Inference for Partially Observed Point Process Models

BY JAMES S. MARTIN, AJAY JASRA & EMMA McCoy

Department of Mathematics, Imperial College London, SW7 2AZ, London, UK,

james.martin04@ic.ac.uk, ajay.jasra@ic.ac.uk , emma.mccoy@ic.ac.uk

Abstract

This paper presents a simulation-based framework for sequential inference from partially and discretely observed point process (PP's) models. We build upon sequential Monte Carlo (SMC) methods for such cases, 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 running example of intensity estimation for financial models.

Some Key Words: Point Processes, Sequential Monte Carlo, Intensity Estimation

1 Introduction

Partially observed point processes provide a rich class of models to describe both sequential and static data. For example, such models provide natural parameterizations for stochastic volatility models (Barndorff-Nielsen & Shephard, 2001) in finance, descriptions of queuing data in operations research (Fearnhead, 2004), important seismological models (Daley & Vere-Jones, 1988) and applications in nuclear physics (Snyder & Miller, 1998). From the Bayesian perspective, there have been some recent MCMC-based contributions to the fitting of such models in the static context (Green, 1995; Roberts et al. 2004). For complex dynamic models, that is, when data arrive sequentially in time, studies date back to at least Snyder (1972). However, the fitting procedure is more difficult requiring the application of SMC (e.g. Doucet et al. (2000)) and MCMC methods. The main developments in this field include the work of: Centanni & Minozzo (2006a,b); Del Moral et al (2006,2007); Doucet et al (2006); Rydberg & Shephard (2000), see also Whiteley et al. (2011) for some related ideas.

As a result of the above methods, inference in such models is now feasible and hence interest in their application has increased. For example in statistical finance; it is possible to apply realistic models for tick-by-tick data, Centanni & Minozzo (2006b), without approximation of the model. Therefore, there is motivation for further improving the methodology in the afore-mentioned references, to allow the application of even more realistic statistical models.

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 $t_0 = 0 < t_1 < \cdots < t_{\widetilde{m}} < T = t_{\widetilde{m}+1}$, on

the unobserved marked PP $(k_{t_n}, \phi_{1:k_{t_n}}, \zeta_{1:k_{t_n}})$, where $\phi_{1:k_{t_n}} = (\phi_1, \ldots, \phi_{k_{t_n}})$ are the ordered event times (constrained to $[0, t_n]$) and $\zeta_{1:k_{t_n}} = (\zeta_1, \ldots, \zeta_{k_{t_n}})$ are marks, given the observations $y_{1:r_{t_n}}$. In other words to compute, for $n \geq 1$, at time t_n the posterior densities

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

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

In addition, there are static parameters $\theta \in \Theta$ specifying the probability model and these parameters are to be estimated in a Bayesian manner, or using maximum likelihood. The inferential procedure will depend upon the computational power that is available. 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 (used interchangeably with time-step). Thus an on-line algorithm is necessarily sequential but not vice-versa. The reader should keep in mind such conventions for the duration of the article. Our methods, when the parameters are estimated in a Bayesian fashion, are sequential.

1.1 Previous Methodology

One of the first works applying advanced computational methods to PP models, was Rydberg & Shephard (2000). The authors focus upon a Cox model; the unobserved PP parameterizes the intensity of an observed PP. Rydberg & Shephard (2000) used the auxiliary particle filter (Pitt & Shephard, 1997) to simulate from the posterior density of the integrated intensity at a given time point of the process. This was superseded by Centanni & Minozzo (2006a,b), which allows one to infer the conditional density of the unknown intensity at any given time of the process, up to the current observation. Centanni & Minozzo (2006a,b) perform an MCMC-type filtering (smoothing as defined above) algorithm, estimating static parameters using stochastic EM; note this is specifically designed for marked doubly stochastic PPs. The methodology of Centanni & Minozzo (2006a,b) cannot easily be adapted to the case in which static parameters are to be estimated using Bayesian methods. The approach of Centanni & Minozzo (2006a) also requires some minor clarification to establish its theoretical validity; this is verified, under strong assumptions, in Proposition 1 of the present article.

SMC samplers (Del Moral et al. 2006), which are the focus of this paper, provide an approach which can be applied to all the problems stated above. SMC methods involve simulating a set of $N \geq 1$ weighted samples, termed particles, in order to approximate a sequence of target distributions, which to some extent 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, in some sense and under mild assumptions, as $N \to \infty$ (Del Moral, 2004). A key feature of the approach, is that its flexibility stems from the fact that the user must select:

1. the sequence of distributions

2. the mechanism by which particles are propagated .

Poor choices will result in inefficient algorithms. In this article, a variety of approaches to the above issues are considered. It should be noted that the methods developed here can be used for any partially observed PP for which the smoothing density is known up-to a normalizing constant.

