Supplementary Material to Bernoulli Race Particle Filters

Sebastian M Schmon **Arnaud Doucet** **George Deligiannidis**
Department of Statistics, University of Oxford

A Proofs

When we say that T is a geometric random variable with parameter, or success probability, p we mean that

$$\mathbb{P}(T \geq k) = (1-p)^k, \qquad k \geq 1.$$

Proposition 3. *Assume* $\lim_{N\to\infty} \rho_N =: \rho \in (0,1)$. *Then we have the following central limit theorem for the average number of coin flips as* $N \to \infty$

$$\sqrt{N}\left(\frac{1}{N}\sum_{j=1}^{N} C_{j,N} - \frac{1}{\rho_N}\right) \xrightarrow{d} \mathcal{N}\left(0, \frac{1-\rho}{\rho^2}\right),$$

where $\xrightarrow{d}$ *denotes convergence in distribution.*

Proof. The random variables $Z_{i,N} = \left(C_{i,N} - 1/\rho_N\right)/\sqrt{N}$ form a triangular array

$$\{Z_{i,N}; i = 1, \ldots N, N \in \mathbb{N}\}$$

of row wise independent random variables. Define $s_N^2 = \sum_{i=1}^{N} \mathrm{var}\left(Z_{i,N}\right) = \left(1-\rho_N\right)/\rho_N^2$ and notice that $s_N^2 \to 1-\rho \in (0,1)$ as $N \to \infty$. In addition one can easily show that $\mathbb{E}[|Z_{1,N}|^3]$ are bounded uniformly in N and therefore

$$\begin{aligned}\frac{1}{s_N^3}\sum_{i=1}^{N} \mathbb{E}\left[Z_{i,N}\right] &= \frac{\rho_N^{3/2}}{(1-\rho_N)^3}\frac{N}{N^{3/2}}\mathbb{E}\left[|Z_{1,N}|^3\right] \\ &\leq \frac{C}{\sqrt{N}} \to 0, \qquad \text{as } N \to \infty,\end{aligned}$$

that is the *Lyapunov condition* is satisfied and therefore $N \to \infty$

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left(C_{i,N} - \frac{1}{\rho_N}\right) \xrightarrow{d} \mathcal{N}\left(0, s^2\right),$$

where $s^2 = \lim_{N\to\infty} s_N^2 = (1-\rho)/\rho^2$. □

Proposition 4. *For* $N \in \mathbb{N}$ *let* $C_{1,N}, \ldots, C_{N,N}$ *denote independent samples from a geometric distribution with success probability* ρ_N, *then the minimum variance unbiased estimator for* ρ_N *is*

$$\hat{\rho}_N^{\text{mvue}} = \frac{N-1}{\sum_{k=1}^{N} C_{k,N} - 1}. \tag{1}$$

Proof. This is a straightforward application of the Lehmann-Scheffé Theorem [4, Theorem 7.4.1]. Take the unbiased estimator $1\left\{C_{1,N} = 1\right\}$, where $1\{A\}$ denotes the indicator function of the set A, and condition on the complete and sufficient statistic (of the coin flips) $\sum_{k=1}^{N} C_{k,N}$. A straightforward calculation yields

$$\begin{aligned}\mathbb{E}\left[1\left\{C_{1,N}=1\right\} \mid \sum_{k=1}^{N} C_{k,N}=n\right] &= \frac{\mathbb{P}\left(C_1=1, \sum_{k=2}^{N} C_{k,N}=n-1\right)}{\mathbb{P}\left(\sum_{k=1}^{N} C_{k,N}=n\right)} \\ &= \frac{p\binom{n-2}{N-2}p^{N-1}(1-p)^{n-N}}{\binom{n-1}{N-1}p^{N}(1-p)^{n-N}} \\ &= \frac{N-1}{n-1}\end{aligned}$$

and thus

$$\hat{\rho}_N^{\text{mvue}} = \mathbb{E}\left[1\left\{C_{1,N}=1\right\} \mid \sum_{k=1}^{N} C_{k,N}\right] = \frac{N-1}{\sum_{k=1}^{N} C_{k,N}-1}. \qquad \square$$

