

Inference for a class of partially observed point process models

James S. Martin · Ajay Jasra · Emma McCoy

Received: 20 June 2011 / Revised: 23 May 2012 / Published online: 30 August 2012
© The Institute of Statistical Mathematics, Tokyo 2012

Abstract This paper presents a simulation-based framework for sequential inference from partially and discretely observed point process models with static parameters. Taking on a Bayesian perspective for the static parameters, we build upon sequential Monte Carlo methods, investigating the problems of performing sequential filtering and smoothing in complex examples, where current methods often fail. We consider various approaches for approximating posterior distributions using SMC. Our approaches, with some theoretical discussion are illustrated on a doubly stochastic point process applied in the context of finance.

Keywords Point processes · Sequential Monte Carlo · Intensity estimation

1 Introduction

Partially observed point processes provide a rich class of models to describe real data. For example, such models are used for stochastic volatility ([Barndorff-Nielsen and Shephard 2001](#)) in finance, descriptions of queuing data in operations research

J. S. Martin
Australian School of Business, University of New South Wales,
Sydney, NSW 2052, Australia
e-mail: james.martin04@ic.ac.uk

A. Jasra (✉)
Department of Statistics and Applied Probability,
National University of Singapore, 6 Science Drive 2, Singapore 117546, Singapore
e-mail: stjaja@nus.edu.sg

E. McCoy
Department of Mathematics, Imperial College London,
180 Queens Gate, London SW7 2AZ, UK
e-mail: e.mccoy@ic.ac.uk

(Fearnhead 2004), important seismological models (Daley and Vere-Jones 1988) and applications in nuclear physics (Snyder and Miller 1998). For complex dynamic models, i.e., when data arrive sequentially in time, studies date back to at least Snyder (1972). However, fitting Bayesian models requires sequential Monte Carlo (SMC) (e.g. Doucet et al. 2001) and Markov chain Monte Carlo (MCMC) methods. The main developments in this field include the work of Centanni and Minozzo (2006a,b), Green (1995), Del Moral et al. (2006, 2007), Doucet et al. (2006), Roberts et al. (2004), Rydberg and Shephard (2000), see also Whiteley et al. (2011). As we describe below, the SMC methodology may fail in some scenarios and we will describe methodology to deal with the problems that will be outlined.

Informally, the problem of interest is as follows. A process is observed discretely upon a given time-interval $[0, T]$. The objective is to draw inference at time-points $t_0 = 0 < t_1 < \dots < t_{\bar{m}} < T = t_{\bar{m}+1}$, on the unobserved marked point process (PP) $(k_{t_n}, \phi_{1:k_{t_n}}, \zeta_{1:k_{t_n}})$, where $\phi_{1:k_{t_n}} = (\phi_1, \dots, \phi_{k_{t_n}})$ are the ordered event times (constrained to $[0, t_n]$) with k_{t_n} the number of event times up-to t_n and $\zeta_{1:k_{t_n}} = (\zeta_1, \dots, \zeta_{k_{t_n}})$ are marks, given the observations $y_{1:r_{t_n}}$, with r_{t_n} the number of observations up-to t_n . In other words to compute, for $n \geq 1$, at time t_n

$$\pi_n(k_{t_n}, \phi_{1:k_{t_n}}, \zeta_{1:k_{t_n}} | y_{1:r_{t_n}}) \text{ smoothing} \quad (1)$$

$$\pi_n(k_{t_n} - k_{t_{n-1}}, \phi_{k_{t_{n-1}}+1:k_{t_n}}, \zeta_{k_{t_{n-1}}+1:k_{t_n}} | y_{1:r_{t_n}}) \text{ filtering.} \quad (2)$$

In addition, there are static parameters specifying the probability model and these parameters will be estimated in a Bayesian manner. At this stage a convention in our terminology is established. An algorithm is said to be *sequential* if it is able to process data as it arrives over time. An algorithm is said to be *on-line* if it is sequential and has a fixed computational cost per iteration/time-step.

One of the first works applying computational methods to PP models was Rydberg and Shephard (2000). They focus upon a Cox model where the unobserved PP parameterizes the intensity of the observations. Rydberg and Shephard (2000) used the auxiliary particle filter (Pitt and Shephard 1997) to simulate from the posterior density of the intensity at a given time point. This was superseded by Centanni and Minozzo (2006a,b), which allows one to infer the intensity at any given time, up to the current observation. Centanni and Minozzo (2006a,b) perform an MCMC-type filtering algorithm, estimating static parameters using stochastic EM. The methodology cannot easily be adapted to the case where the static parameters are given a prior distribution. In addition, the theoretical validity of the approach has not been established, this is verified in Proposition 1.

SMC samplers (Del Moral et al. 2006) are the focus of this paper and can be applied to all the problems stated above. SMC methods simulate a set of $N \geq 1$ weighted samples, termed particles, in order to approximate a sequence of distributions, which may be chosen by the user, but which include (or are closely related to) the distributions in (1) and (2). Such methods are provably convergent as $N \rightarrow \infty$ (Del Moral 2004). A key feature of the approach is that the user must select:

1. the sequence of distributions,
2. the mechanism by which particles are propagated.

If points 1 and 2 are not properly addressed, there can be a substantial discrepancy between the proposal and target, thus the variance of the weights will be large and estimation inaccurate. This issue is particularly relevant when the targets are defined on a sequence of nested spaces, as is the case for the PP models—the space of the point process trajectories becomes larger with the time-parameter n . Thus, in choosing the sequence of target distributions, we are faced with the question of how much the space should be enlarged at each iteration of the SMC algorithm and how to choose a mechanism to propose particles in the new region of the space. This issue is referred to as *the difficulty of extending the space*.

Two solutions are proposed. The first is to *saturate* the state-space, it is supposed that the observation interval, $[0, T]$, of the PP is known a priori. The sequence of target distributions is then defined on the whole interval and one sequentially introduces likelihood terms, i.e. the sequence of target distributions is initially the prior distribution with the unobserved process allowed to lie on $[0, T]$. As the likelihood can be written as a product of r_T terms, each subsequent target (up-to proportionality) is the old one, multiplied by the density of the next data-point in the sequence. This idea circumvents the problem of extending the space, at an extra computational cost. Inference for the original density of interest can be achieved by importance sampling (IS). This approach cannot be used if T is unknown. In the second approach, entitled *data-point tempering*, the sequence of target distributions is defined by sequentially introducing likelihood terms, as above, except that the hidden process can only lie on $[0, t_n]$. This is achieved as follows: given that the PP has been sampled on $[0, t_n]$ the target is extended onto $[0, t_{n+1}]$ by sampling the missing part of the PP. Then one introduces likelihood terms into the target that correspond to the data (as in [Chopin 2002](#)). Once all of the data have been introduced, the target density is (1). It should be noted that neither of the methods is online, but some simple fixes are detailed.

Section 2 introduces a doubly stochastic PP model from finance which serves as a running example. In Sect. 3, the ideas of [Centanni and Minozzo \(2006a,b\)](#) are discussed; it is established that the method is theoretically valid under some assumptions. The difficulty of extending the state space is also demonstrated. In Sect. 4, we introduce our SMC methods. In Sect. 5 our methods are illustrated on the running example. In Sect. 6, we detail extensions to our work.

Some notations are introduced. We consider a sequence of probability measures $\{\varpi_n\}_{1 \leq n \leq m^*}$ on spaces $\{(G_n, \mathcal{G}_n)\}_{1 \leq n \leq m^*}$, with dominating σ -finite measures. Bounded and measurable functions on G_n , $f_n : G_n \rightarrow \mathbb{R}$, are written $\mathcal{B}_b(G_n)$ and $\|f_n\| = \sup_{x \in G_n} |f_n(x)|$. ϖ_n will refer to either the probability measure $\varpi_n(dx)$ or the density $\varpi_n(x)$.

2 Model

The model we use to illustrate our ideas is from statistical finance. An important type of financial data is ultra high-frequency data which consist of the irregularly spaced times of financial transactions and their corresponding monetary value. Standard models for the fitting of such data have relied upon stochastic differential equations driven by Wiener dynamics, a debatable assumption due to the continuity of the sample paths.

As noted in Centanni and Minozzo (2006b), it is more appropriate to model the data as a Cox process. Due to the high frequency of the data, it is important to be able to perform sequential/on-line inference. Data are observed in $[0, T]$. In the context of finance, the assumption that T be fixed is entirely reasonable. For example, when the model is used in the context of equities, the model is run for the trading day; indeed due to different (deterministic) patterns in financial trading, it is likely that the fixed parameters below are varied according to the day.

A marked PP, of $r_T \geq 1$ points, is observed in time-period $[0, T]$. This is written $y_{1:r_T} = (\omega_{1:r_T}, \xi_{1:r_T}) \in \Omega_{r,T} \times \Xi^{r_T}$ with $\Omega_{r,T} = \{\omega_{1:r_T} : 0 < \omega_1 < \dots < \omega_{r_T} < T\}$, $\Xi \subseteq \mathbb{R}$. Here, the ω are the transaction times and ξ are the log-returns on the financial transactions. An appropriate model for such data, as in Centanni and Minozzo (2006b), is

$$\begin{aligned} \tilde{p}(\xi_{1:r_T} | \mu, \sigma) &= \prod_{i=1}^{r_T} \tilde{p}(\xi_i; \mu, \sigma) \\ \tilde{p}(\omega_{1:r_T} | \{\lambda_T\}) &\propto \prod_{i=1}^{r_T} \{\lambda_{\omega_i}\} \exp \left\{ - \int_0^T \lambda_u du \right\} \end{aligned}$$

with \tilde{p} a generic density, $\xi_i | \mu, \sigma$ are assumed to be t -distributed on 1 degree of freedom, location μ , scale σ and λ_u is the intensity of the hidden process up-to time u . The unobserved intensity process is assumed to follow the dynamics $d\lambda_t = -s\lambda_t dt + dJ_t$ with $\{J_t\}$ a compound Poisson process: $J_t = \sum_{j=1}^{k_t} \zeta_j$ with $\{K_t\}$ a Poisson process with rate parameter ν and i.i.d. jumps $\zeta_j \sim \mathcal{E}x(1/\gamma)$, $\mathcal{E}x(\cdot)$ is the exponential distribution. That is, for $t \in [0, T]$,

$$\lambda_t = \lambda_0 e^{-st} + \sum_{j=1}^{k_t} \zeta_j e^{-s(t-\phi_j)} \tag{3}$$

with ϕ_j the jump times of the unobserved Poisson process and λ_0 fixed throughout (using a short preliminary time series that is available in practice).

