{v} \cdot \nabla U]_+)p_t(\mathbf{v})(p_t(\mathbf{w}) - \Delta t\mathbf{v} \cdot \nabla p_t(\mathbf{w}) + O(\Delta t^2)), \tag{A.9}$$

$$= p_t(\mathbf{v})p_t(\mathbf{w})\left[1 + \Delta t\mathbf{v} \cdot \nabla U - \Delta t[\mathbf{v} \cdot \nabla U]_+\right] + O(\Delta t^2), \tag{A.10}$$

where in (A.9) we did a Taylor expansion and in (A.10) we used (A.1).

The second term in (A.7) corresponds to being at $(\mathbf{w} - \mathbf{v}_r\Delta t, \mathbf{v}_r)$ at time t, moving $\mathbf{v}_r\Delta t$ and bouncing. This occurs with probability $\Delta t[\mathbf{v}_r \cdot \nabla U]_+ = \Delta t[-\mathbf{v} \cdot \nabla U]_+$, so we have

$$[p_{t+\Delta t}(\mathbf{w}, \mathbf{v})]_r = \Delta t[-\mathbf{v} \cdot \nabla U]_+ p_t(\mathbf{w} - \mathbf{v}_r\Delta t, \mathbf{v}_r), \tag{A.11}$$

$$= \Delta t[-\mathbf{v} \cdot \nabla U]_+ p_t(\mathbf{w}, \mathbf{v}_r) + O(\Delta t^2), \tag{A.12}$$

where again we did a Taylor expansion in (A.11). Adding (A.10) and (A.12), and using

$$[\mathbf{v}\cdot\nabla U]_+ - [-\mathbf{v}\cdot\nabla U]_+ = \mathbf{v}\cdot\nabla U\,, \tag{A.13}$$

equation (A.7) becomes

$$p_{t+\Delta t}(\mathbf{w},\mathbf{v}) \;\;=\;\; p_t(\mathbf{w},\mathbf{v}) + O(\Delta t^2)\,, \tag{A.14}$$

which implies that the distribution is stationary, $\frac{dp_t(\mathbf{w},\mathbf{v})}{dt} = 0$.

A.2. Noisy Gradient

Consider now a noisy gradient represented as

$$\nabla\tilde{U}(\mathbf{w}) = \nabla U(\mathbf{w}) + \mathbf{n}_\mathbf{w}\,, \qquad\qquad \mathbf{n}_\mathbf{w}\sim p(\mathbf{n}_\mathbf{w}|\mathbf{w})\,, \qquad \mathbf{n}_\mathbf{w}\in\mathbb{R}^D\,, \tag{A.15}$$

where we assume that $p(\mathbf{n}_w|\mathbf{w})$ has zero mean.

First note that the requirement that $\mathbf{n}_\mathbf{w}$ and $\mathbf{n}'_\mathbf{w}$ are conditionally independent given $\mathbf{w}$ and $\mathbf{w}'$, with $\mathbf{w}\neq\mathbf{w}'$, is needed to preserve under the noise the Markov property of the sampler, which requires the bounce point process intensity to depend only on $\mathbf{w}$, and not the past history of the $\mathbf{w}$ trajectory.

Next we decompose the random vector $\mathbf{n}_\mathbf{w}$ into two orthogonal components,

$$\mathbf{n}_\mathbf{w} = y\mathbf{v} + \mathbf{n}_\mathbf{v}\,, \tag{A.16}$$

with $y = \mathbf{v}\cdot\mathbf{n}_\mathbf{w}$, and $\mathbf{n}_\mathbf{v}\cdot\mathbf{v} = 0$. This induces a corresponding decomposition in the probability density as

$$d\mathbf{n}_\mathbf{w}p(\mathbf{n}_\mathbf{w}|\mathbf{w}) = dyd\mathbf{n}_\mathbf{v}p(y|\mathbf{w})p(\mathbf{n}_\mathbf{v}|y,\mathbf{w},\mathbf{v})\,, \tag{A.17}$$

and note that from the assumption that $p(\mathbf{n}_\mathbf{w}|\mathbf{w})$ has zero mean it follows that $p(y|\mathbf{w})$ has zero mean. The noisy projected gradient becomes

$$\mathbf{v}\cdot\nabla U(\mathbf{w}) + y\,, \qquad y\sim p(y|\mathbf{w})\,. \tag{A.18}$$

To study the invariance of $p(\mathbf{w},\mathbf{v})$ under the noisy BPS, let us consider again the decomposition (A.7) into straight and bounced infinitesimal trajectories. The probability that the particle is at $(\mathbf{w}-\mathbf{v}\Delta t,\mathbf{v})$ at time t and moves a distance $\mathbf{v}\Delta t$ without bouncing is the average of $1-\Delta t[\mathbf{v}\cdot\nabla U(\mathbf{w})+y]_+$ over all the possible realizations of y, and is therefore given by

$$1-\Delta t P_\mathbf{v} \;\equiv\; 1-\Delta t\int_{-\infty}^{+\infty}[\mathbf{v}\cdot\nabla U(\mathbf{w})+y]_+p(y|\mathbf{w})dy\,, \tag{A.19}$$

$$\;=\; 1-\Delta t\int_{-\mathbf{v}\cdot\nabla U(\mathbf{w})}^{+\infty}(\mathbf{v}\cdot\nabla U(\mathbf{w})+y)p(y|\mathbf{w})dy\,, \tag{A.20}$$

where the above expression defines $P_\mathbf{v}$. The first term of (A.7) is therefore

$$[p_{t+\Delta t}(\mathbf{w},\mathbf{v})]_d \;=\; (1-\Delta tP_\mathbf{v})p(\mathbf{w}-\mathbf{v}\Delta t,\mathbf{v})\,, \tag{A.21}$$

$$\;=\; p_t(\mathbf{w},\mathbf{v}) - \Delta t\mathbf{v}\cdot\nabla p_t(\mathbf{w})p_t(\mathbf{v}) - \Delta tP_\mathbf{v}p_t(\mathbf{w})p_t(\mathbf{v}) + O(\Delta t^2)\,,$$

$$\;=\; p_t(\mathbf{w})p_t(\mathbf{v})[1+\Delta t\mathbf{v}\cdot\nabla U(\mathbf{w}) - \Delta tP_\mathbf{v}] + O(\Delta t^2)\,, \tag{A.22}$$

similarly to (A.8)-(A.10).