Proposition 5. *For $N \in \mathbb{N}$ let $C_{1,N}, C_{2,N}, \dots$ denote independent samples from a Geometric distribution with success probability ρ_N, then*

$$\sqrt{N}\left(\hat{\rho}_N^{\text{mvue}} - \rho_N\right) \xrightarrow{d} \mathcal{N}\left(0, (1-\rho)\,\rho^2\right).$$

Proof. By Proposition 3 we have convergence

$$\sqrt{N}\left(\frac{1}{N}\sum_{i=1}^{N} C_{i,N} - \frac{1}{\rho_N}\right) \xrightarrow{d} \mathcal{N}\left(0, \frac{1-\rho}{\rho^2}\right).$$

In addition,

$$\left|\sqrt{N}\left(\frac{\sum_{i=1}^{N} C_{i,N}-1}{N-1} - \frac{1}{\rho_N}\right) - \sqrt{N}\left(\frac{1}{N}\sum_{i=1}^{N} C_{i,N} - \frac{1}{\rho_N}\right)\right| \to 0$$

almost surely, so

$$\sqrt{N}\left(\frac{\sum_{i=1}^{N} C_{i,N}-1}{N-1} - \frac{1}{\rho_N}\right) \xrightarrow{d} \mathcal{N}\left(0, \frac{1-\rho}{\rho^2}\right).$$

By the δ-Method with $g(x) = 1/x$, $|g'(x)| = 1/x^2$ we obtain

$$\begin{aligned}\sqrt{N}\left(g\left(\frac{\sum_{i=1}^{N} C_{i,N}-1}{N-1}\right) - g\left(\frac{1}{\rho_N}\right)\right) &= \sqrt{N}\left(\frac{N-1}{\sum C_{i,N}-1} - \rho_N\right) \\ &\xrightarrow{d} \mathcal{N}\left(0, \frac{1-\rho}{\rho^2}\cdot\rho^4\right) = \mathcal{N}\left(0, (1-\rho)\,\rho^2\right). \qquad \square\end{aligned}$$

Theorem 6. *The estimator*

$$\hat{\rho}_{N,T} = \prod_{t=1}^{T} \frac{1}{N}\sum_{k=1}^{N} c_t^k \cdot \frac{N-1}{\sum_{k=1}^{N} C_{k,N}-1}.$$

is unbiased for $p(y_{1:T})$, i.e.

$$\mathbb{E}\left[\hat{\rho}_{N,T}\right] = p(y_{1:T}).$$

Proof. The main argument is that the standard proof for unbiasedness using backward induction can be adapted to include our estimator using the geometric random variables $C_{1,N}, \ldots, C_{N,N}$. Denote by $X, \overline{X}$ the random variables before and after resampling respectively. Then we have

$$
\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} w_T^k \mid X_{T-1}^{1:N}\right] \\
&= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} g(y_T \mid X_T^i) \mid X_{T-1}^{1:N}\right] \\
&= \frac{1}{N}\sum_{i=1}^{N}\int g(y_T \mid x_T^i) p(x_T^i \mid X_{T-1}^{1:N}) dx_T^i \\
&= \frac{1}{N}\sum_{i=1}^{N}\int g(y_T \mid x_T^i) \sum_{j=1}^{N} f(x_T^i \mid X_{T-1}^j) \frac{g(y_{T-1} \mid X_{T-1}^j)}{\sum_{k=1}^N g(y_{T-1} \mid X_{T-1}^k)} dx_T^i \\
&= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\int g(y_T \mid x_T^i) f(x_T^i \mid X_{T-1}^j) \frac{g(y_{T-1} \mid X_{T-1}^j)}{\sum_{k=1}^N g(y_{T-1} \mid X_{T-1}^k)} dx_T^i \\
&= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N} p(y_T \mid X_{T-1}^j) \frac{g(y_{T-1} \mid X_{T-1}^j)}{\sum_{k=1}^N g(y_{T-1} \mid X_{T-1}^k)} \\
&= \frac{\sum_{j=1}^N p(y_{T-1:T} \mid X_{T-1}^j)}{\sum_{k=1}^N g(y_{T-1} \mid X_{T-1}^k)}.
\end{aligned}
$$