We define the following notation:

$$\begin{aligned} \bar{x}_n &= (k_{t_n}, \phi_{1:k_{t_n}}, \zeta_{1:k_{t_n}}), \\ \bar{x}_{n,1} &= (k_{t_n} - k_{t_{n-1}}, \phi_{k_{t_{n-1}}+1:k_{t_n}}, \zeta_{k_{t_{n-1}}+1:k_{t_n}}), \\ \bar{y}_n &= (\omega_{1:r_{t_n}}, \xi_{1:r_{t_n}}), \\ \bar{y}_{n,1} &= (\omega_{r_{t_{n-1}}+1:r_{t_n}}, \xi_{r_{t_{n-1}}+1:r_{t_n}}). \end{aligned}$$

Here \bar{x}_n (respectively, \bar{y}_n) is the restriction of the hidden (observed) PP to events in $[0, t_n]$. Similarly, $\bar{x}_{n,1}$ (respectively, $\bar{y}_{n,1}$) is the restriction of the hidden (observed) PP to events in $[t_{n-1}, t_n]$.

The objective is to perform inference at times $0 < t_1 < \dots < t_{\bar{m}} < T = t_{\bar{m}+1}$, i.e., to update the posterior distribution conditional on the data arriving in $[t_{n-1}, t_n]$. To summarize, the posterior distribution at time t_n is

$$\begin{aligned}
 \pi_n(\bar{x}_n, \mu, \sigma | \bar{y}_n) &\propto \prod_{i=1}^{r_{t_n}} \{ \tilde{p}(\xi_i; \mu, \sigma) \lambda_{\omega_i} \} \exp \left\{ - \int_0^{t_n} \lambda_u du \right\} \\
 &\quad \times \prod_{i=1}^{k_{t_n}} \{ \mathbf{p}(\zeta_i) \} \mathbf{p}(\phi_{1:k_{t_n}}) \mathbf{p}(k_{t_n}) \times \tilde{p}(\mu, \sigma) \\
 &= l_{[0, t_n]}(\bar{y}_n; \bar{x}_n, \mu, \sigma) \times \mathbf{p}(\bar{x}_n) \times \tilde{p}(\mu, \sigma) \tag{4}
 \end{aligned}$$

with $l_{[0, t_n]}$ corresponding to the first part of the equation above, $\mu \sim \mathcal{N}(\alpha_\mu, \beta_\mu)$, $\sigma \sim \mathcal{Ga}(\alpha_\sigma, \beta_\sigma)$, $\phi_{1:k_t} | k_t \sim \mathcal{U}_{\Phi_{k_t}}$, $k_t \sim \mathcal{Po}(\gamma t)$ and where \mathcal{U}_A is the uniform distribution on the set A , $\mathcal{N}(\mu, \sigma^2)$ is the normal distribution of mean μ and variance σ^2 , $\mathcal{Ga}(\alpha, \beta)$ the Gamma distribution of mean α/β and \mathcal{Po} is the Poisson distribution. $\mathbf{p}(\bar{x}_n)$ is the notation for the prior on the hidden point-process and $\tilde{p}(\mu, \sigma)$ is the notation for the prior on (μ, σ) . Later, a π_0 is introduced which will refer to an initial distribution. Note it is possible to perform inference on (μ, σ) independently of the unobserved PP; it will not significantly complicate the simulation methods to include them.

It is of interest to compute expectations w.r.t. the $\{\pi_n\}_{1 \leq n \leq m^*}$, and this is possible, using the SMC methods below (Sect. 3.1). However, such algorithms are not of fixed computational cost; the sequence of spaces over which the $\{\pi_n\}_{1 \leq n \leq m^*}$ lie is increasing. These methods can also be used to draw inference from the marginal posterior of the process, over $(t_{n-1}, t_n]$; such algorithms can be designed to be of fixed computational complexity, for example by constraining any simulation to a fixed-size state-space. This idea is considered further in Sect. 4.3.

3 Previous approaches

One of the approaches for performing filtering for partially observed PPs is from Centanni and Minozzo (2006a). In this section, the parameters (μ, σ) are assumed known. Let

$$\bar{E}_n = \bigcup_{k \in \mathbb{N}_0} \left(\{k\} \times \Phi_{k, t_n} \times (\mathbb{R}^+)^k \right).$$

This is the support of the target densities for this method. The following decomposition is adopted

$$\begin{aligned}
 \pi_n(\bar{x}_n | \bar{y}_n) &= \frac{l_{(t_{n-1}, t_n]}(\bar{y}_{n,1}; \bar{x}_n)}{\tilde{p}_n(\bar{y}_{n,1} | \bar{y}_{n-1})} \mathbf{p}(\bar{x}_{n,1}) \pi_{n-1}(\bar{x}_{n-1} | \bar{y}_{n-1}) \\
 \tilde{p}_n(\bar{y}_{n,1} | \bar{y}_{n-1}) &= \int l_{(t_{n-1}, t_n]}(\bar{y}_{n,1}; \bar{x}_n) \mathbf{p}(\bar{x}_{n,1}) \pi_{n-1}(\bar{x}_{n-1} | \bar{y}_{n-1}) d\bar{x}_n. \tag{5}
 \end{aligned}$$

At time $n \geq 2$ of the algorithm, a reversible jump MCMC kernel (although the analysis below is not restricted to such scenarios) is used for N steps to sample from the approximated density

$$\pi_n^N(\bar{x}_n|\bar{y}_n) \propto l_{(t_{n-1}, t_n]}(\bar{y}_{n,1}; \bar{x}_n) \mathbf{p}(\bar{x}_{n,1}) S_{\bar{x}, n-1}^N(\bar{x}_{n-1})$$

where $S_{\bar{x}, n-1}^N(\bar{x}_{n-1}) := \frac{1}{N} \sum_{i=1}^N \mathbb{I}_{\{\bar{X}_{n-1}^{(i)}\}}(\bar{x}_{n-1})$ with $\bar{X}_{n-1}^{(1)}, \dots, \bar{X}_{n-1}^{(N)}$ obtained from a reversible jump MCMC algorithm of invariant measure π_{n-1}^N . The algorithm for $n = 1$ targets π_1 exactly; there is no empirical density $S_{\bar{x}, 0}^N$. At time $n = 1$, the algorithm starts from an arbitrary point $\bar{x}_1^{(1)} \in \bar{E}_1$. For $n \geq 2$ the initialization is from a draw from the empirical $S_{\bar{x}, n-1}^N$ and the prior \mathbf{p} (this can be modified); $N - 1$ additional samples are simulated.

The above algorithm can be justified, theoretically, using the Poisson equation (e.g. Glynn and Meyn 1996) and induction arguments. Below the assumption (A) is made; see appendix for the assumption (A) as well as the proof. The expectation below is w.r.t. the simulated process discussed above, given the observed data.

Proposition 1 *Assume (A). Then for any $n \geq 1$, $\bar{y}_n, p \geq 1$ there exists $B_{p,n}(\bar{y}_n) < +\infty$ such that for any $f_n \in \mathcal{B}_b(\bar{E}_n)$*

$$\mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \frac{1}{N} \sum_{i=1}^N f_n(\bar{X}_n^{(i)}) - \int_{\bar{E}_n} f_n(\bar{x}_n) \pi_n(d\bar{x}_n) \right|^p \middle| \bar{y}_n \right]^{1/p} \leq \frac{B_{p,n}(\bar{y}_n) \|f_n\|}{\sqrt{N}}. \tag{6}$$

This result helps to establish the theoretical validity of the method in Centanni and Minozzo (2006a), which to our knowledge had not been established in that paper or elsewhere. In addition, it allows us to understand where and when the method may be of use; this is discussed in Sect. 3.2.

3.1 SMC methods

SMC samplers aim to approximate a sequence of related probability measures $\{\pi_n\}_{0 \leq n \leq m^*}$ defined upon a common space (E, \mathcal{E}) . Note that $m^* > 1$ can depend upon the data and may not be known prior to simulation. For partially observed PPs the probability measures are defined upon nested state-spaces; this case can be similarly handled with minor modification. SMC samplers introduce a sequence of auxiliary probability measures $\{\tilde{\pi}_n\}_{0 \leq n \leq m^*}$ on state-spaces of increasing dimension $(E_{[0,n]} := E_0 \times \dots \times E_n, \mathcal{E}_{[0,n]} := \mathcal{E}_0 \otimes \dots \otimes \mathcal{E}_n)$, such that they admit the $\{\pi_n\}_{0 \leq n \leq m^*}$ as marginals.

The following sequence of auxiliary densities is used:

$$\tilde{\pi}_n(x_{0:n}) = \pi_n(x_n) \prod_{j=0}^{n-1} L_j(x_{j+1}, x_j) \tag{7}$$

where $\{L_n\}_{0 \leq n \leq m^*-1}$ are backward Markov kernels. In our application π_0 is the prior, on E_1 (as defined below). It is clear that (7) admits the $\{\pi_n\}$ as marginals, and hence these distributions can be targeted using precisely the same mechanism as in sequential importance sampling/resampling; the algorithm is given in Algorithm 1.

The ESS in Algorithm 1 refers to the effective sample size (ESS) (Liu 2001). This measures the weight degeneracy of the algorithm; if the ESS is close to N , then this indicates that all of the samples are approximately independent. This is a standard metric by which to assess the performance of the algorithm. The resampling method used throughout the paper is systematic resampling.

One generic approach is to set K_n as an MCMC kernel of invariant distribution π_n and L_{n-1} as the reversal kernel $L_{n-1}(x_n, x_{n-1}) = \pi_n(x_{n-1})K_n(x_{n-1}, x_n)/\pi_n(x_n)$ which we term the *standard reversal kernel*. One can iterate the MCMC kernels, by which we use the positive integer M to denote the number of iterates. It is also possible to apply the algorithm when K_n is a mixture of kernels; see Del Moral et al. (2006) for details.