The second term in (A.7) now has contributions from all those values $(\mathbf{w}-\tilde{\mathbf{v}}_r\Delta t,\tilde{\mathbf{v}}_r)$ at time t, such that a reflection of $\tilde{\mathbf{v}}_r$ with respect to a noisy $\nabla\tilde{U}(\mathbf{w})$ gives $\mathbf{v}$. Such a $\tilde{\mathbf{v}}_r$ exists for every value of the noise vector $\mathbf{n}_w$, and is given by

$$\tilde{\mathbf{v}}_r = \mathbf{v} - 2\frac{(\mathbf{v}\cdot\nabla\tilde{U}(\mathbf{w}))\nabla\tilde{U}(\mathbf{w})}{||\nabla\tilde{U}(\mathbf{w})||^2}\,, \tag{A.23}$$

Therefore the second term in (A.7) contains contributions from all the possible realizations of $\mathbf{n_w}$ and is

$$[p_{t+\Delta t}(\mathbf{w},\mathbf{v})]_r = \Delta t \int_{\mathbb{R}^D} d\mathbf{n_w} [\tilde{\mathbf{v}}_r \cdot \nabla \tilde{U}(\mathbf{w})]_+ p(\mathbf{n_w}|\mathbf{w}) p_t(\mathbf{w} - \tilde{\mathbf{v}}_r \Delta t, \tilde{\mathbf{v}}_r)\,, \tag{A.24}$$

$$= \Delta t p_t(\mathbf{w},\tilde{\mathbf{v}}_r) \int_{-\infty}^{+\infty} dy\, p(y|\mathbf{w}) [-\mathbf{v}\cdot\nabla U(\mathbf{w}) - y]_+ \,, \times \int d\mathbf{n_v} p(\mathbf{n_v}|y,\mathbf{w},\mathbf{v}) + O(\Delta t^2)\,,$$

$$= \Delta t P_{\mathbf{v}_r} p_t(\mathbf{w},\tilde{\mathbf{v}}_r) + O(\Delta t^2)\,, \tag{A.25}$$

where we used $\tilde{\mathbf{v}}_r \cdot \nabla \tilde{U}(\mathbf{w}) = -\mathbf{v}\cdot\nabla U(\mathbf{w}) - y$, the measure decomposition (A.17), $\int d\mathbf{n_v} p(\mathbf{n_v}|y,\mathbf{w},\mathbf{v}) = 1$ and defined

$$P_{\mathbf{v}_r} = \int_{-\infty}^{-\mathbf{v}\cdot\nabla U(\mathbf{w})} dy\, (-\mathbf{v}\cdot\nabla U(\mathbf{w}) - y) p(y|\mathbf{w})\,. \tag{A.26}$$

Adding now (A.22) and (A.25), using $p(\tilde{\mathbf{v}}_r) = p(\mathbf{v})$ (since $p(\mathbf{v})$ is uniform) and

$$P_{\mathbf{v}} - P_{\mathbf{v}_r} = \mathbf{v}\cdot\nabla U(\mathbf{w})\,, \tag{A.27}$$

which follows from (A.20) and (A.26), and the fact that $p(y|\mathbf{w})$ has zero mean, we get again the stationarity condition

$$p_{t+\Delta t}(\mathbf{w},\mathbf{v}) = p_t(\mathbf{w},\mathbf{v}) + O(\Delta t^2)\,. \tag{A.28}$$

B. Biased Approximation

B.1. Biased bouncing rate

In the noiseless case, the velocity bounce is an event in a Poisson process with intensity $\lambda(\mathbf{w}) = [\mathbf{v}\cdot\nabla U(\mathbf{w})]_+$ while in the noisy case, the average Poisson intensity is $\lambda_n(\mathbf{w}) = E_y[\lambda_n(\mathbf{w},y)]$ where

$$\lambda_n(\mathbf{w},y) = [\mathbf{v}\cdot\nabla U(\mathbf{w}) + y]_+\,. \tag{B.29}$$

When a thinning upper bound for $[\mathbf{v}\cdot\nabla U + y]_+$ is unknown and the distribution of y is Gaussian with predicted variance ρ^2, our algorithm makes a bounce proposal from a Poisson process with intensity

$$\lambda_\rho(\mathbf{w}) = \hat{G} + k\rho(\mathbf{w})\,, \tag{B.30}$$

where $\hat{G}$ is our estimate of $\mathbf{v}\cdot\nabla U(\mathbf{w})$. At the proposed bounce point $\mathbf{w}$, we evaluate $\lambda_n(\mathbf{w},y)$, and accept with probability $\min(\lambda_n(\mathbf{w},y)/\lambda_\rho(\mathbf{w}),1)$. The evaluation of $\lambda_n(\mathbf{w},y)$ also provides an estimate $\sigma^2(\mathbf{w})$ of the variance of y. Assuming y is Gaussian, the probability of the bound violation event $1 < \lambda_n/\lambda_\rho$, is

$$q(\mathbf{w}) = 1 - \Phi((\lambda_\rho(\mathbf{w}) - \mathbf{v}\cdot\nabla U(\mathbf{w}))/\sigma(\mathbf{w}))\,, \tag{B.31}$$

where Φ is the standard normal CDF. For a given y, the intensity is therefore,

$$\lambda_b(\mathbf{w},y) = I_{[\frac{\lambda_n}{\lambda_\rho}<1]}\lambda_n(\mathbf{w},y) + I_{[\frac{\lambda_n}{\lambda_\rho}>1]}\lambda_\rho(\mathbf{w}) \tag{B.32}$$

where $I_{[\cdot]}$ is the indicator function. Averaging over y we get

$$\lambda_b(\mathbf{w}) = E_y[\lambda_b(\mathbf{w},y)] \tag{B.33}$$

$$= (1-q(\mathbf{w}))E_{\lambda_n\le\lambda_\rho}[\lambda_n(\mathbf{w},y)] + q(\mathbf{w})\lambda_\rho(\mathbf{w}) \tag{B.34}$$

If the probability of bound violation has a universal upper bound $q(\mathbf{w}) < q, \forall \mathbf{w}$, we assume

$$|\lambda_b(\mathbf{w}) - \lambda_n(\mathbf{w})| \le K_q = Cq + O(q^2) \tag{B.35}$$

where C is a constant.

