



# Multilevel particle filters for Lévy-driven stochastic differential equations

Ajay Jasra<sup>1</sup> · Kody J. H. Law<sup>2,3</sup>  · Prince Peprah Osei<sup>1</sup>

Received: 3 January 2018 / Accepted: 13 October 2018 / Published online: 20 October 2018  
© The Author(s) 2018, corrected publication 2018

## Abstract

We develop algorithms for computing expectations with respect to the laws of models associated to stochastic differential equations driven by pure Lévy processes. We consider filtering such processes as well as pricing of path dependent options. We propose a multilevel particle filter to address the computational issues involved in solving these continuum problems. We show via numerical simulations and theoretical results that under suitable assumptions regarding the discretization of the underlying driving Lévy process, the cost to obtain MSE  $\mathcal{O}(\epsilon^2)$  scales like  $\mathcal{O}(\epsilon^{-2})$  for our method, as compared with the standard particle filter  $\mathcal{O}(\epsilon^{-3})$ .

**Keywords** Lévy-driven SDE · Lévy processes · Particle filters · Multilevel particle filters · Barrier options

## 1 Introduction

Lévy processes have become very useful recently in several scientific disciplines. A non-exhaustive list includes physics, in the study of turbulence and quantum field theory; economics, for continuous time-series models; insurance mathematics, for computation of insurance and risk, and mathematical finance, for pricing path dependent options. Earlier application of Lévy processes in modeling financial instruments dates back in Madan and Seneta (1990) where a variance gamma process is used to model market returns.

A typical computational problem in mathematical finance is the computation of the quantity  $\mathbb{E}[f(Y_t)]$ , where  $Y_t$  is the time  $t$  solution of a stochastic differential equation driven by a Lévy process and  $f \in \mathcal{B}_b(\mathbb{R}^d)$ , a bounded Borel measurable function on  $\mathbb{R}^d$ . For instance  $f$  can be a pay-off function. Typically one uses the Black–Scholes model, in which the underlying price process is lognormal. However, often the asset price exhibits big jumps over the time

horizon. The inconsistency of the assumptions of the Black–Scholes model for market data has led to the development of more realistic models for these data in the literature. General Lévy processes offer a promising alternative to describe the observed reality of financial market data, as compared to models that are based on standard Brownian motions.

In the application of standard and multilevel particle filter methods to SDEs driven by general Lévy processes, in addition to pricing path dependent options, we will consider filtering of partially-observed Lévy process with discrete-time observations. In the latter context, we will assume that the partially-observed data are regularly spaced observations  $z_1, \dots, z_n$ , where  $z_k \in \mathbb{R}^d$  is a realization of  $Z_k$  and the density of  $Z_k$  given  $Y_{k\tau} = y_{k\tau}$  is known, where  $\tau$  is the time scale. Real S&P 500 stock price data will be used to illustrate our proposed methods as well as the standard particle filter. We will show how both of these problems can be formulated as general Feynman–Kac type problems (Moral 2004), with time-dependent potential functions modifying the Lévy path measure.

The multilevel Monte Carlo (MLMC) methodology was introduced in Heinrich (2001) and first applied to the simulation of SDE driven by Brownian motion in Giles (2008). Recently, Dereich and Heidenreich (2011) provided a detailed analysis of the application of MLMC to a Lévy-driven SDE. This first work was extended in Dereich (2011) to a method with a Gaussian correction term which can substantially improve the rate for pure jump

✉ Kody J. H. Law  
kody.law@manchester.ac.uk

<sup>1</sup> Department of Statistics and Applied Probability, National University of Singapore, Singapore 117546, Singapore

<sup>2</sup> School of Mathematics, University of Manchester, Manchester, UK

<sup>3</sup> Computer Science and Mathematics Division, Oak Ridge National Laboratory Oak Ridge, Oak Ridge, TN 37831, USA

processes (Asmussen and Rosiński 2001). The authors in Ferreiro-Castilla et al. (2014) use the MLMC method for general Lévy processes based on Wiener–Hopf decomposition. We extend the methodology described in Dereich and Heidenreich (2011) to a particle filtering framework. This is challenging due to the following reasons. First, one must choose a suitable weighting function to prevent the weights in the particle filter being zero (or infinite). Next, one must control the jump part of the underlying Lévy process such that the path of the filter does not blow up as the time parameter increases. In pricing path dependent options, for example knock out barrier options, we adopt the same strategy described in Jasra and Moral (2011) and Jasra and Doucet (2009) for the computation of the expectation of the functionals of the SDE driven by general Lévy processes.