Now, considering the case where we estimate the weights using the geometric random variables, we have

$$
\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} c_T^k \cdot \frac{N-1}{\sum_{k=1}^N C_{k,N,T} - 1} \mid X_{T-1}^{1:N}\right] \\
&= \mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} c_T^k \mathbb{E}\left\{\frac{N-1}{\sum_{k=1}^N C_{k,N,T} - 1} \mid X_T^{1:N}, \overline{X}_{T-1}^{1:N}\right\} \mid X_{T-1}^{1:N}\right] \\
&= \mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} c_T^k \frac{\sum_{k=1}^N w_T^k}{\sum_{k=1}^N c_T^k} \mid X_{T-1}^{1:N}\right] \\
&= \mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} w_T^k \mid X_{T-1}^{1:N}\right] \\
&= \frac{\sum_{k=1}^N p(y_{T-1:T} \mid x_{T-1}^k)}{\sum_{k=1}^N g(y_{T-1} \mid x_{T-1}^k)}
\end{aligned}
$$

and

$$
\begin{aligned}
&\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T-1}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T-1}-1}\right)\cdot\left(\frac{1}{N}\sum_{j=1}^{N} c_{T}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T}-1}\right)\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T-1}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T-1}-1}\right)\cdot\left(\frac{1}{N}\sum_{j=1}^{N} c_{T}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T}-1}\right)\mid X_{T-2:T-1}^{1:N},\overline{X}_{T-2}^{1:N},C_{1:N,N,T-1}\right]\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T-1}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T-1}-1}\right)\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T}-1}\right)\mid X_{T-2:T-1}^{1:N},\overline{X}_{T-2}^{1:N},C_{1:N,N,T-1}\right]\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T-1}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T-1}-1}\right)\frac{\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})}{\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})}\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^{N} c_{T-1}^{j}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N,T-1}-1}\right)\mid X_{T-1}^{k},\overline{X}_{T-2}^{k}\right]\frac{\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})}{\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})}\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\left(\frac{1}{N}\sum_{k=1}^{N} w_{T-1}^{k}\right)\cdot\frac{\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})}{\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})}\mid X_{T-2}^{1:N}\right].
\end{aligned}
$$

For the final expectation, we can calculate

$$
\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} w_{T-1}^{k}\cdot\frac{\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})}{\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})}\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})\frac{\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})}{\sum_{k=1}^{N} g(y_{T-1}\mid X_{T-1}^{k})}\mid X_{T-2}^{1:N}\right]\\
&=\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} p(y_{T-1:T}\mid X_{T-1}^{k})\mid X_{T-2}^{1:N}\right]\\
&=\frac{1}{N}\sum_{k=1}^{N}\int p(y_{T-1:T}\mid x_{T-1}^{k})p(x_{T-1}^{k}\mid X_{T-2}^{1:N})dx_{T-1}^{k}\\
&=\frac{1}{N}\sum_{k=1}^{N}\int p(y_{T-1:T}\mid x_{T-1}^{k})\frac{\sum_{j=1}^{N} g(y_{T-2}\mid X_{T-2}^{j})f(x_{T-1}^{k}\mid X_{T-2}^{j})}{\sum_{j=1}^{N} g(y_{T-2}\mid X_{T-2}^{j})}dx_{T-1}^{k}\\
&=\sum_{j=1}^{N} p(y_{T-1:T}\mid X_{T-2}^{j})\frac{g(y_{T-2}\mid X_{T-2}^{j})}{\sum_{k=1}^{N} g(y_{T-2}\mid X_{T-2}^{k})}=\frac{\sum_{j=1}^{N} p(y_{T-2:T}\mid X_{T-2}^{j})}{\sum_{j=1}^{N} g(y_{T-2}\mid X_{T-2}^{j})}.
\end{aligned}
$$