B.2. Preliminaries

We are interested bounding the distance between the equilibrium distribution of the biased, noisy BPS process with mean intensity $\lambda_b(\mathbf{w})$, and the exact, noisy process with mean intensity $\lambda_n(\mathbf{w})$. We start with some preliminary results.

Wasserstein Distance and Kantorovich Duality

We will consider the Wasserstein distance, defined as

$$d_{\mathcal{W}}(p_1, p_2) = \sup_{f \in C_L} |E_{p_1}[f] - E_{p_2}[f]| \,, \tag{B.36}$$

where C_L is the set of 1-Lipshitz continuous functions,

$$C_L = \{f : \mathbb{R}^d \to \mathbb{R} : |f(y) - f(x)| \leq |y - x|\} \,. \tag{B.37}$$

Given random variables $\mathbf{z}_1 \sim p_1$, $\mathbf{z}_2 \sim p_2$, a coupling is a joint distribution $(\mathbf{z}_1, \mathbf{z}_2) \sim p_{12}$ with marginals p_1 and p_2. The Kantorovich duality (Villani, 2008) asserts that

$$d_{\mathcal{W}}(p_1, p_2) = \inf_{p_{12}} E_{p_{12}}[|\mathbf{z}_1 - \mathbf{z}_2|] \,. \tag{B.38}$$

Generators

To simplify the notation, let us define $\mathbf{z} = (\mathbf{w}, \mathbf{v})$, $\mathbf{z}_r = (\mathbf{w}, \tilde{\mathbf{v}}_r)$. The infinitesimal generator of a stochastic process is defined as

$$\mathcal{L}f(\mathbf{z}) = \lim_{\delta t \to 0} \frac{E[f(\mathbf{z}_{t+\delta t})|\mathbf{z}_t = \mathbf{z}] - f(\mathbf{z})}{\delta t} \,, \tag{B.39}$$

and note that it satisfies

$$\begin{aligned} E[\mathcal{L}f] &= \lim_{\delta t \to 0} \frac{\int d\mathbf{z}_{t+\delta t} d\mathbf{z} p(\mathbf{z}_{t+\delta t}|\mathbf{z}) p(\mathbf{z}) f(\mathbf{z}_{t+\delta t}) - E[f(\mathbf{z})]}{\delta t} \,, & \text{(B.40)} \\ &= \lim_{\delta t \to 0} \frac{\int d\mathbf{z}_{t+\delta t} p(\mathbf{z}_{t+\delta t}) f(\mathbf{z}_{t+\delta t}) - E[f(\mathbf{z})]}{\delta t} \,, & \text{(B.41)} \\ &= 0, & \text{(B.42)} \end{aligned}$$

where the expectation is with respect to the distribution $p(\mathbf{z})$ invariant under the stochastic process, and we used $\int d\mathbf{z} p(\mathbf{z}_{t+\delta t}|\mathbf{z}) p(\mathbf{z}) = p(\mathbf{z}_{t+\delta t})$. In our case, the generator of a BPS process with intensity $\lambda_n(\mathbf{w}, y)$ is (Davis, 1984; Fearnhead et al., 2016)

$$\mathcal{L}_{\lambda_n} f(\mathbf{z}) = \mathbf{v} \cdot \nabla_{\mathbf{w}} f(\mathbf{z}) + E_y[\lambda_n(\mathbf{w}, y)(f(\mathbf{z}_r) - f(\mathbf{z}))] \tag{B.43}$$

and similarly for $\lambda_b(\mathbf{w})$.

Let us define

$$f_\lambda(\mathbf{z}, t) = E_\lambda[f(\mathbf{z}_t)|\mathbf{z}_0 = \mathbf{z}] \,, \tag{B.44}$$

where the expectation is with respect to the distribution of the stochastic process with intensity λ at time t and with a given initial condition. This expression satisfies the backward Kolmogorov equation

$$\frac{\partial f_\lambda(\mathbf{z}, t)}{\partial t} = \mathcal{L}_\lambda f_\lambda(\mathbf{z}, t) \,, \tag{B.45}$$

and also (Jacod & Shiryaev, 1987)

$$\lim_{t \to \infty} f_\lambda(\mathbf{z}, t) = E_\lambda[f] \,, \tag{B.46}$$

where the expectation $E_\lambda[\cdot]$ is with respect to the distribution invariant under the stochastic process with intensity λ.

Ergodicity

We assume that the random process defined by SBPS is polynomial ergodic (although see the recent (Deligiannidis et al., 2017)). In particular, we assume that two distributions started at reflected velocities $p_{\lambda_n,t,\mathbf{z}} = p_{\lambda_n}(\mathbf{z}_t|\mathbf{z}_0 = \mathbf{z})$, $p_{\lambda_n,t,\mathbf{z}_r} = p_{\lambda_n}(\mathbf{z}_t|\mathbf{z}_0 = \mathbf{z}_r)$ converge as

$$d_{\mathcal{W}}(p_{\lambda_n,t,\mathbf{z}}, p_{\lambda_n,t,\mathbf{z}_r}) \leq \frac{C_A}{(\alpha + t)^\beta} \tag{B.47}$$

where α, β, C_A are constants.[1]

Poisson Equation

Given a function $f(\mathbf{z})$, we will consider below the Poisson equation

$$\mathcal{L}_\lambda u_f(\mathbf{z}) = f(\mathbf{z}) - E_\lambda[f]\,. \tag{B.48}$$