Algorithm 1 A generic SMC sampler. Note that $T(N)$ is termed a threshold function such that $1 \leq T(N) \leq N$ and ESS is the effective sample size

- 0. Set $n = 0$; for $i = 1, \dots, N$ sample $X_0^{(i)} \sim \varrho_0$ and compute $w_0(X_0^{(i)}) \propto \pi_0(X_0^{(i)})/\varrho_0(X_0^{(i)})$.
- 1. Compute the normalized importance weights,

$$w_n^{(i)} = \frac{w_n(X_{0:n}^{(i)})}{\sum_{j=1}^N w_n(X_{0:n}^{(j)})}$$

if the $ESS = \{\sum_{j=1}^N w_n(X_{0:n}^{(j)})\}^2 / \sum_{j=1}^N \{w_n(X_{0:n}^{(j)})\}^2 < T(N)$ then resample the particles and set the importance weights to uniform. Set $n = n + 1$, if $n = m^* + 1$ stop.

- 2. For $i = 1, \dots, N$ sample $X_n^{(i)} | X_{n-1}^{(i)} = x_{n-1}^{(i)} \sim K_n(x_{n-1}^{(i)}, \cdot)$, and compute:

$$W_n(X_{n-1:n}^{(i)}) \propto \frac{\pi_n(X_n^{(i)})L_{n-1}(X_{n-1}^{(i)}, X_n^{(i)})}{\pi_{n-1}(X_{n-1}^{(i)})K_n(X_{n-1}^{(i)}, X_n^{(i)})} \tag{8}$$

$w_n(X_{0:n}^{(i)}) = W_n(X_{n-1:n}^{(i)})w_{n-1}(X_{0:n-1}^{(i)})$ and return to the start of 1.

3.1.1 Nested spaces

As described in Sect. 1, in complex problems it is often difficult to design efficient SMC algorithms. In the example in Sect. 2, the state-spaces of the subsequent densities are not common. The objective is to sample from a sequence of densities on the space, at time n ,

$$E_n = \left(\bigcup_{k \in \mathbb{N}_0} \{k\} \times \Phi_{k,t_n} \times (\mathbb{R}^+)^k \right) \times \mathbb{R} \times \mathbb{R}^+ \quad 1 \leq n \leq m^* - 1$$

with $E_0 = E_1$. That is, for any $1 \leq n \leq m^* - 1$, $E_n \subseteq E_{n+1}$. Two standard methods for extending the space, as in [Del Moral et al. \(2006\)](#) are to propagate particles by application of ‘birth’ and the ‘extend’ moves.

Consider the model in Sect. 2. The following SMC steps are used to extend the space at time n of the algorithm.

- *Birth.* A new jump is sampled uniformly in $[\phi_{k_{n-1}}, t_n]$ and a new mark from the prior. The incremental weight is

$$W_n(\bar{x}_{n-1:n}, \mu, \sigma) \propto \frac{\pi_n(\bar{x}_n, \mu, \sigma | \bar{y}_n)(t_n - \phi_{k_{n-1}})}{\pi_{n-1}(\bar{x}_{n-1}, \mu, \sigma | \bar{y}_n) \mathbf{p}(\zeta_{k_{t_n}})}.$$

- *Extend.* A new jump is generated according to a Markov kernel that corresponds to the random walk:

$$\log \left\{ \frac{\phi_{k_{t_n}} - \phi_{k_{t_n-1}}}{t_n - \phi_{k_{t_n}}} \right\} = \vartheta Z + \log \left\{ \frac{\phi_{k_{t_{n-1}}} - \phi_{k_{t_{n-1}-1}}}{t_n - \phi_{k_{t_{n-1}}}} \right\}$$

with $Z \sim \mathcal{N}(0, 1)$, $\vartheta > 0$. The new mark is sampled from the prior. The backward kernel and incremental weight are discussed in [Del Moral et al. \(2007\)](#), Sect. 4.3.

Note, as remarked in [Whiteley et al. \(2011\)](#), we need to be able to sample any number of births. With an extremely small probability, a proposal from the prior is included to form a mixture kernel.

In addition to the above steps an MCMC sweep is included after the decision of whether or not to resample the particles is taken (see step 1. of Algorithm 1): an MCMC kernel of invariant measure π_n is applied. The kernel is much the same as in [Green \(1995\)](#).

3.1.2 Simulation experiment

We applied the benchmark sampler, as detailed above, to some synthetic data in order to monitor the performance of the algorithm. Standard practice in the reporting of financial data is to represent the time of a trade as a positive real number, with the integer part representing the number of days passed since January 1st 1900 and the non-integer part representing the fraction of 24 h that has passed during that day; thus, 1 min corresponds to an interval of length $1/1,440$. Therefore we use a synthetic data set with intensity of order of magnitude 10^3 . The ticks ω_i were generated from a specified intensity process $\{\lambda_t\}$ that varied smoothly between three levels of constant intensity at $\lambda = 6,000$, $\lambda = 2,000$ and $\lambda = 4,000$. The log returns ξ_i were sampled from the Cauchy-distribution, location $\mu = 0$ and scale $\sigma = 2.5 \times 10^{-4}$. The entire data set was of size $r_T = 3,206$, $[0, T] = [0, 0.9]$ with $t_n = n * 0.003$. The intensity from which they were generated had constant levels at 6,000 in the interval $[0.05, 0.18]$; at 4,000 in the interval $[0.51, 0.68]$; and at 2,000 in the intervals $[0.28, 0.42]$ and $[0.78, 0.90]$.

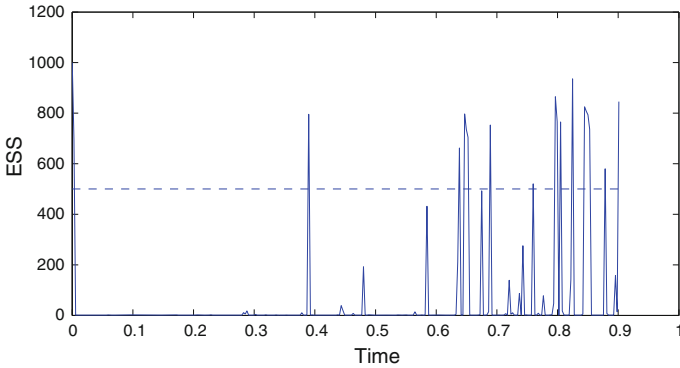


Fig. 1 Effective sample size plots for the SMC sampler described in Algorithm 1, implemented with $N = 1,000$ particles and with $M = 5$ MCMC sweeps at each iteration. The dashed line indicates the resampling threshold at $N/2 = 500$ particles; resampling is needed at 94.4% of the time steps

The sampler was implemented with all combinations $\{(M, N)\}$ for $N \in \{100, 1,000\}$ and $M \in \{1, 5, 20\}$, resampling whenever the effective sample size fell below $N/2$ (recall N is the number of particles and M the MCMC iterations). When performing statistical inference, the intensity (3) used parameters $\gamma = 0.001$, $\nu = 150$ and $s = 20$.

It was found that for this SMC sampler, the system consistently collapses to a single particle representation of the distribution of interest within an extremely short time period. That is, resampling is needed at almost every time step, which leads to an extremely poor representation of the target density. Figure 1 shows the ESS at each time step for a particular implementation. As can be seen, the algorithm behaves extremely poorly for this model.

3.2 Discussion

We have reviewed two existing techniques for the analysis of partially observed PPs. It should be noted that there are other methods, for example in Varini (2007). In that paper, the intensity has a finite number of functional forms and the uncertainty is related to the type of form at each inference time t_n .

The relative advantage of the approach of Centanni and Minozzo (2006a), against SMC samplers, is the fact that the state-space need not be extended. On page 1586 of Centanni and Minozzo (2006a) the authors describe the filtering/smoothing algorithm, for the process on the entire interval $[0, t_n]$ at time n ; the theory discussed in Proposition 1 suggests that this method is not likely to work well as n grows. The bound, which is perhaps a little loose is, for $n \geq 2$

$$B_{p,n}(\bar{y}_n) = \frac{2}{\epsilon_n(\bar{y}_n)} [B_p + 1] + \hat{k}_n B_{p,n-1}(\bar{y}_{n-1})$$

with $B_{p,1}(\bar{y}_1) = \frac{2}{\epsilon_1(\bar{y}_1)} [B_p + 1]$, B_p a constant related to the Bürkholder/Davis inequalities (e.g. Shiryaev 1996), $\epsilon_n(\bar{y}_n) \in (0, 1)$ and $\hat{k}_n > 0$ a constant that is model/data

dependent which is possibly bigger than 1. The bound indicates that the error can increase over time, even under the exceptionally strong assumption (A) in appendix. This is opposed to SMC methods which are provably stable, under similar assumptions (and that the entire state is updated), as $n \rightarrow \infty$ (Del Moral 2004). In other words, whilst the approach of Centanni and Minozzo is useful in difficult problems, it is less general with potentially slower convergence rate than SMC. Intuitively, it seems that the method of Centanni and Minozzo (2006a) is perhaps only useful when considering the process on $(t_{n-1}, t_n]$, as the process further back in time is not rejuvenated in any way. As a result, static parameter estimation may not be very accurate. In addition, the method cannot be extended to a sequential algorithm such that fully Bayesian inference is possible. As noted above, SMC samplers can be used in such contexts, but require a computational budget that grows with the time parameter n .