1.2 The Difficulty of Extending the Space

In problems where the underlying state-spaces and target distributions are of a complex form, constructing efficient SMC algorithms is not straight-forward. If the sequence of target distributions and the mechanism by which particles are propagated is not chosen carefully, there can be a substantial discrepancy between the proposal and target. As a result, the variance of the importance weights will be very large and estimation will be 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 by how much to enlarge the space at each iteration of the SMC algorithm and how to choose a mechanism by which to propose particle components in the new region of the space. We refer to this issue as *the difficulty of extending the space*. This difficulty is, in some ways, similar to that faced when designing reversible jump MCMC algorithms (Green, 1995); there it is difficult to generate good dimension changing proposals.

Two approaches to the sampling problem 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 by sequentially introducing likelihood terms corresponding to observations made on consecutive intervals. This idea, whilst not amenable to on-line implementation, circumvents the problem of extending the space, at an extra computational cost. In addition, inference for the original density of interest can be achieved by simple importance sampling (IS). The parallel with reversible jump is clear; such saturation methods have been used in that context also; see Carlin & Chib (1995). The disadvantage of this approach is that it cannot be used if the data are not constrained to a known observation interval. Due to this draw-back, another solution is investigated.

In the second approach, entitled *data-point tempering*, the sequence of target distributions are defined by sequentially introducing likelihood terms corresponding to each observed event. This is achieved as follows: given that the PP has been sampled, appropriately, on $[0, t_n]$ the target is extended onto $[0, t_{n+1}]$ by sampling the missing part of the PP from the prior. Then, subsequently, introducing likelihood terms into the target that correspond to the data (as in Chopin (2002)), one at a time. Once all of the data have

been introduced, the target density will correspond to the one of interest i.e. (1). The motivation for this approach is to soften the difficulty of extending the space: new likelihood terms are introduced slowly in order to avoid degeneracy of the importance weights. It should be noted that neither of the methods are online, but some simple fixes are proposed and investigated.

1.3 Structure and Objectives of the Paper

To summarize, this paper presents the methods by which to perform the following inferential tasks for partially observed PP models:

- Bayesian sequential filtering/smoothing.
- On-line filtering of the PP, with known parameters.

Indeed, one of the main objectives of this article is to both introduce, investigate and evaluate the relative merits of our proposed methodology.

Section 2 introduces a PP model from finance which serves as a running example. In Section 3 the ideas of Centanni & Minnozo (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 Section 4 we introduce our SMC methods. In Section 5 our methods are illustrated upon the running example. In Section 6 we conclude the paper, detailing extensions to our work.

2 Motivating Example

2.1 Notations and Conventions

Some notations are introduced. Throughout we are interested in a sequence of probability measures $\{\varpi_n\}_{1\leq n\leq m^*}$ on a sequence of spaces $\{(G_n, \mathcal{G}_n)\}_{1\leq n\leq m^*}$, with dominating $\sigma-$finite measures on the spaces, $n \in \{1, \ldots, m^*\}$. The class of probability measures on G_n are written $\mathcal{P}(G_n)$. The bounded and measurable functions on G_n, $f_n : G_n \to \mathbb{R}$, are written $\mathcal{B}_b(G_n)$ and the sup-norm is $\|f_n\| = \sup_{x\in G_n} |f_n(x)|$. The abusive notation ϖ_n will refer to either the probability measure $\varpi_n(dx)$ or the density $\varpi_n(x)$. In addition, $\varpi_n(f_n) := \int_{G_n} f_n(x)\varpi_n(dx)$.

2.2 Finance Problem

Our motivating example is from statistical finance. An important type of financial data is ultra high frequency data which consists 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 & 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.

2.2.1 The Likelihood

A marked PP, of $r_T \geq 1$ points, is observed in a given 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 $\Omega_{r,T} = \{\omega_{1:r_T} : 0 < \omega_1 < \cdots < \omega_{r_T} < T\}$, $\Xi \subseteq \mathbb{R}$. 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 & Minozzo (2006b), is

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

with $\tilde{p}$ a generic density, $\varphi(\cdot;\mu,\sigma)$ the density of a t-distribution on 1 degree of freedom, location μ, scale σ and λ_u is the intensity, the latter is described in the subsequent Section.

We remark that, 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.

2.2.2 Dynamics of Partially Observed Process

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 $t \in [0, T]$,

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

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

2.2.3 Posterior and Inference Problem

The objective is to perform inference at times $0 < t_1 < \cdots < t_{\tilde{m}} < T = t_{\tilde{m}+1}$, that is, 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

$$\pi_n(k_{t_n}, \zeta_{1:k_{t_n}}, \phi_{1:k_{t_n}}, \mu, \sigma | \omega_{1:r_{t_n}}, \xi_{1:r_{t_n}}) \propto$$

$$\prod_{i=1}^{r_{t_n}} \{\varphi(\xi_i; \mu, \sigma)\lambda_{\omega_i}\} \exp\left\{ - \int_0^{t_n} \lambda_u du \right\} \times \prod_{i=1}^{k_{t_n}} \{\tilde{p}(\zeta_i)\} \tilde{p}(\phi_{1:k_{t_n}}) \tilde{p}(k_{t_n}) \times \tilde{p}(\mu, \sigma)$$