We assume the existence of the solution

$$u_f(\mathbf{z}) = \int_0^\infty ds(E_\lambda[f] - f_\lambda(\mathbf{z}, s))\,, \tag{B.49}$$

where $f_\lambda(\mathbf{z}, s)$ was defined in (B.44). The fact that this expression solves (B.48) can be easily verified using (B.45), (B.46) and $f_\lambda(\mathbf{z}, 0) = f(\mathbf{z})$. For $f \in C_L$ (see (B.37)), this solution satisfies

$$\begin{aligned}
|u_f(\mathbf{z}) - u_f(\mathbf{z}_r)| &= \left|\int_0^\infty ds(f_\lambda(\mathbf{z}, s) - f_\lambda(\mathbf{z}_r, s))\right| && \text{(B.50)}\\
&\leq \int_0^\infty ds E_\lambda[\mathbf{z}_s - \mathbf{z}_{r,s}|]\,, \quad \text{using the Lipshitz property} && \text{(B.51)}\\
&\leq \int_0^\infty ds\, d_{\mathcal{W}}(p_{\lambda,s,\mathbf{z}}, p_{\lambda,s,\mathbf{z}_r})\,, \quad \text{using (B.38)} && \text{(B.52)}\\
&\leq \frac{C_A}{(\beta+1)\alpha^{\beta+1}}\,, \quad \text{using the ergodicity assumption (B.47)}\,. && \text{(B.53)}
\end{aligned}$$

B.3. Distance Bound from Stein's Method

We now prove a bound on the distance between the exact and biased distributions, using Stein's method (Barbour, 1990; Ross, 2011; Stein et al., 1972), which was recently used for the related Zig-Zag process (Huggins & Zou, 2017).

$$\begin{aligned}
d_{\mathcal{W}}(p_{\lambda_n}, p_{\lambda_b}) &= \sup_{f\in C_L} |E_{\lambda_n}[f] - E_{\lambda_b}[f]|\,, && \text{(B.54)}\\
&= \sup_{f\in C_L} |E_{\lambda_n}[\mathcal{L}_{\lambda_b} u_f]|\,, \quad \text{using (B.48)} && \text{(B.55)}\\
&= \sup_{f\in C_L} |E_{\lambda_n}[\mathcal{L}_{\lambda_b} u_f] - E_{\lambda_n}[\mathcal{L}_{\lambda_n} u_f]|\,, \quad \text{using (B.42)} && \text{(B.56)}\\
&\leq \sup_{f\in C_L} E_{\lambda_n}[|(\mathcal{L}_{\lambda_b} - \mathcal{L}_{\lambda_n})u_f|]\,. && \text{(B.57)}
\end{aligned}$$

Note that this last expression involves an integral over just one distribution, unlike the first expression (B.54). Inside the expectation we have, using (B.43),

$$(\mathcal{L}_{\lambda_n} - \mathcal{L}_{\lambda_b})u_f(\mathbf{z}) \leq |\lambda_n(\mathbf{w}) - \lambda_b(\mathbf{w})| E_y[|u_f(\mathbf{z}_r) - u_f(\mathbf{z})|]\,, \tag{B.58}$$

[1] This assumed property follows usually from the existence of small sets (Lemma 3 in (Bouchard-Côté et al., 2015)) along with an appropriate Lyapunov function (Roberts et al., 2004).

where the expectation over y is because $\mathbf{z}_r = (\mathbf{w}, \mathbf{v}_r)$ depends on the noise y (see (A.23)). Using (B.35) and (B.53), we get finally

$$d_{\mathcal{W}}(p_{\lambda_n}, p_{\lambda_b}) \leq \frac{K_q C_A}{(\beta+1)\alpha^{\beta+1}} \,. \tag{B.59}$$

Interestingly, this bound depends on the mixing speed of the process generated by $\mathcal{L}_{\lambda_n}$ (see (B.47)), even though the distance is between two *equilibrium* distributions.

C. SBPS algorithm

Algorithm 1 provides a description of the SBPS algorithm with a linear regression based thinning proposal intensity. We have omitted velocity refreshments for the sake of clarity. Δt in the code below is the resolution of the piecewise linear proposal intensity, which should be smaller than the typical time between bounces. In all experiments a value of $\Delta t = .01$ was used.

Algorithm 1 Stochastic Bouncy Particle Sampler

SBPS:
Initialize particle position $\mathbf{w} \in \mathbb{R}^D$, velocity $\mathbf{v} \in S^{D-1}$, $t \leftarrow 0$, regression coefficients $\hat{\beta}_0, \hat{\beta}_1, \rho(t)$
while desired **do**
  $t, \lambda(t) =$ Sample_Proposal_Time($\hat{\beta}_0, \hat{\beta}_1, \rho(t)$)
  $\mathbf{w} \leftarrow \mathbf{w} + \mathbf{v} * t$
  Store $\mathbf{w}, t$
  Observe $\nabla \tilde{U}(\mathbf{w})$, Var $[\mathbf{v} \cdot \nabla \log p(x_{r_i}|\mathbf{w})]$
  (optional: Update preconditioner and apply it to gradient - see **??**)
  Calculate $\tilde{G}(t), c(t)$
  $\mathbf{v} =$ Accept/Reject_Proposal($\tilde{G}(t), \lambda(t), \mathbf{v}$)
  $\hat{\beta}_0, \hat{\beta}_1, \rho(t) =$ Update_Local_Regression_Coefficients($\tilde{G}(t), c(t), t$)
end while
Return piecewise linear trajectory of $\mathbf{w}$

Sample_Proposal_Time($\hat{\beta}_0, \hat{\beta}_1, \rho(t)$):
$t_{next\_proposal} \leftarrow 0$
Initialize set of interpolation points $p = \{[\hat{\beta}_1 t_{next\_proposal} + \hat{\beta}_0 + k\rho(t_{next\_proposal})]_+\}$
Initialize piecewise linear proposal intensity $\lambda(t) = Inter(p)$*
Sample $u \sim$ Unif$[0, 1]$
while $-log(u) > \int_0^{t_{next\_proposal}} \lambda(t)dt$ **do**
  $t_{next\_proposal} \leftarrow t_{next\_proposal} + min(\Delta t, -log(u) - \int_0^{t_{next\_proposal}} \lambda(t)dt)$
  $p \leftarrow p \cup [\hat{\beta}_1 t_{next\_proposal} + \hat{\beta}_0 + k\rho(t_{next\_proposal})]_+$
  $\lambda = Inter(p)$
end while
Return $t_{next\_proposal}, \lambda(t_{next\_proposal})$
* $Inter(p)$ is a linear interpolation of the points in p and their respective times since the last proposal