As mentioned above, SMC methods are provably stable under some conditions as the time parameter grows. However, some remarks related to the method in Algorithm 1 can help to shed some light on the poor behaviour in Sect. 3.1.2. Consider the scenario when one is interested in statistical inference on $[0, t_1]$. Suppose for simplicity, one can write the posterior on this region as

$$\pi(\bar{x}_1) \propto \exp \left\{ \sum_{i=1}^{r_{t_1}} g_i(y_i; \bar{x}_1) \right\} \mathbf{p}(\bar{x}_1) \tag{9}$$

for fixed r_{t_1}, μ, σ , with $g_i : \mathbb{R}^+ \times \Xi \times \bar{E}_1 \rightarrow \mathbb{R}$. If one considers just pure importance sampling, then conditioning upon the data, one can easily show that for any π_1 -(square) integrable f with $\int f(\bar{x}_1)\mathbf{p}(\bar{x}_1)d\bar{x}_1 = 0$, the asymptotic variance in the associated Gaussian central limit theorem is lower-bounded by:

$$\left(\int f(\bar{x}_1)^2 \exp \left\{ 2 \sum_{i=1}^{r_{t_1}} g_i(y_i; \bar{x}_1) \right\} p(\bar{x}_1)d\bar{x}_1 \right) / \left(\int \exp \left\{ \sum_{i=1}^{r_{t_1}} g_i(y_i; \bar{x}_1) \right\} p(\bar{x}_1)d\bar{x}_1 \right).$$

Then, for any mixing type sequence of data the asymptotic variance will for some f and in *some* scenarios, grow without bound as r_{t_1} grows—this is a very heuristic observation that requires further investigation. Hence, given this discussion and our empirical experience, it seems that we require a new methodology, especially for complex problems.

3.3 Possible solutions to the problems of extending the state-space

An important remark associated with the simulations in Sect. 3.1.2 is that it cannot be expected that simply increasing the number of particles will necessarily a significantly better estimation procedure. The algorithm completely crashes to a single particle and it seems that naively increasing computation will not improve the simulations.

As discussed above, the inherent difficulty of sampling from the given sequence of distributions is that of extending the state-space. It is known that conditional on all parameters except the final jump, the optimal importance distribution is the full

conditional density (Del Moral et al. 2006). In practice, for many problems it is either not possible to sample from this density or to evaluate it exactly (which is required). In the case that it is possible to sample from the full conditional, but the normalizing constant is unknown, the normalizing constant problem can be dealt with via the random weight idea (Rousset and Doucet 2006). In the context of this problem we found that the simulation from the full conditional density of $\phi_{k_{in}}$ was difficult, to the extent that sensible rejection algorithms and approximations for the random weight technique were extremely poor.

Another solution, in Del Moral et al. (2007), consists of stopping the algorithm when the ESS drops and using an additional SMC sampler to facilitate the extension of the state-space. However, in this example, the ESS is so low, that it cannot be expected to help. Due to above discussion, it is clear that a new technique is required to sample from the sequence of distributions; two ideas are presented below. One idea, in the context of estimating static parameters, that could be adopted is SMC² (Chopin et al. 2012) which has appeared after the first versions of this article.

4 Proposed methods

In the following section, two approaches are presented to deal with the problems in Sect. 3.1.2. First, a state-space saturation approach, where sampling of PP trajectories is performed over a state space corresponding to a fixed observation interval. Second, a data-point tempering approach. In this approach, as the time parameter increases, the (artificial) target in the new region is simply the prior and the data are then sequentially added to the likelihood, softening the state-space extension problem. Both of these procedures use the basic structure of Algorithm 1, with some refinements, that are mentioned in the text. As for the procedure in Algorithm 1 we add dynamic resampling steps; when MCMC kernels are used, one can resample before sampling—see Del Moral et al. (2006) for details.

4.1 Saturating the state-space

A simple idea, which has been used in the context of reversible jump, is to saturate the state-space. The idea relies upon knowing the observation period of the PP $([0, T])$ a priori to the simulation. This is realistic in a variety of applications. For example, in Sect. 2, often we may only be interested in performing inference for a day of trading and thus can set $[0, T]$.

In detail, it is proposed to sample, in the case of the example in Sect. 2, from the sequence of target densities defined on the space

$$E = \left(\bigcup_{k \in \mathbb{N}_0} \{k\} \times \Phi_{k,T} \times (\mathbb{R}^+)^k \right) \times \mathbb{R} \times (\mathbb{R}^+)^2. \tag{10}$$

The (marginal, that is in the sense of (7)) target densities are now, denoted with a S as a super-script:

$$\begin{aligned} \pi_n^S(\bar{x}_n, \mu, \sigma | \bar{y}_n) &\propto \prod_{i=1}^{r_{t_n}} \{ \tilde{p}(\xi_i; \mu, \sigma) \lambda_{\omega_i} \} \exp \left\{ - \int_0^{t_n} \lambda_u du \right\} \\ &\times \prod_{i=1}^{k_{t_n}} \{ \mathbf{p}(\xi_i) \} \mathbf{p}^S(\phi_{1:k_{t_n}}) \mathbf{p}^S(k_{t_n}) \times \tilde{p}(\mu, \sigma) \quad 1 \leq n \leq T \end{aligned}$$

where the prior on the point process is:

$$\begin{aligned} \mathbf{p}^S(\phi_{1:k_{t_n}}) &= \frac{1}{k_{t_n}!} \mathbb{I}_{\{\phi_{1:k_{t_n}}: 0 < \phi_1 < \dots < \phi_{k_{t_n}}\}}(\phi_{1:k_{t_n}}) \mathbb{I}_{\mathbb{N}}(k_{t_n}) + \mathbb{I}_{\{0\}}(k_{t_n}) \\ \mathbf{p}^S(k_{t_n}) &= \frac{(\gamma T)^{k_{t_n}} e^{-\gamma T}}{k_{t_n}!}. \end{aligned}$$

We then use, for K_n , an MCMC kernel of invariant measure π_n^S and the standard reversal kernel discussed in Sect. 3.1 for the backward kernel. The initial distribution is the prior and the weight at time 0 is proportional to 1 for each particle. The incremental weights at subsequent time-points are simply:

$$W_n(\bar{x}_{n-1}, \mu_{n-1}, \sigma_{n-1}) \propto \frac{\pi_n^S(\bar{x}_{n-1}, \mu_{n-1}, \sigma_{n-1} | \bar{y}_n)}{\pi_{n-1}^S(\bar{x}_{n-1}, \mu_{n-1}, \sigma_{n-1} | \bar{y}_{n-1})} \quad 1 \leq n \leq T.$$

Inference w.r.t. the original $\{\pi_n\}_{1 \leq n \leq m^*}$ can be performed via IS as the supports of the targets of interest are contained within the proposals (i.e. via the targets of the saturated algorithm).

4.2 Data-point tempering

A simple solution to the state-space extension problem, which allows data to be incorporated sequentially, albeit not being of fixed computational complexity is as follows. When the time parameter increases, the new part of the process is simulated according to the prior. Then each new data point is added to the likelihood in a sequential manner. In other words if there are n data points, then there are $m^* = n + \tilde{m}$ time-steps of the algorithm.

To illustrate, consider only the scenario of the data in $[0, r_{t_1}]$, with $r_{t_1} > 0$. Then our sequence of (marginal) targets are: $\pi_0^{\text{TE}}(\bar{x}_1, \mu, \sigma) = \mathbf{p}(\bar{x}_1) \tilde{p}(\mu) \tilde{p}(\sigma)$ and for $1 \leq n \leq r_{t_1}$

$$\pi_n^{\text{TE}}(\bar{x}_1, \mu, \sigma | y_{1:n}) \propto \prod_{i=1}^n \{ \tilde{p}(\xi_i; \mu, \sigma) \lambda_{\omega_i} \} \exp \left\{ - \int_0^{r_{t_1}} \lambda_u du \right\} \mathbf{p}(\bar{x}_1) \tilde{p}(\mu) \tilde{p}(\sigma).$$

Then, when considering the extension of the point-process onto $[0, t_2]$, one has a (marginal) target that is:

$$\pi_{r_{t_1}+1}^{\text{TE}}(\bar{x}_2, \mu, \sigma | \bar{y}_1) \propto \prod_{i=1}^{r_{t_1}} \{ \tilde{p}(\xi_i; \mu, \sigma) \lambda_{\omega_i} \} \exp \left\{ - \int_0^{t_1} \lambda_u du \right\} \mathfrak{p}(\bar{x}_2) \tilde{p}(\mu) \tilde{p}(\sigma)$$

When one extends the state-space, we sample from the prior on the new segment, which leads to a unit incremental weight (up-to proportionality)—no backward kernel is required here. Then, when adding data, we simply use MCMC kernels to move the particles (the kernels as in Sect. 3.1.1) and the standard reversal kernel discussed in Sect. 3.1 for the backward kernel. This leads to an incremental weight that is the ratio of the consecutive densities at the previous state.

The potential advantage of this idea is that, when extending the state-space, there is no extra data, to potentially complicate the likelihood. Thus, it is expected that if the prior does not propose a significant number of new jumps that the incremental weights should be of relatively low variance. The subsequent steps, when considering the jumps in $[t_n, t_{n+1})$ are performed on a common state-space and hence should not be subject to as substantial variability as when the state-space changes. This idea could also be adapted to the case that the likelihood on the new interval is tempered instead (e.g. Jasra et al. 2007).

As a theoretical investigation of this idea, we return to the discussion of Sect. 3.2 and in particular, where the joint target density is (9). We consider the data-point tempering which starts with a draw from the prior and sequentially adds data points. In other words, runs for $r_{t_1} + 1$ time-steps with

$$\pi_n(\bar{x}_1) \propto \exp \left\{ \sum_{i=1}^n g_i(y_i; \bar{x}_1) \right\} \mathfrak{p}(\bar{x}_1)$$

with a $-\infty < \underline{g} < \bar{g} < \infty$ such that for each i, y_i and all $\bar{x}_1, \underline{g} \leq g_i(y_i; \bar{x}_1) \leq \bar{g}$. The algorithm resamples at every time-step and uses MCMC kernels, which are assumed to satisfy, for some $\tau \in (0, 1)$, and each $1 \leq n \leq r_{t_1}, r_{t_1}, \bar{x}_1, \bar{x}'_1$

$$K_n(\bar{x}_1, \cdot) \geq \tau K_n(\bar{x}'_1, \cdot).$$

At the very final time-step one also resamples after the final weighting of the particles. Write $\bar{X}_1^1, \dots, \bar{X}_1^N$ as the samples that approximate target (9). Suppose $f \in \mathcal{B}_b(\bar{E}_1)$, then there is a Gaussian central limit theorem for

$$\sqrt{N} \left(\frac{1}{N} \sum_{i=1}^N f(\bar{X}_1^i) - \int_{\bar{E}_1} f(\bar{x}_1) \pi_{r_{t_1}}(\bar{x}_1) d\bar{x}_1 \right).$$

Writing the asymptotic variance as $\sigma_{\text{TE}, r_{t_1}}^2(f)$, we have the following result whose proof is in appendix.