$$= l_{[0,t_n]}(\omega_{1:r_{t_n}}, \xi_{1:r_{t_n}}; \lambda_0, k_{t_n}, \zeta_{1:k_{t_n}}, \phi_{1:k_{t_n}}, \mu, \sigma) \times \mathsf{p}(k_{t_n}, \zeta_{1:k_{t_n}}, \phi_{1:k_{t_n}}) \times \tilde{p}(\mu, \sigma) \tag{4}$$

with $\mu \sim \mathcal{N}(\alpha_\mu, \beta_\mu)$, $\sigma \sim \mathcal{G}a(\alpha_\sigma, \beta_\sigma)$, $\phi_{1:k_t} | k_t \sim \mathcal{U}_{\Phi_{k,t_n}}$, $k_t \sim \mathcal{P}o(\gamma t)$ 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, $\mathcal{G}a(\alpha, \beta)$ the Gamma distribution of mean α/β and $\mathcal{P}o$ is the Poisson distribution. $\mathsf{p}(k_{t_n}, \zeta_{1:k_{t_n}}, \phi_{1:k_{t_n}})$ is the notation for the prior on the marked point-process and $\tilde{p}(\mu, \sigma)$ is the notation for the prior on (μ, σ).

In the sequel we will need to deal explicitly will components of the hidden and observed PP's which correspond to events in various intervals. For this purpose, with a slight abuse and for integer ℓ we define the following notation:

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

Under this convention, $\bar{x}_n$ (respectively $\bar{y}_n$) is the the restriction of the hidden (observed) PP to that with events in $[0, t_n]$. Similarly $\bar{x}_{n,\ell}$ (respectively $\bar{y}_{n,\ell}$) is the the restriction of the hidden (observed) PP to that with events in $[t_{n-\ell}, t_n]$. Also note that, given $(\bar{x}_{n-1}, \bar{x}_{n,1})$, $\bar{x}_n$ is known.

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 (Section 3.2). 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. The methods can be used to draw inference from the marginal posterior of the process, on $(t_{n-1}, t_n]$ using an algorithm of fixed computational complexity; for example by constraining any simulation to a fixed-size state-space; see Section 4.4.

3 Previous Approaches

3.1 Reversible Jump Method of Centanni & Minozzo

One of the approaches for performing filtering for partially observed PP's is from Centanni & Minozzo (2006a). In this Section the parameters (λ_0, μ, σ) are assumed known. Let

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

This is the support of the target densities for this method.

The following decomposition is adopted

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

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

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

where $S_{x,n-1}^N(\bar{x}_{n-1}) := \frac{1}{N} \sum_{i=1}^N \mathbb{I}_{\{\bar{X}_{n-1}^{(i)}\}}(\bar{x}_{n-1})$ with $\bar{X}_{n-1}^{(1)}, \ldots, \bar{X}_{n-1}^{(N)}$ 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 $S_{x,0}^N$. At time $n = 1$ the algorithm starts from an arbitrary point $\bar{x}_1^{(1)} \in \bar{E}_1$ and subsequent steps are initialized by a draw from the empirical $S_{x,n-1}^N$ and the prior p (this can be modified); $N - 1$ additional samples are simulated.

The above algorithm can be justified, theoretically, by using the Poisson equation (e.g. Glynn & Meyn (1996)) and induction arguments. Below the assumption (A) is made; see the appendix for details as well as the proof. Also, the expectation below is w.r.t. the 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 $f_n \in \mathcal{B}_b(\bar{E}_n)$*

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

This result helps to establish the theoretical validity of the method in Centanni & 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 Section 3.3.

3.2 SMC Methods

3.2.1 Overview

As noted in Section 1, the approach of Centanni & Minozzo has some drawbacks and therefore SMC samplers are considered. The aim is to simulate from and compute expectations w.r.t. 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 hence may not be known prior to simulation. In our case the probability measures are defined upon nested state-spaces: with the exception of concerns about the existence of the importance weights (i.e. as well-defined Radon-Nikodym derivatives), this case can be similarly handled.

The SMC sampler approach consists of introducing a sequence of auxiliary probability measures $\{\widetilde{\pi}_n\}_{0\leq n\leq m^*}$ on state-spaces of increasing dimension ($E_{[0,n]} := E_0 \times \cdots \times E_n, \mathcal{E}_{[0,n]} := \mathcal{E}_0 \otimes \cdots \otimes \mathcal{E}_n$), such that they admit the $\{\pi_n\}_{0\leq n\leq m^*}$ as marginals.

The following sequence of auxiliary densities is used:

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

where $\{L_n\}_{0\leq n\leq m^*-1}$ are a sequence of Markov kernels that act backward in time and are thus termed *backward* Markov kernels. In our application π_0 is the prior, on E_1 (as defined below). It is clear that (7) admit 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 Figure 1. Further details of the motivation for this construction can be found in Del Moral et al. (2006).

The ESS in Figure 1 refers to the effective sample size (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 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 $L_{n-1}(x_n, x_{n-1}) = \pi_n(x_{n-1})K_n(x_{n-1}, x_n)/\pi_n(x_n)$. See Del Moral et al. (2006) for guidelines to select the forward and backward kernels. It is possible to 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; again see Del Moral et al. (2006) for details.