Accept/Reject_Proposal($\tilde{G}(t), \lambda(t), \mathbf{v}$):
Draw $u \sim$ Unif$[0, 1]$
if $u > \tilde{G}(\mathbf{w})/\lambda(t)$ **then**
  Proposed bounce time accepted:
  Initialize $\{\tilde{G}(t_i), c(t_i)\}$ and regression coefficients $\hat{\beta}_0, \hat{\beta}_1, \rho(t)$ using $\tilde{G}(t), c(t)$
  Return $\mathbf{v} - 2\frac{(\mathbf{v} \cdot \nabla \tilde{U}(\mathbf{w}))\nabla \tilde{U}(\mathbf{w})}{||\nabla \tilde{U}(\mathbf{w})||^2}$
else
  Proposed bounce time rejected, maintain current trajectory:
  Return $\mathbf{v}$
end if

Update_Local_Regression_Coefficients($\tilde{G}(t), c(t), t$):
Add $\tilde{G}(t), c(t)$ to $\{\tilde{G}(t_i), c(t_i)\}$
(optional: Perform hyperparameter learning step on regression priors)
Update regression coefficients $\hat{\beta}_0, \hat{\beta}_1, \rho(t')$ using standard Bayesian regression formula
(optional: If $\hat{\beta}_1 < 0$ set $\hat{\beta}_1$ to non-negative value, update $\hat{\beta}_0$ accordingly)
Return $\hat{\beta}_0, \hat{\beta}_1, \rho(t)$

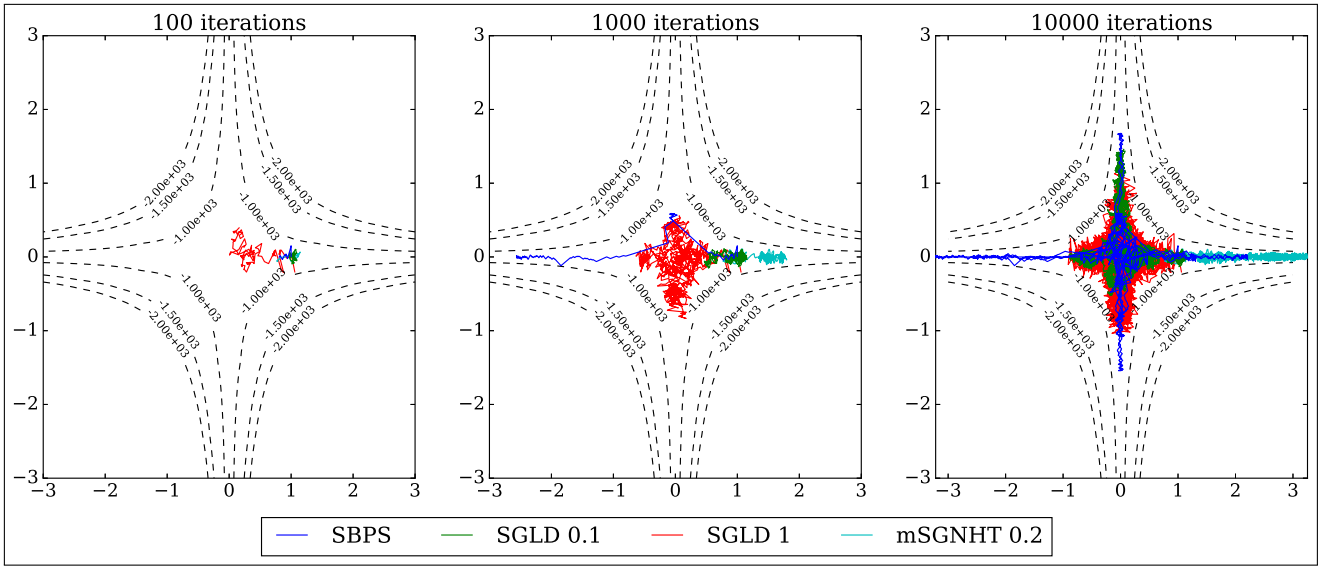

Figure 1: Sample traces of SBPS, SGLD and mSGNHT sampling a highly non-Gaussian posterior. SBPS appears to explore the posterior more fully and avoids regions of low density as opposed to the large step size SGLD, leading to less bias.

D. A Highly non-Gaussian Example

We explored a case of sampling from a Bayesian posterior where the Laplace approximations is not accurate. Figure 1 shows results for a highly non-log-concave 2D hyperboloid posterior, with data generated according to $y_i \sim \mathcal{N}(w_0^* w_1^*, \sigma)$. After introducing a weak Gaussian prior, the resulting log posterior takes the form

$$L(w_0, w_1 | \{y_i\}, \sigma) = \sum_{i=1}^{N} - \frac{(y_i - w_0 w_1)^2}{2\sigma^2} - \frac{c}{2} ||w||_2^2 \,. \tag{D.60}$$

This posterior was approximated by observing mini-batches of data as in the previous examples. The scaling symmetry that is manifest in the invariance of the likelihood with respect to $w_0, w_1 \to \lambda w_0, \frac{w_1}{\lambda}$ leads to the highly non-Gaussian hyperboloid form. Similar symmetries are encountered in posteriors of deep neural networks with ReLU activation (Dinh et al., 2017). The parameters used were $N = 1000, n = 100, k = 3, c = .0001, w_0^* = w_1^* = 0, \sigma = 1$. σ was not learned.

Figure 1 shows comparisons with SGLD and mSGNHT, while Local BPS and SS-ZZ cannot be applied since there seems to be no simple exact upper bound for thinning in this case. Note that for the step sizes shown, both SGLD and mSGNHT deviate into low density regions while not mixing as well as SBPS. The smaller SGLD step size used does not deviate as much but exhibits even slower mixing.