Proposition 2 *For SMC sampler described above, with final target (9) then we have for any $f \in \mathcal{B}_b(\bar{E}_1)$ that there exists a $B \in (0, +\infty)$ such that for any $r_{t_1} \geq 1, \bar{y}_1$*

$$\sigma_{\text{TE}, r_{t_1}}^2(f) \leq B.$$

The upper-bound does not grow with the number of data. That is, by increasing the computational complexity linearly in the number of data, one has an algorithm whose error does not grow as more data (and regions) are added. This is similar to the observation of Beskos et al. (2011), when increasing the dimension of the target density. We note that the result is derived under exceptionally strong assumptions. In general, when one considers r_{t_1} growing, one requires sharper tools than the Dobrushin coefficients used here (e.g. Eberle and Marinelli 2012); this is beyond the scope of the current article and our result above is illustrative (and hence potentially over-optimistic).

4.3 Online implementation

A key characteristic that has not yet been addressed is the fact that each approach has a computational complexity that is increasing with time. In a procedure that would otherwise be well suited to providing online inference, this is an unattractive feature. A large contribution to this increasing computational budget derives from the MCMC sweeps at the end of each iteration. As the space over which the invariant MCMC kernel is being applied is increased, so does the expense of the algorithm. An improvement to the computational demand of the samplers can therefore be made by keeping the space over which the MCMC kernel is applied constant. The *reduced computational complexity* (RCC) alternative to each of the samplers is also designed by amending the algorithms such that, at time t_n , the MCMC sweep operates over, at most, 20 changepoints, i.e. over the interval $[\phi_{k_n-19}, t_n)$. Due to the well-known path degeneracy problem in SMC (see Kantas et al. 2011), the estimates will be poor approximations of the true values, when including static parameters and extending the space of the point process for a long time. We note, at least for our application, it is reasonable to consider T fixed and thus, this is less problematic.

5 The finance problem revisited

We now return to the example from Sect. 2 and the settings as in Sect. 3.1.2.

5.1 Simulated data

The saturated and tempered samplers, as well as their RCC alternatives, were implemented using the simulated data set (in Sect. 3.1.2), in order to compare their respective performances against the benchmark sampler and to compare the accuracy of the resulting intensity estimates against an observed intensity process. All of the alternative samplers were implemented under the same conditions, using the algorithm and model parameters as described for the implementation of the benchmark sampler. All results are averaged over 10 runs of the algorithm.

In assessing the performance of the sampler, quantities of interest are, once again, the resampling rate and the processing time, as well as the minimum ESS recorded throughout the execution of the sampler. The resampling rates for all three samplers

Table 1 Table showing the resampling rates of each of the three SMC samplers and their reduced computational complexity alternatives, for the six algorithm parameterizations that were tested

	$M = 1$ (%)		$M = 5$ (%)		$M = 20$ (%)	
	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$
Benchmark	31.3	52.0	42.3	94.4	74.0	99.7
Benchmark-RCC	37.6	88.1	69.0	99.7	99.4	99.7
Saturated	21.0	21.3	19.7	20.1	18.2	17.6
Saturated-RCC	20.7	20.7	18.5	18.8	15.4	15.4
Tempered	2.0	2.0	1.9	1.9	1.7	1.7
Tempered-RCC	2.0	2.0	1.7	1.8	1.4	1.4

The ESS plots for the saturated and tempered samplers with $N = 1,000$, $M = 5$ are given in Fig. 2 for comparison with the corresponding ESS plot for the benchmark sampler, given in Fig. 1

Table 2 Table showing the minimum ESS encountered during implementation by each of the three SMC samplers and their reduced computational complexity alternatives, for the six algorithm parameterizations that were tested

	$M = 1$		$M = 5$		$M = 20$	
	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$
Benchmark	1.0	1.0	1.0	1.0	1.0	1.0
Benchmark-RCC	1.0	1.0	1.0	1.0	1.0	1.0
Saturated	38.1	410.2	38.6	397.0	38.6	398.9
Saturated-RCC	38.5	401.2	40.6	394.4	43.0	425.9
Tempered	47.6	484.7	47.7	475.5	47.9	483.4
Tempered-RCC	47.8	475.7	48.4	481.7	48.3	486.6

and their RCC alternatives are presented in Table 1, with the corresponding minimum ESS's attained recorded in Table 2 and the corresponding processing times in Table 3. Figure 2 displays the evolution of the ESS over a particular run of the algorithm. Figure 3 shows the estimated intensity at each time t_n , given data up to time t_n . From Table 1, it is clear to see that, for the saturated and tempered samplers, an increase in M results in a decrease in the resampling rates, i.e. a decrease in sampler degeneracy, as expected. It is also plain to see from Table 2 that, as N increases, so does the minimum ESS, and thus the reliability of the estimates. From Tables 1, 2, and Fig. 3, and comparing Figs. 1 and 2, it is clear that the saturated and tempered samplers significantly outperformed the benchmark sampler.

We use the posterior medians to report intensities. Since we have access to a ‘true’ intensity process, the accuracy of these estimated intensity process is measured using the root mean square error (RMSE). Table 4 presents the RMSEs of the intensity estimates (given the data up to t_n , averaged over each t_n) and Table 5 presents the RMSEs of the smoothed (conditional upon the entire data set) intensity estimates resulting from each of the three samplers and their RCC alternatives. The most important result to note is the performance of the saturated and tempered samplers in comparison with

Table 3 Table showing the processing time, in seconds, for each of the three samplers and their reduced computational complexity alternatives, for the six algorithm parameterizations that were tested

	$M = 1$		$M = 5$		$M = 20$	
	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$
Benchmark	612.9	9,689.1	2,849.7	45,690.4	13,352.1	144,621.3
Benchmark-RCC	449.0	7,910.9	1,132.7	10,657.6	3,106.2	31,208.5
Saturated	1,125.3	10,667.8	3,234.3	39,061.1	15,381.9	141,817.3
Saturated-RCC	637.5	6,215.2	1,200.7	11,412.6	4,391.9	47,662.8
Tempered	1,160.2	10,633.4	3,138.4	38,679.6	14,086.7	130,899.1
Tempered-RCC	666.0	6,424.4	1,156.3	11,209.1	3,231.3	34,795.3

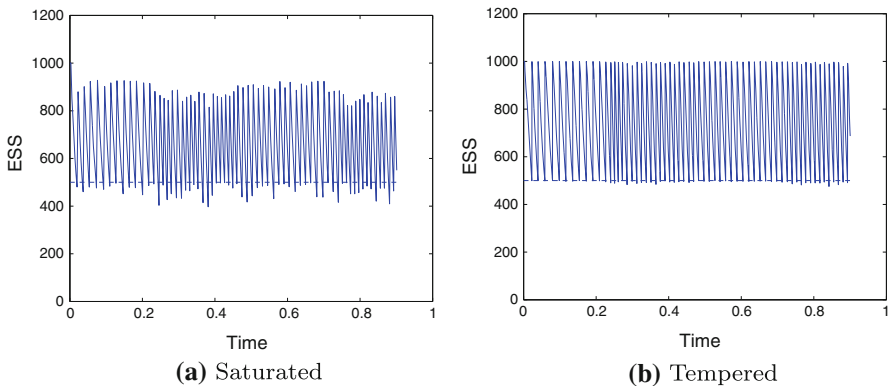


Fig. 2 Effective sample size plots for the SMC samplers with state space saturation (*left*) and data point tempering (*right*), run with $N = 1,000$ particles and with $M = 5$ MCMC sweeps at each iteration. The *dashed line* indicates the resampling threshold at $N/2 = 500$ particles; the corresponding resampling rates are 20.1% for the saturated sampler and 1.9% for the tempered sampler

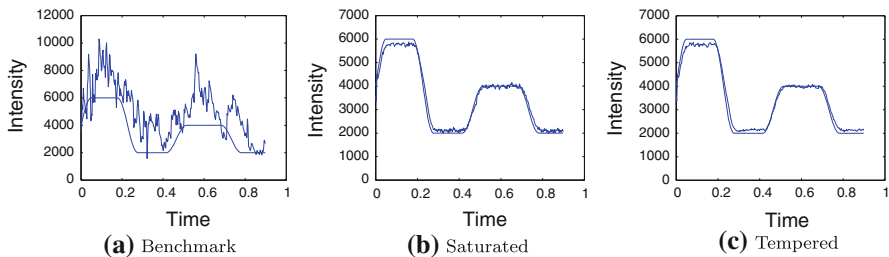


Fig. 3 Estimates (given the data up to t_n) of the intensity of a simulated data set, generated by the benchmark SMC sampler (*left*) and the samplers with state space saturation (*centre*) and data point tempering (*right*), run with $N = 1,000$ particles and with $M = 5$ MCMC sweeps at each iteration. The model parameters were $\gamma = 0.001$, $\nu = 150$ and $s = 20$

the benchmark sampler. As can be seen in terms of accuracy for intensity estimates, the two proposed alterations to the sampler improve the performance consistently and significantly. Looking at the resampling rates and processing times, in Tables 1 and 3,

Table 4 Table showing the root mean square error of the intensity

	$M = 1$		$M = 5$		$M = 20$	
	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$
Benchmark	688.561	1,116.639	620.432	1,942.992	1,330.232	1,501.263
Benchmark-RCC	676.932	2,026.956	880.824	2,247.313	1,472.126	1,264.533
Saturated	242.834	192.580	228.390	193.778	237.315	198.223
Saturated-RCC	229.449	189.279	224.692	193.379	225.592	194.623
Tempered	254.396	196.928	247.754	201.681	248.367	202.501
Tempered-RCC	256.012	191.407	227.241	197.043	230.805	200.227

This is given the data up to t_n , averaged over each t_n and for each of the three samplers and their reduced computational complexity alternatives, for the six algorithm parameterizations that were tested

Table 5 Table showing the smoothed root mean square error of the intensity

	$M = 1$		$M = 5$		$M = 20$	
	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$	$N = 100$	$N = 1,000$
Benchmark	768.702	670.656	495.019	627.909	489.243	571.107
Benchmark-RCC	698.640	1,034.890	572.794	572.841	535.004	599.031
Saturated	360.794	264.331	296.953	114.064	153.444	89.397
Saturated-RCC	478.871	265.477	405.767	266.980	468.853	205.243
Tempered	350.015	170.321	271.712	128.078	157.709	81.666
Tempered-RCC	485.825	249.529	475.348	193.898	514.107	180.914

The entire data set is given and for each of the three samplers and their reduced computational complexity alternatives, for the six algorithm parameterizations that were tested

respectively, we can see that, as expected, although the tempered sampler resampled the particles significantly less than the benchmark sampler, the individual incorporation of each data point resulted in a greater computational cost. These two aspects of the benchmark and tempered samplers appear to have countered each other, resulting in their processing times being largely similar.