Repeated application of these steps yields

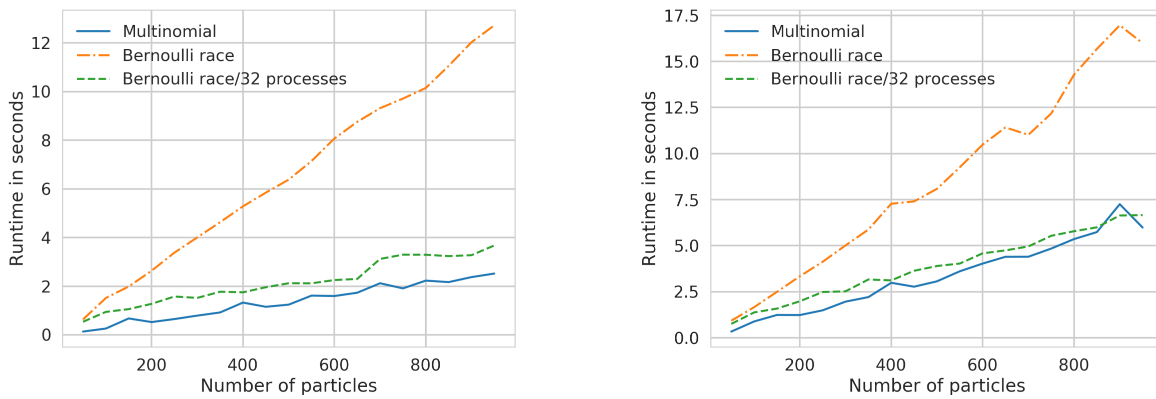
$$
\begin{aligned}
\mathbb{E}\left[\prod_{t=1}^{T}\frac{1}{N}\sum_{k=1}^{N} c_{t}^{k}\cdot\frac{N-1}{\sum_{k=1}^{N} C_{k,N}-1}\right]&=\mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} g(y_1\mid X_1^{k})\frac{\sum_{j=1}^{N} p(y_{1:T}\mid X_1^{j})}{\sum_{j=1}^{N} g(y_1\mid X_1^{j})}\right]\\
&=p(y_{1:T}).
\end{aligned}
$$

□

B Further Details to the Applications

B.1 Locally optimal proposal: run-time

We conjecture that the advantage of the BRPF over the RWPF grows as the state transition get computationally more expensive. We will demonstrate that on a toy example.



(a) Simulating from known Gaussian.

(b) Rejection Sampler.

Figure 1: Comparing the run-time for different implementation for the proposal $q(\cdot \mid x, y)$. In case (a) the Gaussian proposal is sampled using a naïve implementation. In (b) we use a rejection sampler proposing from the state transition.

First note that in case of the Gaussian state space model, the locally optimal proposal,

$$\begin{aligned} q^*(x_t \mid y_t, x_{t-1}) &\propto g(y_t \mid x_t) f(x_t \mid x_{t-1}) \\ &\propto \varphi\left(x_t; \frac{1}{2}(a x_{t-1} + y_t), 2.5^2\right) \end{aligned}$$

is known analytically and is Gaussian. Therefore, in this special case the rejection sampler can be avoided. This will not usually be the case, hence we use the rejection sampler for our simulation. However, this will significantly speed up the state transition. In Figure 1 we compare the run-time of both, the RWPF and BRPF for different numbers of particles. In scenario (a) we use the cheap transition. Here the RWPF is very efficient. However, in scenario (b), which we also show in the main paper, where sampling from the transition is more expensive, we can implement the BRPF with the same run-time as the RWPF.

B.2 Sine diffusion: run-time

Here we compare the run-time for the RWPF and BRPF for the sine diffusion state space model. As pointed out in the main text, the BRPF is slower when implemented sequentially, but we observe in Figure 2 that the difference almost completely vanishes when use a parallel implementation with 16 processes.