3.2.2 Nested Spaces

As described in Section 1.2, in complex problems it is often difficult to design efficient SMC algorithms. In the example in Section 2.2, the state-spaces of the subsequent densities are not common. The objective is

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

- 2. For $i = 1, \ldots, N$ 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.

Figure 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.

to sample from a sequence of densities on the space, at time n,

$$E_n = \left(\bigcup_{k \in \mathbb{N}_0} \{k\} \times \Phi_{k,t_n} \times (\mathbb{R}^+)^k \right) \times \mathbb{R} \times (\mathbb{R}^+)^2$$

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. It is now demonstrated that such a procedure does not always work well.

3.2.3 Finance Problem Revisited

Consider the example of Section 2.2. The following SMC algorithm is used: At time n of the algorithm, two moves are used to sample the particles; birth and extend.

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

$$W_n = \frac{\pi_n(\lambda_0, k_{t_{n-1}} + 1, \zeta_{1:k_{t_{n-1}}}, \zeta_{k_{t_n}+1}, \phi_{1:k_{t_{n-1}}}, \phi_{k_{t_n}}, \mu, \sigma)(t_n - \phi_{k_{t_{n-1}}})}{\pi_{n-1}(\lambda_0, k_{t_{n-1}}, \zeta_{1:k_{t_{n-1}}}, \phi_{1:k_{t_{n-1}}}, \mu, \sigma) p(\zeta_{k_{t_n}+1})}.$$

- **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 as in Del Moral et al. (2007).

Note, as remarked in Whiteley et al. (2011), we need to be able to sample any number of births; this was a mistake in Del Moral et al. (2006). With an extremely small probability, a proposal from the prior is included; in practice it is never simulated.

In addition to the above steps an MCMC sweep is included, in the same way as Del Moral et al. (2006). That is, after the decision of whether or not to resample the particles is taken, an MCMC kernel of invariant measure π_n is applied. The kernel is much the same as in Green (1995) and we refer the reader to that paper for details.

3.2.4 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 1^{st} 1900 and the non-integer part representing the fraction of 24 hours that has passed during that day; thus, one minute corresponds to an interval of length $1/1440$. It is intuitive, therefore, to 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 $\lambda = 6000$, $\lambda = 2000$ and $\lambda = 4000$. 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 $r_T = 3206$, $[0, T] = [0, 0.9]$ with $t_n = n * 0.003$. The intensity from which they were generated had constant levels at 6000 in the interval [0.05,0.18]; at 4000 in the interval [0.51,0.68]; and at 2000 in the intervals [0.28,0.42] and [0.78,0.90].

The sampler was implemented with all permutations $\{(M, N)\}$ for $N \in \{100, 1000\}$ and $M \in \{1, 5, 20\}$, resampling whenever the effective sample size fell below $N/2$. When performing statistical inference, the intensity (3) used parameters $\gamma = 0.001$, $\nu = 150$ and $s = 20$.

Upon implementation, it was found that the system consistently collapsed to a single particle representation of the distribution of interest within an extremely short time period. That is, the sampler resamples at almost every time step, which leads to an extremely poor representation of the target density. Figure 2 shows the ESS at each time step for the current implementation. As can be seen, the algorithm behaves extremely poorly for this model.

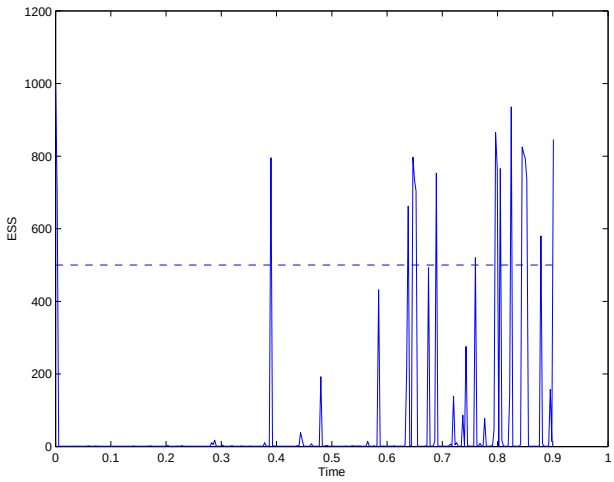


Figure 2: Effective Sample Size plots for the SMC sampler described in Figure 1, implemented with $N = 1000$ particles and with $M = 5$ MCMC sweeps at each iteration. The dashed line indicates the resampling threshold at $N/2 = 500$ particles; the sampler resamples at 94.4% of the time steps.

3.3 Discussion

We have reviewed two existing techniques for the Bayesian analysis of partially observed PP's. 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 & Minozzo (2006a) is the fact that the state-space need not be extended. On page 1586 of Centanni & 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 is likely to increase over time, even under the exceptionally strong assumption (A) in the appendix. This is opposed to SMC methods which are provably stable, under similar assumptions (and that the entire state is updated), as $n \to \infty$ (Del Moral, 2004); we discuss this below. In other words, whilst the approach of Centanni & 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 &

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, 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 requires a computational budget that grows with the time parameter n.