We consider also the effect that changes in M and N have on the accuracy of estimates provided by the saturated and tempered samplers. For the saturated and tempered samplers, the results in Tables 4 and 5 corroborate the expected improvement in accuracy, in both for the sequential estimates at t_n given data up-to t_n and smoothed estimates (given the entire data), that results from an increase in the number of particles used. Whilst for the sequential estimates, there is no clear improvement in accuracy with increasing M , an improvement can be seen in the accuracy of the smoothed estimates.

Finally, using the simulated data, we consider the performance of the samplers when limiting the space over which the invariant MCMC kernels are applied, i.e. the RCC alternatives. As can be seen from Table 4, the RCC alteration does not sacrifice any accuracy in the estimates of the intensity (given the data up to each time t_n); however, it can be seen from Table 5 that the accuracy of the smoothed intensity estimates is

rather poor. This is to be expected, due to path degeneracy; we note that one cannot estimate static parameters with the RCC approach unless the time window T is quite small.

5.2 Real data

All three samplers were also tested on real financial data, with the RCC alternatives also being used to generate intensity estimates, given the data up to t_n : the share price of ARM Holdings, plc., traded on the LSE was used. The entire data set was of size $r_T = 1819$, $[0, T] = [0, 0.3]$ (represents 3/10 of a trading day, that is, 3/10 of 24 hours; the first trade is just after 9 am and the last around 16:15.) with $t_n = n * 0.001$. Genuine financial data are likely to correspond to a more volatile latent intensity process than that which was used to generate the synthetic data set, and so the parameterization of the target posterior should be chosen such that large jumps in the intensity process are possible, and such that the intensity may also revert quickly to a lower intensity level. Hence, we specify: $\{\gamma, \nu, s\} = \{0.001, 500, 250\}$. Each of the samplers were run using $N = 1,000$ particles, applying $M = 5$ MCMC sweeps at each iteration, whilst the resampling rates and the minimum ESS obtained for each procedure were monitored to ensure that the algorithms did not collapse.

Clearly, there is no ‘known’ intensity process against which to compare the point-wise estimates produced by the samplers. In addition, any inverse-duration based representation of the intensity against which useful comparisons could be drawn would involve making assumptions on the smoothness of the intensity process itself. Thus, we turn to measuring the one-step-ahead predictive accuracy of the estimators of the intensity. This is achieved as follows: denoting the intensity estimated over the interval $[t_{n-j}, t_n]$ as $\hat{\lambda}_{n,j}$, one predicts the expected number of ticks in the interval $[t_{n+i-j}, t_{n+i}]$ as $(\hat{\lambda}_{n,j})^{-1}$ for $i \geq 1$ and $j \geq 1$, where j is the number of periods over which the prediction is made and i is a lag index. The prediction errors are then calculated based on the predicted and observed number of ticks in the period t_{n+i-j}, t_{n+i} ; the root mean square prediction error (RMSPE) will be used. We will report on the one-step-ahead estimates ($i = 1$), estimating the intensity over each interval with $j = 1$.

Table 6 Table showing the root mean square prediction errors for the intensity estimates [given data up to time t_n and entire data (smoothed)] given by each of the three samplers for the parameter values $N = 1,000$, $M = 5$

	Data t_n RMSPEs	Smoothed RMSPEs	Processing times (s)	Resampling rates (%)
Saturated	2.18876	2.13479	4,064.5	39.5
Saturated-RCC	2.19112	–	2,193.1	39.9
Tempered	2.34671	2.11468	4,605.5	19.8
Tempered-RCC	2.42776	–	2,237.3	19.9

The RMSPEs for the smoothed intensity estimates given by the RCC alternatives to the samplers are also provided, along with the observed processing times and resampling rates for each sampler

Table 6 presents the RMSPEs for the intensity estimates resulting from the samplers and the RCC alternatives. It was observed that, in calculating the RMSPEs for lag indices $i = 1, \dots, 100$ using each sampler, both the saturated and the tempered samplers displayed the smallest error at $i = 1$, i.e. their respective one-step-ahead predictions were more accurate than those made for lags up to 2.64 h (each observation interval corresponds to 0.0264 days = 1.584 min).

The RCC samplers provide significant computational savings and do not seem to degrade substantially, w.r.t. the error criteria. Again, we remark that, in general, one should not trust the estimates of the RCC, but as seen here, they can provide a guideline for the intensity values.

6 Summary

In this paper, we have considered SMC simulation for partially observed point processes and implemented them for a particular doubly stochastic PP. Two solutions were given, one based upon saturating the state-space, which is suitable in a wide variety of applications and data-point tempering which can be used in sequential problems. We also discussed RCC versions of these algorithms, which reduce computation, but will be subject to the path degeneracy problem when including static parameters and considering the smoothing distribution. We saw that the methods can be successful, in terms of weight degeneracy versus the benchmark approach detailed in [Del Moral et al. \(2007\)](#). In addition, for real data it was observed that predictions using the RCC could be reasonable (relative to the normal versions of the algorithms), but caution on using these estimates should be used.

The methodology we have presented is not online. As we have seen, when one modifies the approaches to have fixed computational complexity, the path degeneracy problem occurs and one cannot deal with scenario with static parameters. In this case, we are working with Dr. N. Whiteley on a technique based upon fixed window filtering. This is an on-line algorithm which allows data to be incorporated as they arrive with computational cost which is non-increasing over time, but is biased. The approach involves sampling from a sequence of distributions which are constructed such that, at time t_n , previously sampled events in $[0, t_{n-\ell}]$ can be discarded. In order to be exact (in the sense of targeting the true posterior distributions), this scheme would involve point-wise evaluation of an intractable density. We are working on a sensible approximation of this density, at the cost of introducing a small bias.

7 Appendix

7.1 Proposition 1

In this appendix we give a proof of Proposition 1. For probability measure ϖ and function f , $\varpi(f) := \int f(x)\varpi(dx)$. For any collection of points $(\chi_1^{(1)}, \dots, \chi_{n-1}^{(N)}) \in \bar{E}_{n-1}^N$ pg write

$$S_{n-1}^N(x) = \frac{1}{N} \sum_{i=1}^N \mathbb{I}_{\{\chi_{n-1}^{(i)}\}}(x).$$

The transition kernels are written K_1 (which is not to be confused with the K_1 from the SMC samplers algorithm) and for any $n \geq 2, N \geq 1, N$ -empirical density $S_{n-1}^N, K_{S_{n-1}^N, n}$ is the kernel of invariant distribution

$$\frac{l_{(t_{n-1}, t_n]}(\bar{y}_{n,1}; \bar{x}_n)}{p_n(\bar{y}_{n,1}|\bar{y}_{n-1})} \mathbf{p}(\bar{x}_{n,1}) S_{n-1}^N(\bar{x}_{n-1})$$

where we have dropped the $\tilde{\cdot}$ notation from the main text of the article. Recall the generic notation $\bar{x}_n \in \bar{E}_n$. We drop the dependence upon the data and denote

$$g_n(\bar{x}_n) = \frac{l_{(t_{n-1}, t_n]}(\bar{y}_{n,1}; \bar{x}_n)}{p_n(\bar{y}_{n,1}|\bar{y}_{n-1})} \mathbf{p}(\bar{x}_{n,1}). \tag{11}$$

The N -empirical measure of points generated up to time $n - 1$ is written $S_{\bar{x}, n-1}^N$. For a given $n \geq 1, f_n : \bar{E}_n \rightarrow \mathbb{R}$ we have the notation $K_{n, S_{n-1}^N}(f_n)(x) := \int_{\bar{E}_n} f_n(y) K_{n, S_{n-1}^N}(x, dy)$ and $i \geq 1, K_{n, S_{n-1}^N}^i(f_n)(x) := \int_{\bar{E}_n} K_{n, S_{n-1}^N}^{i-1}(x, dy) K_{n, S_{n-1}^N}(f_n)(y), K_{n, S_{n-1}^N}^0(x, dy) = \delta_x(dy)$ the Dirac measure. The σ -finite measure $d\bar{x}_{n,1}$ is defined on the space $\bar{E}_n \setminus \bar{E}_{n-1}$; in practice, it is the product of an appropriate version of Lebesgue and counting measures.

The following assumption is made.

Assumption (A). There exist an $\epsilon_1 \in (0, 1)$ and probability measure κ_1 on \bar{E}_1 such that for any $\bar{x}_1 \in \bar{E}_1$

$$K_1(\bar{x}_1, \cdot) \geq \epsilon_1 \kappa_1(\cdot).$$

For any $n \geq 2$, there exist an $\epsilon_n \in (0, 1)$ and probability measure κ_n on $\bar{E}_n \setminus \bar{E}_{n-1}$ such that for any $\bar{x}_n \in \bar{E}_n$ and any collection of points $(\chi_{n-1}^{(1)}, \dots, \chi_{n-1}^{(N)}) \in \bar{E}_{n-1}^N$

$$K_{S_{n-1}^N, n}(\bar{x}_n, \cdot) \geq \epsilon_n S_{n-1}^N(\cdot) \kappa_n(\cdot).$$

For any $n \geq 2$

$$\sup_{\bar{x}_{n-1} \in \bar{E}_{n-1}} \int_{\bar{E}_n \setminus \bar{E}_{n-1}} |g_n(\bar{x}_{n-1}, \bar{x}_{n,1})| d\bar{x}_{n,1} < +\infty$$

where g_n is as in (11).