C Cox Process inference

To further demonstrate the range of possible application and for further illustration of the Bernoulli race particle filter, we consider another application of the BRPF to a Cox process whose intensity function is governed by Gaussian process that is normalized through a sigmoid function. The underlying Gaussian process is modelled as a Gauss–Markov process, allowing us to perform inference sequentially as done by Li and Godsill [5] based on models of Adams et al. [1] and Christensen et al. [3].

C.1 Likelihood and Estimation

We assume the latent Gaussian process to evolve according to the Ornstein–Uhlenbeck process

$$dX_t = AX_t dt + h dB_t \qquad t \in [0, \mathcal{T}], \tag{2}$$

where $(B_t)_{t\in[0,\mathcal{T}]}$ denotes a Brownian motion from 0 to $\mathcal{T}$. We will describe the structural assumptions on A and h in more detail below. Denote the observed data by $s_{1:n} \subset [0, \mathcal{T}]$, where n is the number of observations.

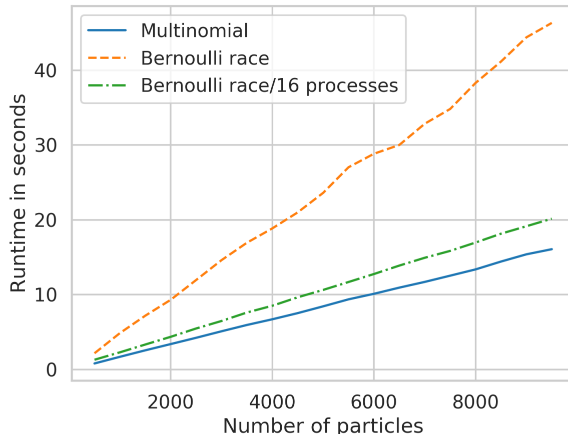


Figure 2: Comparison of the run-time for RWPF and BRPF for sine diffusion state space model. For BRPF we show a sequential and a parallel implementation with 16 cores. For all algorithms we show wall-clock time for number of particles N.

We model the data by a Cox process with intensity function

$$\lambda(t) = \sigma(X_t) = \frac{1}{1+\exp(-X_t)} = \frac{\exp(X_t)}{1+\exp(X_t)},$$

where $(X_t)_{t\in[0,\mathcal{T}]}$ is the SDE (2) introduced before. The likelihood function of the Cox process conditional on the intensity is Poisson with

$$p(s_{1:n} \mid \lambda, \mathcal{T}) = \exp\left(-\int_0^{\mathcal{T}} \lambda(s)ds\right)\prod_{i=1}^{n}\lambda(s_i). \tag{3}$$

The likelihood is clearly intractable due to the integral which involves the latent Gaussian process. To implement a particle filter for sequential inference we need to find an unbiased coin with probability proportional to the integral in (3). For any interval $[t_0, t_1] \subset [0, \mathcal{T}]$ we can find an estimator for $\exp(-\int_{t_0}^{t_1}\lambda(s)ds)$ by observing that for $U \sim U[t_0, t_1]$ and a fixed process $X = (X_t)_{t\in[t_0,t_1]}$ we have

$$\begin{aligned}\mathbb{E}\left[\lambda(U) \mid X = x\right] &= \int_{t_0}^{t_1} \frac{\lambda(s)}{t_1 - t_0} ds \\ &= \frac{1}{t_0 - t_1}\int_{t_0}^{t_1} \frac{1}{1+\exp(-x_s)} ds\end{aligned}$$

and hence, if $V \sim U[0, 1]$ independent of all other random variables, we obtain

$$P\left(V \leq \lambda(U) \mid X = x\right) = \int_{t_0}^{t_1} \frac{\lambda(s)}{t_1 - t_0} ds.$$