The rest of the paper is organised as follows. In Sect. 2, we briefly review the construction of general Lévy processes, the numerical approximation of Lévy-driven SDEs, the MLMC method, and finally the construction of a coupled kernel for Lévy-driven SDEs which will allow MLMC to be used. Section 3 introduces both the standard and multilevel particle filter methods and their application to Lévy-driven SDEs. Section 4 features numerical examples of pricing barrier options and filtering of partially observed Lévy processes. The computational savings of the multilevel particle filter over the standard particle filter is illustrated in this section.

## 2 Approximating SDE driven by Lévy processes

In this section, we briefly describe the construction and approximation of a general  $d'$ -dimensional Lévy process  $\{X_t\}_{t \in [0, K]}$ , and the solution  $Y := \{Y_t\}_{t \in [0, K]}$  of a  $d$ -dimensional SDE driven by  $X$ . Consider a stochastic differential equation given by

$$dY_t = a(Y_{t-})dX_t, \quad y_0 \in \mathbb{R}^d, \quad (1)$$

where  $a : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d'}$ , and the initial value is  $y_0$  (assumed known). In particular, in the present work we are interested in computing the expectation of bounded and measurable functions  $f : \mathbb{R}^d \rightarrow \mathbb{R}$ , that is  $\mathbb{E}[f(Y_t)]$ .

### 2.1 Lévy processes

For a general detailed description of the Lévy processes and analysis of SDEs driven by Lévy processes, we shall refer the reader to the monographs of Bertoin (1996), Sato (1999), Applebaum (2004) and Protter (2004). Lévy processes are stochastic processes with stationary and independent increments, which begin almost surely from the origin and are stochastically continuous. Two important fundamental tools

available to study the richness of the class of Lévy processes are the Lévy–Khintchine formula and the Lévy–Itô decomposition. They respectively characterize the distributional properties and the structure of sample paths of the Lévy process. Important examples of Lévy processes include Poisson processes, compound Poisson processes and Brownian motions.

There is a strong interplay between Lévy processes and infinitely divisible distributions such that, for any  $t > 0$  the distribution of  $X_t$  is infinitely divisible. Conversely, if  $F$  is an infinitely divisible distribution then there exists a Lévy process  $X$  such that the distribution of  $X_1$  is given by  $F$ . This conclusion is the result of Lévy–Khintchine formula for Lévy processes we describe below. Let  $X$  be a Lévy process with a triplet  $(\nu, \Sigma, b)$ ,  $b \in \mathbb{R}^{d'}$ ,  $0 \leq \Sigma = \Sigma^T \in \mathbb{R}^{d' \times d'}$ , where  $\nu$  is a measure satisfying  $\nu(\{0\}) = 0$  and  $\int_{\mathbb{R}^{d'}} (1 \wedge |x|^2) \nu(dx) < \infty$ , such that

$$\mathbb{E}[e^{i\langle u, X_t \rangle}] = \int_{\mathbb{R}^{d'}} e^{i\langle u, x \rangle} \pi(dx) = e^{t\psi(u)}$$

with  $\pi$  the probability law of  $X_t$ , where

$$\begin{aligned} \psi(u) = & i\langle u, b \rangle - \frac{\langle u, \Sigma u \rangle}{2} + \int_{\mathbb{R}^{d'} \setminus \{0\}} \\ & \times \left( e^{i\langle u, x \rangle} - 1 - i\langle u, x \rangle \mathbb{1}_{\{|x| < 1\}} \right) \nu(dx), \quad u \in \mathbb{R}^{d'}. \end{aligned} \quad (2)$$

The measure  $\nu$  is called the Lévy measure of  $X$ . The triplet of Lévy characteristics  $(\nu, \Sigma, b)$  is simply called Lévy triplet. Note that in general, the Lévy measure  $\nu$  can be finite or infinite. If  $\nu(\mathbb{R}^{d'}) < \infty$ , then almost all paths of the Lévy process have a finite number of jumps on every compact interval and it can be represented as a compensated compound Poisson process. On the other hand, if  $\nu(\mathbb{R}^{d'}) = \infty$ , then the process has an infinite number of jumps on every compact interval almost surely. Even in this case the third term in the integrand ensures that the integral is finite, and hence so is the characteristic exponent.