E. Sampling from Multimodal Targets

Our simple linear model for $G(t)$ (the projected gradient of the log posterior), appears to be sufficiently accurate even when the Laplace approximation is violated (as shown in the previous examples), but in some highly multimodal cases we have found this approximation to be insufficient.

In this section we present a slight modification of SBPS to sample from such targets as well. The potential troubles arise because in a multimodal target one may encounter situations where the measured $G(t)$ drop quickly between successive observations, leading to strong negative regression slopes. In our regression model, we get an interpolation (cf. equation (13)) leading to an upper bound of the form

$$\hat{\beta}_1 t + \hat{\beta}_0 + k\rho(t) \tag{E.61}$$

with $\hat{\beta}_1 < 0$. Since the leading order t dependence in $\rho(t)$ is linear, if k is too small the linear term in (E.61) may be negative. This will lead the sampler to propose long times between samples and thus enter low target density regions of the space.

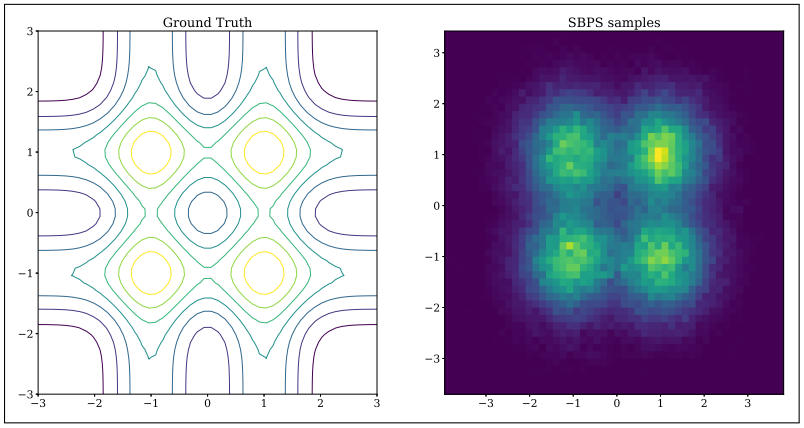

Figure 2: SBPS sampling from a highly multimodal target. As can be seen from the sample histogram on the right, SBPS manages to accurately capture the multimodal target. The results are from 1000 epochs of sampling from a dataset of size $N = 1000$

In such cases, we propose to make additional auxiliary observations at times $\{t_{aux}\}$ along the current linear trajectory of the particle and update the linear bound accordingly before making the next proposal. On a large enough scale this procedure will make auxiliary observations $\tilde{G} > 0$ leading to a positive slope in (E.61). This in turn will prevent the particle from entering low target density regions.

Note that these auxiliary observations can be performed with the same minibatch of data from the last bounce proposal. In principle, such strong negative slopes can occur even for a unimodal target if the subsampling noise is highly non-Gaussian, and this mechanism can also be used in those situations.

We illustrate this mechanism in a simple distribution defined as

$$L(w) = \sum_{i=1}^{N} \sum_{k=1}^{D} L_i(w_k) + \text{const.} \tag{E.62}$$

where $w \in \mathbb{R}^D$ and

$$L_i(w_k) = \log \left[e^{-\frac{(w_k - 1 - \mu_k^i)^2}{2\sigma_L^2}} + e^{-\frac{(w_k + 1 - \mu_{D+k}^i)^2}{2\sigma_L^2}} \right] \tag{E.63}$$

and each μ_k^i is drawn from $\mathcal{N}(0, \sigma_\mu)$. This is a highly multimodal toy distribution. Although it does not come from a posterior distribution, it allows us to illustrate the proposed mechanism in a clean setting. Figure 2 shows results for $D = 2$, $N = 1000, \sigma_L = .25, \sigma_\mu = .01$ and mini-batch size $n = 10$. We used $\{t_{aux}\} = \{10p\bar{t}, p \in \mathbb{N}\}$ where $\bar{t}$ is the mean proposal time of all past proposals during the sampling process. While one can add multiple auxiliary points in this way, in practice we have found that one auxiliary point ($p = 1$) is sufficient. Figure 2 shows that SBPS is able to correctly sample from the target while avoiding the issues posed by the multimodality of the distribution. We note that this modified mechanism does not affect the rest of the examples presented in this paper.

F. SGLD Step Size Scan

In the logistic regression example of Section 6.1, we compare SBPS with Stochastic Gradient Langevin Dynamics (SGLD) (Welling & Teh, 2011) with fixed step size. A natural question is how to choose an appropriate step size that ensures the fastest possible mixing without introducing an unacceptable amount of bias. Our criterion was to pick the biggest possible (i.e., fastest-mixing) step size such that the resulting variance of the per-data-point Negative Log Likelihood (NLL) coincides with that of the Laplace approximation. The latter gives a per-data-point NLL distribution of $\frac{1}{2N}\chi^2(d) + NLL_{\hat{\mathbf{w}}}/N$ where $\hat{\mathbf{w}}$ is the MAP estimator (Bickel & Doksum, 2015). The results of this parameter scan are shown in Figure 3 and suggest a step size of 0.1.

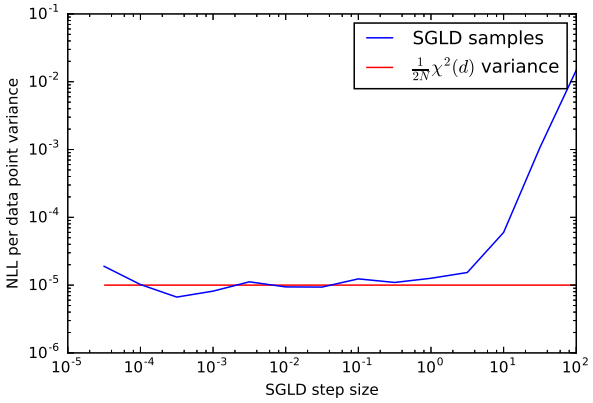

Figure 3: Per-data-point variance of the NLL in the logistic regression example of Section 7.1, using SGLD samples with step sizes of $10^{-i/2}, i = 0...9$. The samplers were initialized at the MAP. We select the biggest step size whose empirical variance is below that from the Laplace approximation, $\frac{d}{2N^2}$.