3.3.1 Instability of SMC Sampler

As mentioned above, SMC methods are provably stable under some conditions as the time parameter grows. However, some remarks related to the method in Figure 1 can help to shed some light on the poor behaviour in Section 3.2.4. Consider the scenario when one is interested in statistical inference on $[0, t_1]$ say, but we allow the number of data points to grow to infinity. Suppose for simplicity, one can write the posterior on this region as

$$\pi_1(\bar{x}_1) = \exp\{\sum_{i=1}^{r_1} g(y_i; \bar{x}_1)\} \tag{9}$$

for fixed $r_1, \lambda_0, \mu, \sigma$. Suppose that $\underline{g} \leq e^g \leq \overline{g}$.

We consider the algorithm started with a draw from the prior (the first target) and in one step adds the data (i.e. performs an MCMC step) and resamples. This attempts to replicate the scenario when one extends the state-space. Write $S_1^N(dx) = \frac{1}{N}\sum_{i=1}^N \delta_{\bar{x}_1^i}$ the empirical measure which approximates (9). Then we have the following simple result. The proof follows directly from Theorem 4.5 of Del Moral et al. (2011); the proof is hence omitted.

Proposition 2. *For SMC sampler described above with final target* (9) *and assuming the MCMC kernel satisfies* $(\mathcal{M})_1$ *in Del Moral et al. (2011), then we have for any* $N \geq 1$, $p \geq 1$, $f \in \mathcal{B}_b(\bar{E}_1)$ *and* $r_1 \geq 1$, $\bar{y_1}$ *that there exist a* $B_p < \infty$ *depending only on* p *and* f *such that*

$$\mathbb{E}[|S_1^N(f) - \pi_1(f)|^p]^{1/p} \leq \frac{B_p}{\sqrt{N}}\left[\frac{\overline{g}}{\underline{g}}\right]^{r_1}.$$

The result shows that, whilst in some scenarios, SMC methods can have a stable error, the particular SMC method associated to that in Figure 1 will have Monte Carlo error that is likely to grow with the number of data. Clearly we cannot guarantee that the error grows, as we only have an upper-bound on the error. However, given Proposition 2 and our empirical experience, it seems that we require a new methodology, especially for complex problems.

4 Proposed Methods

In the following Section tow approaches are presented to deal with the problems in Section 3.2.4. 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.

4.1 Possible Solutions to the problems of Extending the State-Space

An important remark associated to the simulations in Section 3.2.4, 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.

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 & Doucet, 2006). In the context of this problem we found that the simulation from the full conditional density of $\phi_{k_{t_n}}$ 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 effective sample size (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.

4.2 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 beginning the simulations. This is realistic in a variety of applications. For example, in Section 2.2, often we may only be interested in performing inference for a day of trading, and thus can set $[0, T]$.

In details, it is proposed to sample, in the case of the example in Section 2.2, 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 target densities are as before, except the prior on the $\phi_{1:k}$, allows them to lie outside t_n. We then use, for K_n, an MCMC kernel of invariant measure π_n. Inference w.r.t. the original $\{\pi_n\}_{1 \leq n \leq m^*}$ can be performed via IS.

4.3 Data-Point Tempering

A simple solution to the 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, then there are $m^* = n + \widetilde{m} + 1$ time-steps of the algorithm. These steps, when the latter use MCMC kernels, lead to an incremental weight that is simply the likelihood ratio.

The potential advantage of this step 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 are tempered instead (e.g. Jasra et al. (2007)).

As a theoretical justification of this idea, we return to the discussion of Section 3.3.1 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. The algorithm resamples at every time-step. As in Section 3.3.1, write S_1^N as the empirical measure that approximates target (9). As with Proposition 2 the proof follows directly from Theorem 4.5 of Del Moral et al. (2011) and is omitted.

Proposition 3. *For SMC sampler described above, with final target* (9) *and assuming the MCMC kernels satisfy* $(\mathcal{M})_1$ *with same constant, in Del Moral et al. (2011), then we have for any* $N \geq 1$, $p \geq 1$, $f \in \mathcal{B}_b(\bar{E}_1)$ *and* $r_1 \geq 1$, $\bar{y_1}$ *that there exist a* $B_p < \infty$ *depending only on* p *and* f *such that*

$$\mathbb{E}[|S_1^N(f) - \pi_1(f)|^p]^{1/p} \leq \frac{B_p}{\sqrt{N}}.$$

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 target density.

4.4 Online Implementation

A key characteristic that has not yet been addressed is the fact that each 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 $\left[\phi_{k_{t_n}-19}, t_n\right)$. Due to the well-known path degeneracy problem in SMC (see Andrieu et al. (2008)), the estimates and 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 Section 2.2 and the settings as in Section 3.2.4.