It should be noted that the uniform ergodicity assumption on $K_{S_{n-1}^N, n}(\bar{x}_n, \cdot)$ is quite strong. If the kernel $K_{S_{n-1}^N, n}$ were an Metropolis-Hastings independence sampler of

proposal $S_{n-1}^N \times q_n(\cdot) \bar{x}_n = (\bar{x}_{n-1}, \bar{x}_{n,1})$, then

$$K_{S_{n-1}^N, n}(\bar{x}_n, \cdot) \geq \min \left\{ 1, \frac{g_n(\bar{v}_n)q_n(\bar{x}_{n,1})}{g_n(\bar{x}_n)q_n(\bar{v}_{n,1})} \right\} S_{n-1}^N(\cdot)q_n(\cdot)$$

satisfies the assumption if $q_n(\bar{x}_{n,1})/g_n(\bar{x}_n)$ is uniformly lower-bounded. Note also, due to the suppression of the data from the notation, it is typical that ϵ_n would depend upon \bar{y}_n .

Proof 1 The proof is inductive on n . Some details are omitted as the proof is quite similar to the control of adaptive MCMC chains, e.g. [Andrieu et al. \(2011\)](#). It should be noted the proof for this algorithm differs as the kernel possesses an invariant measure that does not change with the iteration $i \in \{1, \dots, N\}$.

Let $n = 1$ then, by (A) K_1 is a uniformly ergodic Markov kernel of invariant measure π_1 . It is simple to use the Poisson equation to prove the proposition, which is given to establish the induction. Let $\hat{f}_1(\bar{x}_1) = \sum_{i=0}^{\infty} [K_1^i(f_1)(\bar{x}_1) - \pi_1(f_1)]$ be the solution to the Poisson equation; $\hat{f}_1 - K_1(\hat{f}_1) = f_1 - \pi_1(f_1)$. Then

$$\begin{aligned} \sum_{i=1}^N [f_1(\bar{x}_1^{(i)}) - \pi_1(f_1)] &= \sum_{i=1}^N [\hat{f}_1(\bar{x}_1^{(i)}) - K_1(\hat{f}_1)(\bar{x}_1^{(i)})] \\ &= \sum_{i=1}^{N-1} [\hat{f}_1(\bar{x}_1^{(i+1)}) - K_1(\hat{f}_1)(\bar{x}_1^{(i)})] + \hat{f}_1(\bar{x}_1^{(1)}) - K_1(\hat{f}_1)(\bar{x}_1^{(N)}) \end{aligned}$$

the first quantity on the RHS is a Martingale, M_N^1 , w.r.t. the filtration \mathcal{F}_1^i (i.e. the σ -algebra generated by Markov chain). Then, using the Minkowski inequality

$$\begin{aligned} \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \frac{1}{N} \sum_{i=1}^N [f_1(\bar{x}_1^{(i)}) - \pi_1(f_1)] \right|^p \right]^{1/p} &\leq \frac{1}{N} \left\{ \mathbb{E}_{\bar{x}_1^{(1)}} \left[|M_N^1|^p \right]^{1/p} + |\hat{f}_1(\bar{x}_1^{(1)})| \right. \\ &\quad \left. + \mathbb{E}_{\bar{x}_1^{(1)}} \left[|K_1(\hat{f}_1)(\bar{x}_1^{(N)})|^p \right]^{1/p} \right\}. \end{aligned}$$

The last term can be dealt with as follows.

$$\begin{aligned} \mathbb{E}_{\bar{x}_1^{(1)}} \left[|K_1(\hat{f}_1)(\bar{x}_1^{(N)})|^p \right]^{1/p} &\leq \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \sum_{i=0}^{\infty} [K_1^i(f_1)(\bar{x}_1^{(N+1)}) - \pi_1(f_1)] \right|^p \right]^{1/p} \\ &\leq \|f_1\| \sum_{i=0}^{\infty} \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| [K_1^i - \pi_1] \left(\frac{f_1}{\|f_1\|} \right) (\bar{x}_1^{(N+1)}) \right|^p \right]^{1/p} \\ &\leq \frac{\|f_1\|}{\epsilon_1}. \end{aligned}$$

Here, we have applied the conditional Jensen inequality and the bound on the total variation distance for uniformly ergodic Markov chains: $\forall x \in \bar{E}_1$,

$\sup_{f: \bar{E}_1 \rightarrow [0,1]} |K_1^i(f)(x) - \pi_1(f)| \leq (1 - \epsilon_1)^i$. Note that this bound holds for any $\bar{x}_1 \in \bar{E}_1$. The Martingale term is bounded using the Bürkholder and Davis inequalities (i.e. the inequality below holds for any $p \geq 1$):

$$\mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| M_N^1 \right|^p \right]^{1/p} \leq B_p \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \sum_{i=1}^{N-1} [f_1(\bar{x}_1^{(i)}) - K_1(f_1)(\bar{x}_1^{(i)})]^2 \right|^{p/2} \right]^{1/p}.$$

When $p \geq 2$ the Minkowski inequality and the above manipulations yield a bound $\sqrt{N}B(p, \epsilon_1)\|f_1\|$, with $B(p, \epsilon_1)$ a constant only depending upon p and ϵ_1 . When $p \in [1, 2)$ the inequality $(a - b)^2 \leq 2(a^2 + b^2)$ for $a, b \in \mathbb{R}$ is applied then Jensen to yield a similar bound; see [Andrieu et al. \(2011\)](#) and the references therein. Thus, for $n = 1$ it follows $\mathbb{E}_{\bar{x}_1^{(1)}}[|M_N^1|^p]^{1/p} \leq \sqrt{N}B(p, \epsilon_1)\|f_1\|$; note that $B(p, \epsilon_1)$ depends only on ϵ_1 and p —this is important in the sequel. Putting these bounds together and noting that, by the above arguments, the solution to the Poisson equation is uniformly bounded in x the proof at rank $n = 1$ is completed.

Now assume the result at $n - 1$ and consider n . Note that via Fubini

$$\pi_n(f_n) = \int_{\bar{E}_n} f_n(\bar{x}_n)g_n(\bar{x}_n)\pi_{n-1}(d\bar{x}_{n-1})d\bar{x}_{n,1} = \int_{\bar{E}_{n-1}} I(f_n \times g_n)(\bar{x}_{n-1})\pi_{n-1}(d\bar{x}_{n-1})$$

where $I(f_n \times g_n) = \int_{\bar{E}_n \setminus E_{n-1}} f_n(\bar{x}_{n-1}, \bar{x}_{n,1})g_n(\bar{x}_{n-1}, \bar{x}_{n,1})d\bar{x}_{n,1}$. Then application of the Minkowski inequality yields:

$$\begin{aligned} & \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \frac{1}{N} \sum_{i=1}^N f_n(\bar{x}_n^{(i)}) - \pi_n(f_n) \right|^p \right]^{1/p} \\ & \leq \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| \frac{1}{N} \sum_{i=1}^N f_n(\bar{x}_n^{(i)}) - S_{\bar{x},n-1}^N(I(f_n \times g_n)) \right|^p \right]^{1/p} \\ & \quad + \mathbb{E}_{\bar{x}_1^{(1)}} \left[\left| [S_{\bar{x},n-1}^N - \pi_{n-1}](I(f_n \times g_n)) \right|^p \right]^{1/p}. \end{aligned} \tag{12}$$

Due to the induction hypothesis and (A), the second term on the RHS of the inequality is upper-bounded by

$$\frac{B_{p,n-1} \sup_{\bar{x}_{n-1} \in \bar{E}_{n-1}} I(|f_n \times g_n|)(\bar{x}_{n-1})}{\sqrt{N}} \leq \frac{B_{p,n}\|f_n\|}{\sqrt{N}}$$

for some $B_{p,n} < +\infty$; if the data were not suppressed, then there is an explicit dependence upon this quantity. Then considering the first term on the RHS of (12), conditioning upon the σ -algebra $\mathcal{F}_1^N \otimes \dots \otimes \mathcal{F}_{n-1}^N$ generated by the process at time n is a uniformly ergodic Markov chain of invariant distribution $S_{\bar{x},n-1}^N(d\bar{x}_{n-1})g_n(\bar{x}_n)d\bar{x}_{n,1}$.

Thus, for example:

$$\mathbb{E}_{\tilde{x}_1^{(1)}} \left[\left| K_{n, S_{\tilde{x}_{n-1}}^N}(\hat{f}_n)(\tilde{x}_n^{(N)}) \right|^p \right]^{1/p} \leq \mathbb{E}_{\tilde{x}_1^{(1)}} \left[\left(\frac{\|f_n\|}{\epsilon_n} \right)^p \right]^{1/p}$$

adopting exactly the above arguments. Noting that the bound on the conditional expectation is deterministic, i.e. does not depend upon $\mathcal{F}_1^N \otimes \dots \otimes \mathcal{F}_{n-1}^N$, the induction is easily completed. \square

7.2 Proof of Proposition 2

For the proof of Proposition 2, we require a round of notations. We write $\tilde{x}_p = (\tilde{x}_p, \tilde{x}'_p) \in \bar{E}_1^2$ and define the following quantities:

$$G_p(\tilde{x}_p) = \frac{\pi_p(\tilde{x}_p)}{\pi_{p-1}(\tilde{x}_p)} \quad 1 \leq p \leq r_1$$

with $G_0(\tilde{x}_0) = 1$. In addition, set $\eta_0(\cdot) = \mathbf{p}(\cdot)$ and

$$M_p(\tilde{x}_{p-1}, d\tilde{x}_p) = \delta_{x'_{p-1}}(dx_p) K_p(x_p, dx'_p) \quad 1 \leq p \leq r_1$$