G. The effect of the SBPS hyperparameters

In this section we explore, in the logistic regression example of Section 6.1, the effect of two hyperparameters that control the behavior of SBPS: the mini-batch size n, and the width k of the upper confidence band. A third hyperparameter is the rate of velocity refreshments, shown in (Bouchard-Côté et al., 2015) to be necessary in general to prove ergodicity. But, as mentioned in Section 3, in the examples we considered the mini-batch noise was enough to sufficiently randomize possible non-mixing trajectories, so we could safely set this parameter to a very low value.

G.1. Mini-batch size n

Figure 4 shows an exploration of different values of the mini-batch size n. Low values for n lead to high noise for $\tilde{G}$. This in turn yields higher values for the proposal intensity $\gamma(t)$, which leads to shorter linear trajectories between bounce proposals. This is consistent with the results of Figure 4 that show a linear relation between n (i.e. computational cost per bounce proposal) and the average travel time between bounces. The autocorrelation functions (ACFs) were computed from discrete samples obtained by running SBPS with different n's such that the total data cost was the same for all cases, and then discretizing the continuous paths into equal numbers of uniformly spaced samples. As shown, these cost adjusted ACFs are quite similar. On the other hand, the upper-left panel, shows that lower values of n have faster convergence to equilibrium, suggesting that low n should be preferred. But this should be contrasted with the fact that shorter linear trajectories increase the variance of expectations over rapidly changing functions, as discussed in Section 6.3.

G.2. Upper-band width k

Figure 5 shows an exploration of different values of k, the height of the proposal intensity above the estimator mean, in units of predictive standard deviation (see in Eq.(12) in main text). It therefore controls the trade-off between a regime, at low k, of faster mixing and high bias from violations of the thinning upper bound ($[\tilde{G}(t)]_+/\lambda(t) > 1$), and another regime, high k, of low bias and high variance from slower mixing. As expected, the probability of bound violation decreases monotonically with k, as seen in the bottom left panel of Figure 5.

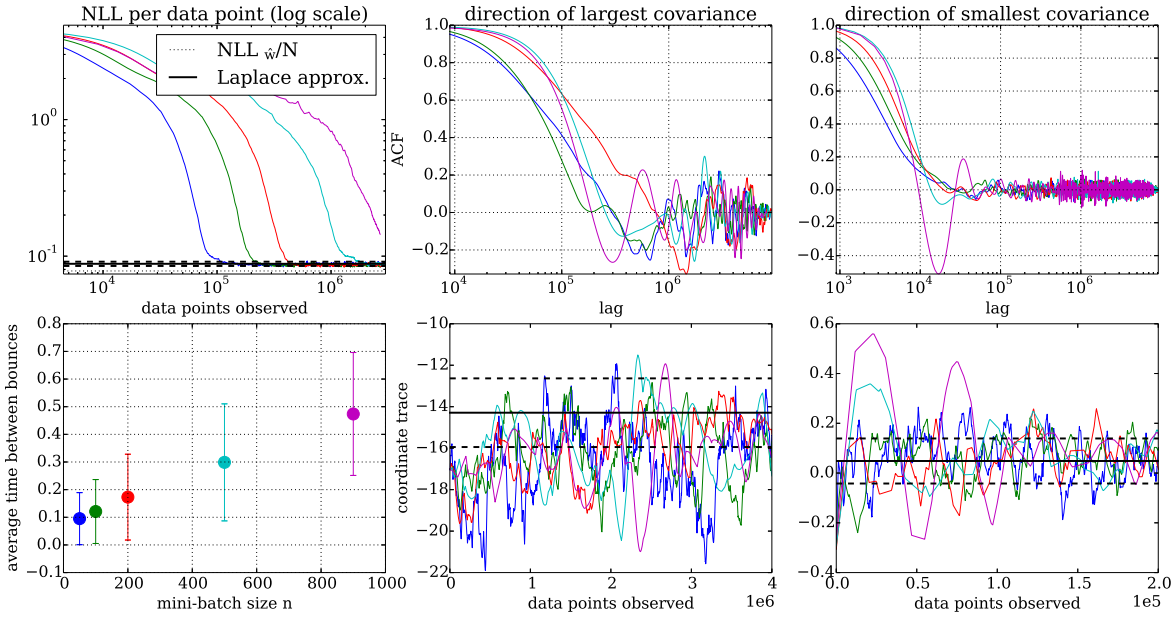

Figure 4: Effect of mini-batch sizes n in the logistic regression example of Section 6.1. Mini-batch sizes were $50, 100, 200, 500, 900$. *Top Left:* Average per-data-point NLL over 5 runs. Note that smaller n lead to faster convergence to a region of low NLL. *Lower Left:* Estimated average time between particle bounces. *Center/Right:* ACF and trajectories from a single run, in the directions of smallest and biggest covariance. The x axis was chosen differently for the trajectory plots for clarity.