5.1 Simulated Data

The saturated and tempered samplers, as well as their RCC alternatives, were implemented using the simulated data set (in Section 3.2.4), 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 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 3 displays the evolution of the ESS over a particular run of the algorithm. Figure 4

	M = 1		M = 5		M = 20	
	N=100	N=1000	N=100	N=1000	N=100	N=1000
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%

Table 1: Table showing the resampling rates of each of the three SMC samplers and their reduced computational complexity alternatives, for the six algorithm parameterisations that were tested. The ESS plots for the saturated and tempered samplers with $N = 1000$, $M = 5$ are given in Figure 3 for comparison with the corresponding ESS plot for the benchmark sampler given in Figure 2

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 and 2, Figure 4 and comparing Figure 3 to Figure 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 the unaltered sampler. As can be seen, for the 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 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.

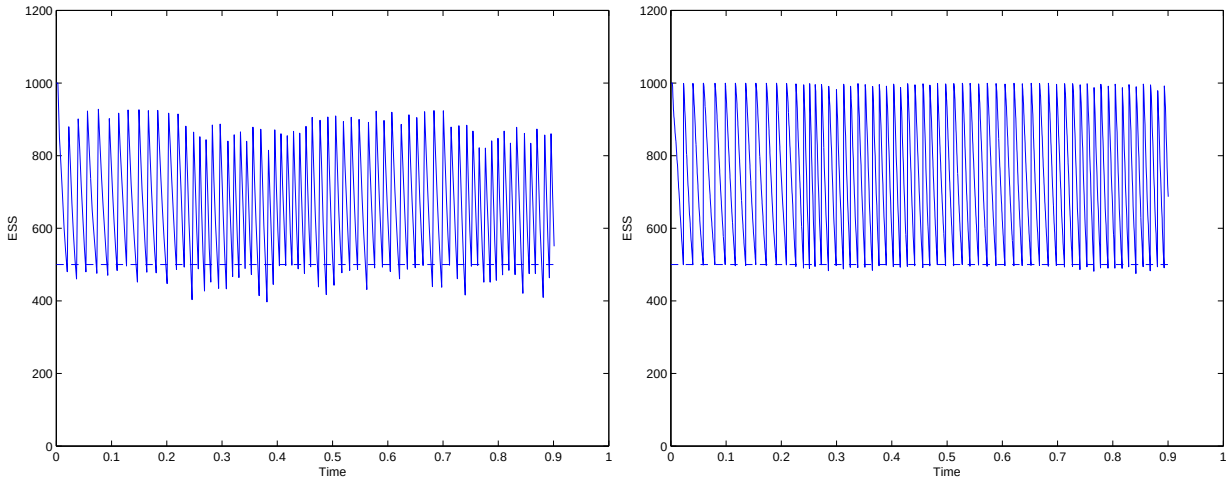
We consider also the effect that changes in M and N have on the accuracy of estimates provided by the

	M = 1		M = 5		M = 20	
	N=100	N=1000	N=100	N=1000	N=100	N=1000
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

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 parameterisations that were tested.

	M = 1		M = 5		M = 20	
	N=100	N=1000	N=100	N=1000	N=100	N=1000
Benchmark	612.9	9689.1	2849.7	45690.4	13352.1	144621.3
Benchmark - RCC	449.0	7910.9	1132.7	10657.6	3106.2	31208.5
Saturated	1125.3	10667.8	3234.3	39061.1	15381.9	141817.3
Saturated - RCC	637.5	6215.2	1200.7	11412.6	4391.9	47662.8
Tempered	1160.2	10633.4	3138.4	38679.6	14086.7	130899.1
Tempered - RCC	666.0	6424.4	1156.3	11209.1	3231.3	34795.3

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 parameterisations that were tested.



(a) Saturated

(b) Tempered

Figure 3: Effective Sample Size plots for the SMC samplers with state space saturation (left) and data point tempering (right), run with $N = 1000$ 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 ....% for the saturated sampler and ....% for the tempered sampler.

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 the filtered and smoothed estimates, that results from an increase in the number of particles used. Whilst for the filtered 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,

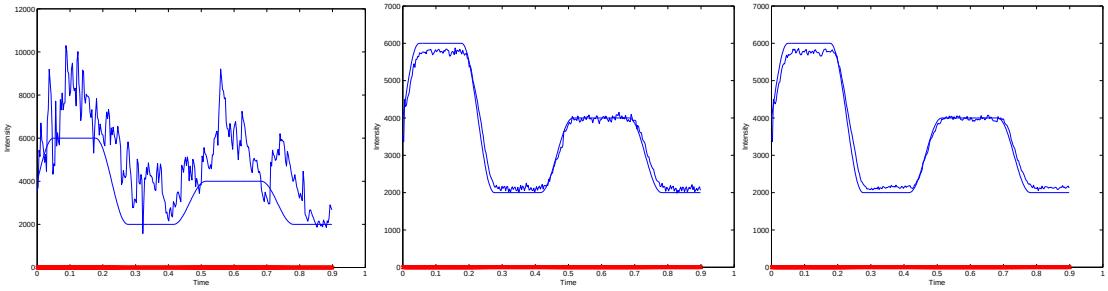
	M = 1		M = 5		M = 20	
	N=100	N=1000	N=100	N=1000	N=100	N=1000
Benchmark	688.561	1116.639	620.432	1942.992	1330.232	1501.263
Benchmark - RCC	676.932	2026.956	880.824	2247.313	1472.126	1264.533
Saturated	242.834	192.580	228.390	193.778	237.315	198.223
Saturated - RCC	229.449	189.279	224.692	193.379	225.592	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

Table 4: Table showing the root mean square error of the intensity. 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 parameterisations that were tested.