Now sample $K \sim \text{Pois}(t_1 - t_0)$, then an unbiased coin flip can be generated using

$$Z = \prod_{i=1}^{K} 1\left\{V_i \leq 1 - \lambda(U_i)\right\},$$

where $1\{A\}$ denotes the indicator function of the set A. This estimator is indeed unbiased as can be seen by

$$\begin{aligned}
\mathbb{E}[Z] &= \mathbb{E}\left[\prod_{i=1}^{K} 1\{V_i \leq 1-\lambda(U_i)\} \mid X = x\right] \\
&= e^{-(t_1-t_0)} \sum_{k=0}^{\infty} \frac{(t_1-t_0)^k}{k!} \prod_{i=1}^{k} \mathbb{E}\left[1\{V_i \leq 1-\lambda(U_i)\} \mid K = k, X = x\right] \\
&= e^{-(t_1-t_0)} \sum_{k=0}^{\infty} \frac{(t_1-t_0)^k}{k!} \left(\int_{t_0}^{t_1} \frac{1-\lambda(s)}{t_1-t_0} ds\right)^k \\
&= e^{-(t_1-t_0)} \sum_{k=0}^{\infty} \frac{\left(\int_{t_0}^{t_1} 1-\lambda(s)ds\right)^k}{k!} \\
&= e^{-(t_1-t_0)} e^{t_1-t_0} \exp\left(-\int_{t_0}^{t_1} \lambda(s)ds\right) \\
&= \exp\left(-\int_{t_0}^{t_1} \lambda(s)ds\right).
\end{aligned}$$

To implement the Bernoulli race particle filter we sample from the prior, which can be done exactly, as seen in the next section. As we are sampling from the prior the particle weights just depend on the observation likelihood. Suppose the particle filter is at current time t_0 and we propose to move to $t_1 = t_0 + \Delta t$. Let $(X_{t_1}^k), k = 1, \ldots, N$ denote a proposed particle path. Then the weight is given by

$$g(s_{i:\, s_i \in [t_0,t_1]} \mid X^k) = \exp\left(-\int_{t_0}^{t_1} \lambda^k(s)ds\right) \prod_{i:\, s_i \in [t_0,t_1]} \lambda(s_i^k),$$

where

$$\lambda^k(s) = \frac{\exp(X_s^k)}{1+\exp(X_s^k)}.$$

In the notation of Section 4 we have

$$\begin{aligned}
c_{t_1}^k &= \prod_{i:\, s_i \in [t_0,t_1]} \lambda^k(s_i), \\
b_{t_1}^k &= \exp\left(-\int_{t_0}^{t_1} \lambda^k(s)ds\right).
\end{aligned}$$

To implement the estimator $\hat{Z}$, we need to be able to simulate a bridge from the stochastic differential equation (2). Since the process at hand is analytically tractable, exact sampling from the bridge is straightforward, see e.g. Bladt et al. [2] using the quantities computed below.

C.2 Ornstein–Uhlenbeck Prior

In this section we provide further details on the assumptions on the Gaussian process and show how we can sample the state transition. We assume the prior distribution on the latent space is given by the Ornstein–Uhlenbeck process (2), that is

$$dX_t = AX_t dt + h dB_t \qquad t \in [0, \mathcal{T}].$$

For A and h we take the values

$$A = \begin{bmatrix} 0 & 1 \\ 0 & \theta \end{bmatrix}, \quad h = \begin{bmatrix} 0 \\ \sigma \end{bmatrix}$$

where θ is a negative real value. The process can be written as a differential equation with random velocity following a mean reverting real-valued Ornstein–Uhlenbeck process,

$$\begin{pmatrix} dx_{1,t} \\ dx_{2,t} \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & \theta \end{bmatrix} \begin{pmatrix} x_{1,t} \\ x_{2,t} \end{pmatrix} dt + \begin{pmatrix} 0 \\ \sigma \end{pmatrix} dW_t.$$

This can be written as

$$\begin{aligned} dx_{1,t} &= x_{2,t}dt \\ dx_{2,t} &= \theta x_{2,t}dt + \sigma dB_t. \end{aligned}$$