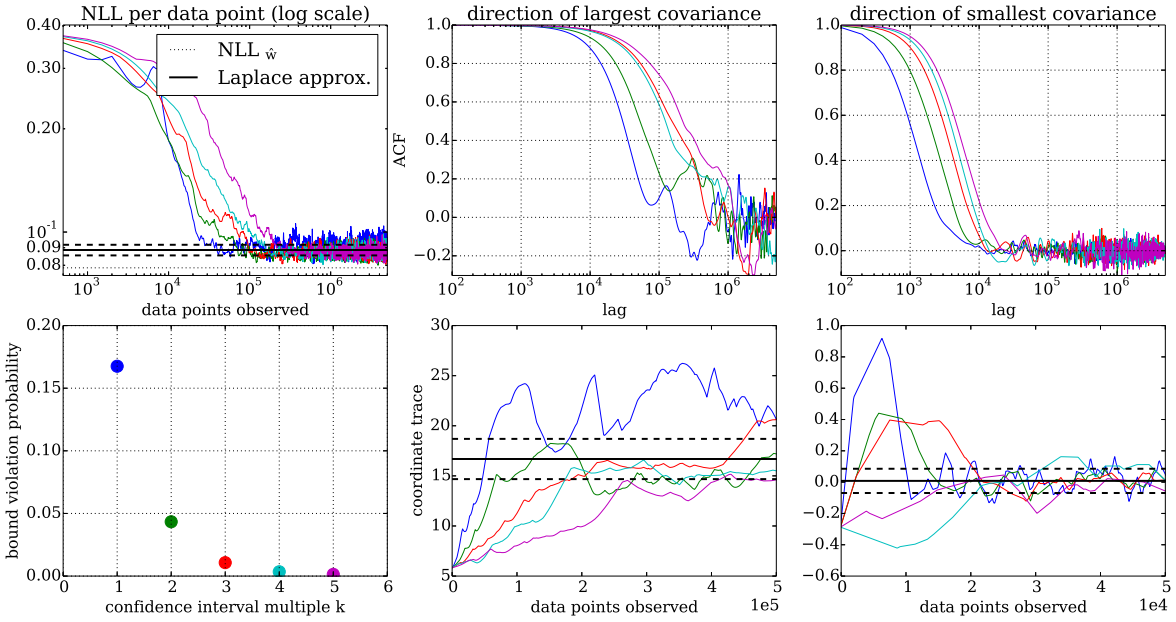

Figure 5: Effect of upper band size k in the logistic regression example of Section 6.1, run with mini-batch size $n = 100$. *Bottom Left:* Rate of upper bound violations as a function of k; the same colors are used in the other plots. *Top Left:* NLL per data point for samples of SBPS with different k values. *Center/Right:* ACF and trajectories in the directions of smallest and biggest covariance. Note that smaller k leads to faster convergence and mixing but increased bias, as visible in the coordinate trace in the direction of biggest covariance. The x axis was chosen differently for the trajectory plots for clarity.

H. Upper Bounds for Logistic Regression

In the case of logistic regression with data $(y_i, \mathbf{x}_i)$ the estimator of $\nabla_{\mathbf{w}} U(\mathbf{w})$ from a mini-batch of size n is

$$\nabla_{\mathbf{w}} \tilde{U}(\mathbf{w}) = \frac{N}{n} \sum_{i=1}^{n} \mathbf{x}_i (\sigma(\mathbf{w} \cdot \mathbf{x}_i) - y_i) \,. \tag{H.64}$$

A simple bound on $\tilde{G}(t)$ is therefore given by

$$\tilde{G}(t) \leq \frac{N}{n} |\sum_{i=1}^{n} (\mathbf{v} \cdot \mathbf{x}_i)(\sigma(\mathbf{w} \cdot \mathbf{x}_i) - y_i)| \,, \tag{H.65}$$

$$\leq \frac{N}{n} \sum_{i=1}^{n} ||\mathbf{v}||_2 ||(\sigma(\mathbf{w} \cdot \mathbf{x}_i) - y_i)\mathbf{x}_i||_2 \,, \tag{H.66}$$

$$\leq \frac{N}{n} \sum_{i=1}^{n} ||x_i||_2 \,, \tag{H.67}$$

$$\leq \sqrt{d} N \max_{i,j} |x_{ij}| \,. \tag{H.68}$$

This is a particular case of a bound derived in (Bierkens et al., 2017). Compared to the bound proposed in (Bouchard-Côté et al., 2015), this bound is more conservative but cheaper to compute and does not require non-negative covariates. It similarly scales like N and when the data used in the experiments was modified so that the covariates were non-negative the bounds differed by a factor lower than 2.

References

Barbour, Andrew D. Stein's method for diffusion approximations. *Probability theory and related fields*, 84(3):297–322, 1990.

Bickel, Peter J and Doksum, Kjell A. *Mathematical Statistics: Basic Ideas and Selected Topics, volume I*, volume 117. CRC Press, 2015.

Bierkens, Joris, Bouchard-Côté, Alexandre, Doucet, Arnaud, Duncan, Andrew B, Fearnhead, Paul, Roberts, Gareth, and Vollmer, Sebastian J. Piecewise Deterministic Markov Processes for Scalable Monte Carlo on Restricted Domains. *arXiv preprint arXiv:1701.04244*, 2017.

Bouchard-Côté, Alexandre, Vollmer, Sebastian J, and Doucet, Arnaud. The Bouncy Particle Sampler: A Non-Reversible Rejection-Free Markov Chain Monte Carlo Method. *arXiv:1510.02451*, 2015.

Davis, Mark HA. Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models. *J. Royal Stat. Soc., Series B (Methodological)*, pp. 353–388, 1984.

Deligiannidis, George, Bouchard-Côté, Alexandre, and Doucet, Arnaud. Exponential Ergodicity of the Bouncy Particle Sampler. *arXiv preprint arXiv:1705.04579*, 2017.

Dinh, Laurent, Pascanu, Razvan, Bengio, Samy, and Bengio, Yoshua. Sharp minima can generalize for deep nets. *arXiv preprint arXiv:1703.04933*, 2017.

Fearnhead, Paul, Bierkens, Joris, Pollock, Murray, and Roberts, Gareth O. Piecewise Deterministic Markov Processes for Continuous-Time Monte Carlo. *arXiv preprint arXiv:1611.07873*, 2016.

Huggins, Jonathan H and Zou, James. Quantifying the accuracy of approximate diffusions and Markov chains. In *AISTATS*, 2017.

Jacod, J. and Shiryaev, A.N. *Limit Theorems for Stochastic Processes*. Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen. Springer-Verlag, 1987. ISBN 9783540178828.

Roberts, Gareth O, Rosenthal, Jeffrey S, et al. General state space markov chains and mcmc algorithms. *Probability Surveys*, 1:20–71, 2004.

Ross, Nathan. Fundamentals of Stein's method. *Probab. Surv*, 8:210–293, 2011.

Stein, Charles et al. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*. The Regents of the University of California, 1972.

Villani, Cédric. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Welling, Max and Teh, Yee W. Bayesian learning via stochastic gradient Langevin dynamics. In *ICML*, pp. 681–688, 2011.