	M = 1		M = 5		M = 20	
	N=100	N=1000	N=100	N=1000	N=100	N=1000
Benchmark	768.702	670.656	495.019	627.909	489.243	571.107
Benchmark - RCC	698.640	1034.890	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

Table 5: Table showing the smoothed root mean square error of the intensity. This is given the entire data set and for each of the three samplers and their reduced computational complexity alternatives, for the six algorithm parameterisations that were tested.

that is, 3/10 of 24 hours; the first trade is just after 9am and the last around 16:15.) with $t_n = n * 0.001$. Genuine financial data is likely to correspond to a more volatile latent intensity process than that which was used to generate the synthetic data set, and so the parameterisation 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 $N = 1000$ 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.



(a) Benchmark (b) Saturated (c) Tempered

Figure 4: 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 $N = 1000$ particles and with $M = 5$ MCMC sweeps at each iteration. The model parameters were $\gamma = 0.001$, $\nu = 150$ and $s = 20$.

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 $\left(\hat{\lambda}_{n,j}\right)^{-1}$ 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 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 $i = 1, \ldots, 100$ using each sampler, both the saturated and 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 hours (each observation interval corresponds to 0.0264 days = 1.584 minutes).

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.

Finally, an existing intensity estimation procedure for inhomogeneous point processes was sought within Matlab and R, against which to compare the predictive errors of the above samplers; the only available

	Data t_n RMSPEs	Smoothed RMSPEs	Processing Times (s)	Resampling Rates
Saturated	2.18876	2.13479	4064.5	39.5%
Saturated - RCC	2.19112	-	2193.1	39.9%
Tempered	2.34671	2.11468	4605.5	19.8%
Tempered - RCC	2.42776	-	2237.3	19.9%

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 $N = 1000$, $M = 5$. 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.

procedure was found within the spatstat package in R. This procedure, however, simply estimates the intensity over an interval as the number of events within the observed interval, scaled to suit a suitable interval of interest (such as the trading day). When applied to the data, the resulting RMSPE for one-step-ahead prediction was found to be 11.2407. As would be expected, the model described here significantly outperforms this naïve estimation procedure.

6 Summary

In this paper we have considered SMC simulation for partially observed point processes. 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. It was observed that the methods can be successful, in terms of weight degeneracy.

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 have conducted, in unpublished experiments (joint with N. Whiteley, University of Bristol) 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. This method is expected to be highly efficient, but is not 'exact', in the sense that Monte Carlo estimates will not converge to the true solutions of interest. 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. This yields an on-line algorithm, which does not suffer from the above problem. 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 have proposed employing a sensible approximation of this density, at the cost of introducing a small bias. This will be investigated in future work.

Acknowledgement

We thank Nick Whiteley for conversations on this work.

Appendix

In this appendix we give a proof of Proposition 1. For any collection of points $(\chi_1^{(1)}, \ldots, \chi_{n-1}^{(N)}) \in \bar{E}_{n-1}^N$ 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

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

Recall the generic notation $\bar{x}_n \in \bar{E}_n$. We drop the dependence upon the data and denote

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

The $N-$empirical measure of points generated up-to time $n-1$ is written $S_{\bar{x},n-1}^N$. For a given $n \geq 1$, $f_n : \bar{E}_n \to \mathbb{R}$ 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 $\sigma-$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 $\widetilde{x}_n \in \bar{E}_n$ and any collection of points $(\chi_{n-1}^{(1)}, \ldots, \chi_{n-1}^{(N)}) \in \bar{E}_{n-1}^N$

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

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 $K_{S^N_{n-1},n}(\bar{x}_n, \cdot)$ is quite strong. If the kernel $K_{S^N_{n-1},n}$ were an Metropolis-Hastings independence sampler of proposal $S^N_{n-1} \times q_n(\cdot)$ $\bar{x}_n = (\bar{x}_{n-1}, \bar{x}_{n,1})$, then

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

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. 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 $i \in \{1, \ldots, N\}$.

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 R.H.S. is a Martingale, M_N^1, w.r.t. the filtration $\mathcal{F}_1^i$ (i.e. the $\sigma-$algebra generated by Markov chain). Then, using the Minkowski inequality

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

The last term can be dealt with as follows.