We add an extra Markov kernel to allow us to use directly formulae in [Del Moral \(2004\)](#); $M_{r_1+1}(\tilde{x}_{r_1}, d\tilde{x}_{r_1+1}) = \delta_{\tilde{x}_{r_1+1}}(d\tilde{x}_{r_1+1})$. Then we define

$$\eta_p(d\tilde{x}_p) = \frac{\int_{(\bar{E}_1^2)^{p-1}} \eta_0(d\tilde{x}_0) \prod_{q=0}^{p-1} G_q(\tilde{x}_q) M_q(\tilde{x}_{q-1}, d\tilde{x}_q)}{\int_{(\bar{E}_1^2)^p} \eta_0(d\tilde{x}_0) \prod_{q=0}^{p-1} G_q(\tilde{x}_q) M_q(\tilde{x}_{q-1}, d\tilde{x}_q)} \quad 1 \leq p \leq r_1 + 1.$$

In addition $Q_p(\tilde{x}_{p-1}, d\tilde{x}_p) = G_{p-1}(\tilde{x}_{p-1}) M_p(\tilde{x}_{p-1}, d\tilde{x}_p)$, $1 \leq p \leq r_1 + 1$, with

$$Q_{p,n}(\tilde{x}_{p-1}, d\tilde{x}_n) = \int_{(\bar{E}_1^2)} Q_{p+1}(\tilde{x}_p, d\tilde{x}_{p+1}) \dots Q_n(\tilde{x}_{n-1}, \tilde{x}_n) \quad 1 \leq p \leq n \leq r_1 + 1$$

with the convention that $Q_{p,p}$ is the identity operator. Also define $P_{p,n}(\tilde{x}_{p-1}, d\tilde{x}_n) = Q_{p,n}(\tilde{x}_{p-1}, d\tilde{x}_n) / Q_{p,n}(1)(\tilde{x}_{p-1})$ and finally

$$\bar{Q}_{p,n}(\tilde{x}_{p-1}, d\tilde{x}_n) = \frac{Q_{p,n}(\tilde{x}_{p-1}, d\tilde{x}_n)}{\eta_p Q_{p,n}(1)}.$$

Proof of Propostion 2 We have from Proposition 9.4.2 of [Del Moral \(2004\)](#) that:

$$\sigma_{\text{TE}, r_1}^2(f) = \sum_{p=0}^{r_1+1} \eta_p(\bar{Q}_{p,r_1+1}(f - \eta_{r_1+1}(f))^2). \tag{13}$$

The objective is to re-write the summand in terms of a difference $P_{p,r_{t_1}+1}(\tilde{x}, \cdot) - P_{p,r_{t_1}+1}(\tilde{x}', \cdot)$ and use the mixing conditions to control the Dobrushin coefficient of the kernel $P_{p,r_{t_1}+1}$; see e.g. [Del Moral et al. \(2012\)](#) section 4. To that end, we can only consider the first $r_{t_1} - 1$ terms, for which the Dobrushin coefficient will satisfy:

$$\beta(P_{p,r_{t_1}+1}) := \sup_{\tilde{x}, \tilde{x}'} \|P_{p,r_{t_1}+1}(\tilde{x}, \cdot) - P_{p,r_{t_1}+1}(\tilde{x}', \cdot)\|_{TV} \leq (1 - \rho)^{\lfloor (r_{t_1} + 1 - p)/2 \rfloor} \quad r_{t_1} - p \geq 1 \tag{14}$$

for some $\rho \in (0, 1)$ that does not depend upon r_{t_1} and $\|\cdot\|_{TV}$ the total variation distance (again see [Del Moral et al. 2012](#), as the condition $(\mathcal{M})_2$ of that paper is satisfied). The reminder of the terms in the sum are easily bounded, independently of r_{t_1} , and we omit these calculations.

Using standard properties of Feynman–Kac formula, we have that each summand in (13) is equal to

$$\eta_p \left(\frac{Q_{p,r_{t_1}+1}(1)^2}{\eta_p(Q_{p,r_{t_1}+1}(1))^2} \frac{\eta_p(Q_{p,r_{t_1}+1}(1)[P_{p,r_{t_1}+1}(f)(\tilde{x}) - P_{p,r_{t_1}+1}(f)]^2)}{\eta_p(Q_{p,r_{t_1}+1}(1))^2} \right)$$

Using Jensen’s inequality, it follows that

$$\begin{aligned} & \eta_p \left(\frac{Q_{p,r_{t_1}+1}(1)^2}{\eta_p(Q_{p,r_{t_1}+1}(1))^2} \frac{\eta_p(Q_{p,r_{t_1}+1}(1)[P_{p,r_{t_1}+1}(f)(\tilde{x}) - P_{p,r_{t_1}+1}(f)]^2)}{\eta_p(Q_{p,r_{t_1}+1}(1))^2} \right) \\ & \leq \|f\|^2 \beta(P_{p,r_{t_1}+1})^2 \frac{\eta_p(Q_{p,r_{t_1}+1}(1))^2}{\eta_p(Q_{p,r_{t_1}+1}(1))^4}. \end{aligned}$$

Using the fact that (see, e.g. section 4 of [Del Moral et al. 2012](#))

$$\sup_{\tilde{x}, \tilde{x}'} \frac{Q_{p,r_{t_1}+1}(1)(\tilde{x})}{Q_{p,r_{t_1}+1}(1)(\tilde{x}')} \leq B$$

for a $B \in (0, +\infty)$ that does not depend on r_{t_1} and using the bound in (14) we can conclude. □

Acknowledgments We thank Nick Whiteley for conversations on this work. The first author acknowledges the support of an EPSRC grant. The second author was supported by an MOE grant. We thank two referees and an associate editor for their comments, which have vastly improved the article.

References

Andrieu, C., Jasra, A., Doucet, A., Del Moral, P. (2011). On non-linear Markov chain Monte Carlo. *Bernoulli*, 17, 987–1014.
 Barndorff-Nielsen, O., Shephard, N. (2001). Non-Gaussian Ornstein-Uhlenbeck models and some of their uses in financial economics (with discussion). *Journal of the Royal Statistical Society Series B*, 63, 167–241.

- Beskos, A., Crisan, D., Jasra, A. (2011). *On the stability of sequential Monte Carlo methods in high dimensions*. Technical Report, Imperial College London, London.
- Centanni, S., Minozzo, M. (2006a). A Monte Carlo approach to filtering for a class of marked doubly stochastic Poisson processes. *Journal of the American Statistical Association*, *101*, 1582–1597.
- Centanni, S., Minozzo, M. (2006b). Estimation and filtering by reversible jump MCMC for a doubly stochastic Poisson model for ultra-high-frequency financial data. *Statistical Modelling*, *6*, 97–118.
- Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, *89*, 539–552.
- Chopin, N., Jacob, P., Papaspiliopoulos, O. (2012). SMC²: A sequential Monte Carlo algorithm with particle Markov chain Monte Carlo updates. *Journal of the Royal Statistical Society Series B* (to appear).
- Daley, D. J., Vere-Jones, D. (1988). *Introduction to the theory of point processes*. New York: Springer.
- Del Moral, P. (2004). *Feynman-Kac formulae. Genealogical and interacting particle systems*. New York: Springer.
- Del Moral, P., Doucet, A., Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society Series B*, *68*, 411–32.
- Del Moral, P., Doucet, A., Jasra, A. (2007). Sequential Monte Carlo for Bayesian computation (with discussion). In: S. Bayarri, J. O. Berger, J. M. Bernardo, A. P. Dawid, D. Heckerman, A. F. M. Smith, M. West (Eds.), *Bayesian statistics* (Vol. 8, pp. 115–149). Oxford: OUP.
- Del Moral, P., Doucet, A., Jasra, A. (2012). On adaptive resampling procedures for sequential Monte Carlo methods. *Bernoulli*, *18*, 252–278.
- Doucet, A., De Freitas, J. F. G., Gordon, N. J. (2001). *Sequential Monte Carlo methods in practice*. New York: Springer.
- Doucet, A., Montesano, L., Jasra, A. (2006). Optimal filtering for partially observed point processes using trans-dimensional sequential Monte Carlo. *International Conference on Acoustics, Speech, and Signal Processing*, *5*, 597–600.
- Eberle, A., Marinelli, C. (2012). Quantitative approximations of evolving probability measures and sequential Markov chain Monte Carlo methods. *Probability Theory and Related Fields* (to appear).
- Fearnhead, P. (2004). Exact filtering for partially-observed queues. *Statistics and Computing*, *14*, 261–266.
- Glynn, P. W., Meyn, S. P. (1996). A Lyapunov bound for solutions of the Poisson equation. *Annals of Probability*, *24*, 916–931.
- Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*, 711–732.
- Jasra, A., Stephens, D. A., Holmes, C. C. (2007). On population-based simulation for static inference. *Statistics and Computing*, *17*, 263–279.
- Kantas, N., Chopin, N., Doucet, A., Singh, S. S., Maciejowski, J. M. (2011). *On particle methods for parameter estimation in general state-space models*. Technical Report, Imperial College London, London.
- Liu, J. S. (2001). *Monte Carlo strategies in scientific computing*. New York: Springer.
- Pitt, M. K., Shephard, N. (1997). Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, *94*, 590–599.
- Roberts, G. O., Papaspiliopoulos, O., Dellaportas, P. (2004). Bayesian inference for non-Gaussian Ornstein-Uhlenbeck stochastic volatility processes. *Journal of the Royal Statistical Society Series B*, *66*, 369–393.
- Rousset, M., Doucet, A. (2006). Discussion of Beskos et al. *Journal of the Royal Statistical Society Series B*, *68*, 374–375.
- Rydberg, T. H., Shephard, N. (2000). A modelling framework for the prices and times of trades made on the New York Stock exchange. In W. J. Fitzgerald, R. L. Smith, A. T. Walden, P. C. Young (Eds.), *Non-linear and non-stationary signal processing* (p. 246). Cambridge: CUP.
- Shiryayev, A. (1996). *Probability*. New York: Springer.
- Snyder, D. L. (1972). Filtering and detection for doubly stochastic Poisson processes. *IEEE Transactions on Information Theory*, *18*, 91–102.
- Snyder, D. L., Miller, M. I. (1998). *Random point processes in space and time*. New York: Springer.
- Varini, E. (2007). A Monte Carlo method for filtering a marked doubly stochastic Poisson process. *Statistical Methods and Applications*, *17*, 183–193.
- Whiteley, N. P., Johansen, A. M., Godsill, S. J. (2011). Monte Carlo filtering of piece-wise deterministic processes. *Journal of Computational and Graphical Statistics*, *20*, 119–139.