This is a linear Gaussian system and thus the solution can be derived analytically, see e.g. Christensen et al. [3]. The discretized system at time $s > r$ can be simulated exactly by sampling

$$X_s \mid (X_r = x) \sim \mathcal{N}\left(e^{A(s-r)x}, e^{A(s-r)}Q(r,s)\left(e^{A(s-r)}\right)^T\right),$$

where

$$Q(r,s) = \int_r^s \exp(-At)hh^T \exp(-At)^T dt.$$

In order to implement the Ornstein–Uhlenbeck process we need the quantities

$$\exp(At) = \sum_{k=0}^{\infty} t^k \frac{A^k}{k!}.$$

Note that

$$A^2 = \begin{bmatrix} 0 & 1 \\ 0 & \theta \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 0 & \theta \end{bmatrix} = \begin{bmatrix} 0 & \theta \\ 0 & \theta^2 \end{bmatrix}$$

$$\begin{aligned} A^k = A \cdot A^{k-1} &= \begin{bmatrix} 0 & 1 \\ 0 & \theta \end{bmatrix}\begin{bmatrix} 0 & \theta^{k-2} \\ 0 & \theta^{k-1} \end{bmatrix} \\ &= \begin{bmatrix} 0 & \theta^{k-1} \\ 0 & \theta^k \end{bmatrix} \end{aligned}$$

Hence, we can compute the matrix exponential analytically

$$\begin{aligned} \exp(At) &= \sum_{k=0}^{\infty} \frac{t^k}{k!} A^k \\ &= \sum_{k=0}^{\infty} \frac{t^k}{k!} \begin{bmatrix} 0 & \theta^{k-1} \\ 0 & \theta^k \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \sum_{k=1}^{\infty} \frac{t^k\theta^k}{k!} \begin{bmatrix} 0 & 1/\theta \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & \exp(\theta t)/\theta - 1/\theta \\ 0 & \exp(\theta t) - 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & \exp(\theta t)/\theta - 1/\theta \\ 0 & \exp(\theta t) \end{bmatrix}. \end{aligned}$$

This gives us

$$\begin{aligned} \exp(-At)h &= \begin{bmatrix} 1 & \exp(-\theta t)/\theta - 1/\theta \\ 0 & \exp(-\theta t) \end{bmatrix}\begin{bmatrix} 0 \\ \sigma \end{bmatrix} \\ &= \begin{bmatrix} \frac{\sigma}{\theta}\exp(-\theta t) - \sigma/\theta \\ \sigma\exp(-\theta t) \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned} &\exp(-At)hh^T \exp(-At) \\ &= \begin{bmatrix} \frac{\sigma}{\theta}\exp(-\theta t) - \sigma/\theta \\ \sigma\exp(-\theta t) \end{bmatrix}\begin{bmatrix} \frac{\sigma}{\theta}\exp(-\theta t) - \sigma/\theta & \sigma\exp(-\theta t) \end{bmatrix} \\ &= \begin{bmatrix} \frac{\sigma^2}{\theta^2}\left(\exp(-\theta t) - 1\right)^2 & \frac{\sigma^2}{\theta}\left(\exp(-2\theta t) - \exp(-\theta t)\right) \\ \frac{\sigma^2}{\theta}\left(\exp(-2\theta t) - \exp(-\theta t)\right) & \sigma^2\exp(-2\theta t) \end{bmatrix}. \end{aligned}$$