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

here we have applied the conditional Jensen inequality and the bound on the total variation distance for uniformly ergodic Markov chains: $\forall x \in \bar{E}_1$, $\sup_{f:\bar{E}_1 \to [0,1]} |K_1^i(f)(x) - \pi_1(f)| \leq (1 - \epsilon_1)^i$. 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 $p \geq 1$):

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

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 $\mathbb{E}_{\bar{x}_1^{(1)}}\left[|M_N^1|^p\right]^{1/p} \leq \sqrt{N}B(p,\epsilon_1)\|f_1\|$; 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:

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

Due to the induction hypothesis and (A), the second term on the R.H.S 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 R.H.S of (12), conditioning upon the $\sigma-$algebra $\mathcal{F}_1^N \otimes \cdots \otimes \mathcal{F}_{n-1}^N$ generated by the process at time n is a uniformly ergodic Markov chain of invariant distribution $S^N_{\bar{x},n-1}(d\bar{x}_{n-1}) g_n(\bar{x}_n) d\bar{x}_{n,1}$. Thus, for example:

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

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

REFERENCES

Andrieu, C., Doucet, A. & Tadic, V. (2008). On-line parameter estimation in general state-space models using pseudo-likelihood. Technical Report, University of Bristol.

Andrieu, C., Jasra, A., Doucet, A. & Del Moral, P. (2011). On non-linear Markov chain Monte Carlo. *Bernoulli* (to appear).

Barndorff-Nielsen, O. & Shephard, N. (2001). Non-Gaussian Ornstein-Uhlenbeck models and some of their uses in financial economics (with discussion). *J. R. Statist. Soc. Ser. B*, **63**, 167–241.

Beskos, A., Crisan, D. & Jasra, A. (2011). On the stability of a class of sequential Monte Carlo methods in high dimensions. Technical Report, Imperial College London.

Carlin, B. & Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *J. R. Statist. Soc. Ser. B*, **57**, 473–484.

Centanni, S. & Minozzo, M. (2006a). A Monte Carlo approach to filtering for a class of marked doubly stochastic Poisson processes. *J. Amer. Statist. Assoc.*, **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. *J. Statist. Mod.*, **6**, 97–118.

Chopin, N. (2002). A sequential particle filter for static models. *Biometrika*, **89**, 539–552.

Daley, D. J. & Vere-Jones, D. (1988). *Introduction to the Theory of Point Processes*, Springer-Verlag: New York.

Del Moral, P., (2004). *Feynman-Kac formulae. Genealogical and interacting particle systems*, Springer-Verlag: New York.

Del Moral, P., Doucet, A. & Jasra, A. (2006). Sequential Monte Carlo samplers, *J. R. Statist. Soc. B* **68**, 411-32.

Del Moral, P., Doucet, A. & Jasra, A. (2007). Sequential Monte Carlo for Bayesian computation (with discussion). *Bayesian Statistics 8*, Ed. Bayarri, S., Berger, J. O., Bernardo, J. M., Dawid, A. P., Heckerman, D. Smith, A. F. M. and West, M. 115-149, OUP: Oxford.

Del Moral, P., Doucet, A. & Jasra, A. (2011). On adaptive resampling procedures for sequential Monte Carlo methods, *Bernoulli* (to appear).

Doucet, A., Montesano, L. & Jasra, A. (2006). Optimal filtering for partially observed point processes using trans-dimensional sequential Monte Carlo, *ICASSP*.

Doucet, A., De Freitas, J. F. G. & Gordon, N. J. (2001). *Sequential Monte Carlo Methods in Practice*. Springer: New York.

Fearnhead, P. (2004). Exact filtering for partially-observed queues. *Statist. Comp.*, **14**, 261–266.

Glynn, P. W. & Meyn S. P. (1996). A Lyapunov bound for solutions of the Poisson equation. *Ann. Prob.*, **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. *Statist. Comp.*, **17**, 263–279.

Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer: New York.

Pitt, M. K. & Shepherd, N. (1997). Filtering via simulation: Auxiliary particle filters. *J. Amer. Statist. Assoc.*, **94**, 590–599.

Roberts, G. O., Papaspiliopoulos, O. & Dellaportas, P. (2004). Bayesian inference for non-Gaussian Ornstein-Uhlenbeck stochastic volatility processes. *J. R. Statist. Soc. B*, **66**, 369-393.

Rousset, M., & Doucet, A. (2006). Discussion of Beskos et al. *J. R. Statist. Soc. 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. *Non-linear and Non-stationary Signal Processing*, Ed. Fitzgerald, W. J., Smith, R. L., Walden, A. T. & Young, P. C., 217–246, CUP: Cambridge.

Shiryaev, A. (1996). *Probability*, Springer: New York.

Snyder, D. L. (1972). Filtering and detection for doubly stochastic Poisson processes. *IEEE Trans. Infor. Th.*, **18**, 91–102.

Snyder, D. L. & Miller, M. I. (1998). *Random Point Processes in Space and Time*, Springer-Verlag: New York.

Varini, E. (2007). A Monte Carlo method for filtering a marked doubly stochastic Poisson Process. *Stat. Meth. & Appl.*, **17**, 183–193.

Whiteley, N. P., Johansen, A. M. & Godsill, S. J. (2011). Monte Carlo filtering of piece-wise deterministic processes. *J. Comp. Graph. Stat* (to appear).