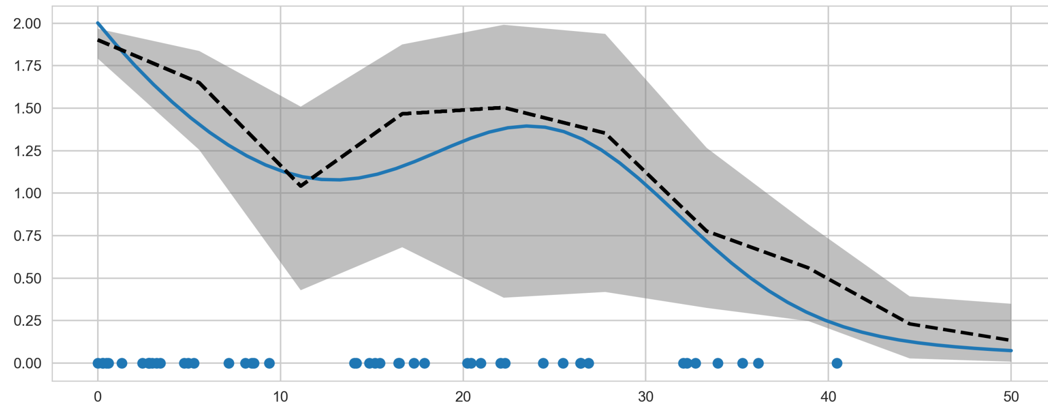


Figure 3: Result of one run of the Bernoulli race particle filter using 30 particles. Arrival times of the Poisson process are shown as dots on the abscissa. The solid blue line shows the true intensity function and the dashed line shows the mean of the particle approximation at times $t_1, t_2, \ldots$. The shaded area highlights the 10% and 90% quantile of the particle approximation.

Thus

$$
\begin{aligned}
Q(r,s) &= \int_r^s \exp(-At)hh^T \exp(-At)^T \, dt \\
&= \begin{bmatrix} \int_r^s \sigma^2 \left(\exp(-\theta t) - 1\right)^2 dt & \int_r^s \frac{\sigma^2}{\theta} \left(\exp(-2\theta t) - \exp(-\theta t)\right) dt \\ \int_r^s \sigma^2 \left(\exp(-2\theta t) - \exp(-\theta t)/\theta\right) dt & \int_r^s \sigma^2 \exp(-2\theta t) dt \end{bmatrix} \\
&= \begin{bmatrix} \frac{\sigma^2}{\theta^3}\left(-2\theta r + e^{-2\theta r} - 4e^{-\theta r} - e^{-2\theta s} + 4e^{-\theta s} + 2\theta s\right) & \frac{\sigma^2}{2\theta^2}\left(e^{-2r\theta} - e^{-2s\theta}\right) - \frac{\sigma^2}{\theta^2}\left(e^{-r\theta} - e^{-s\theta}\right) \\ \frac{\sigma^2}{2\theta^2}\left(e^{-2r\theta} - e^{-2s\theta}\right) - \frac{\sigma^2}{\theta^2}\left(e^{-r\theta} - e^{-s\theta}\right) & \frac{\sigma^2}{2\theta}\left(e^{-2r\theta} - e^{-2s\theta}\right) \end{bmatrix}
\end{aligned}
$$

The covariance matrix for the system transition

$$
\mathrm{Cov}(r,s) = e^{A(s-r)} Q(r,s) \left(e^{A(s-r)}\right)^T
$$

can thus be computed analytically.

C.3 Simulation

We simulate a non-homogeneous Poisson process in the interval $[0, 50]$ using the intensity function

$$
\lambda(s) = 2\exp(-s/15) + \exp\left(-((s-25)/10)^2\right),
$$

see e.g. Adams et al. [1] and Li and Godsill [5]. We run the Bernoulli race particle filter with the above specification and 30 particles to find the intensity function on the interval $[0, 50]$ which is divided into 10 equispaced intervals. The result of one run is shown in Figure 3. We can see that the Bernoulli race particle filter is approximating the true intensity function.

References

[1] R. P. Adams, I. Murray, and D. J. MacKay. "Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM. 2009, pp. 9–16.

[2] M. Bladt, S. Finch, and M. Sørensen. "Simulation of multivariate diffusion bridges". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 78.2 (2016), pp. 343–369.

[3] H. L. Christensen, J. Murphy, and S. J. Godsill. "Forecasting high-frequency futures returns using online Langevin dynamics". In: *IEEE Journal of Selected Topics in Signal Processing* 6.4 (2012), pp. 366–380.

[4] R. V. Hogg, J. W. McVean, and A. T. Craig. *Introduction to Mathematical Statistics*. 6th edition. Pearson Education, Upper Saddle River, New Jersey, 2005.

[5] C. Li and S. J. Godsill. "Sequential Inference Methods for Non-Homogeneous Poisson Processes with State-Space Prior". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 2856–2860.