### 2.2 Simulation of Lévy processes

The law of increments of many Lévy processes is not known explicitly. This makes it more difficult to simulate a path of a general Lévy process than for instance standard Brownian motion. For a few Lévy processes where the distribution of the process is known explicitly, Cont and Tankov (2004) and Schoutens (2003) provided methods for exact simulation of such processes, which are applicable in financial modelling. For our purposes, the simulation of an approximated path of a general Lévy process will be based on the Lévy–Itô decomposition and we briefly describe the construction below. An

alternative construction is based on Wiener–Hopf decomposition. This is used in Ferreiro-Castilla et al. (2014).

The Lévy–Itô decomposition reveals much about the structure of the paths of a Lévy process. We can split the Lévy exponent, or the characteristic exponent of  $X_t$  in (2), into three parts

$$\psi = \psi^1 + \psi^2 + \psi^3.$$

where

$$\begin{aligned} \psi^1(u) &= i\langle u, b \rangle, \quad \psi^2(u) = -\frac{\langle u, \Sigma u \rangle}{2}, \\ \psi^3(u) &= \int_{\mathbb{R}^d \setminus \{0\}} \left( e^{i\langle u, x \rangle} - 1 - i\langle u, x \rangle \mathbb{1}_{\{|x| < 1\}} \right) \nu(dx), \quad u \in \mathbb{R}^d \end{aligned}$$

The first term corresponds to a deterministic drift process with parameter  $b$ , the second term to a Wiener process with covariance  $\Sigma$ , and the last part corresponds to a Lévy process which is a square integrable martingale. This term may either be a compensated compound Poisson process or the limit of such processes, and it is the hardest to handle when it arises from such a limit.

Thus, any Lévy process can be decomposed into three independent Lévy processes thanks to the Lévy–Itô decomposition theorem. In particular, let  $\{L_t\}_{t \in [0, K]}$  denote a process such that the characteristic exponent of  $L_t$  is  $t\psi^3(u)$ , and let  $\{W_t\}_{t \in [0, K]}$  denote a Wiener process independent of the process  $\{L_t\}_{t \in [0, K]}$ . A Lévy process  $\{X_t\}_{t \in [0, K]}$  can be constructed as follows

$$X_t = \sqrt{\Sigma}W_t + L_t + bt, \tag{3}$$

where  $\sqrt{\Sigma}$  denotes the symmetric square-root of  $\Sigma$ . The Lévy–Itô decomposition guarantees that every square integrable Lévy process has a representation as (3). We will assume that one cannot sample from the law of  $X_t$ , hence of  $Y_t$ , and rather we must numerically approximate the process with finite resolution. Such numerical methods have been studied extensively, for example in Jacod et al. (2005) and Rubenthaler (2003).

Let  $|\cdot|$  denote the standard Euclidean  $l_2$  norm, for vectors, and induced operator norm for matrices. It will be assumed that the Lévy process  $X$  (2), and the Lévy-driven process  $Y$  in (1), satisfy the following conditions.

**Assumption 2.1** There exists a  $C > 0$  such that

- (i)  $|a(y) - a(y')| \leq C|y - y'|$ , and  $|a(y)| \leq C$  for all  $y \in \mathbb{R}^d$ ;
- (ii)  $0 < \int_{\mathbb{R}^d} |x|^2 \nu(dx) \leq C^2$ ;

Item (i) provides continuity of the forward map, while (ii) controls the variance of the jumps. These assumptions are

the same as in the paper (Dereich and Heidenreich 2011), with the exception of the second part of (i), which was not required there.

### 2.3 Numerical approximation of a Lévy process and Lévy-driven SDE

Recall (1) and (3). Consider the evolution of discretized Lévy process and the Lévy-driven SDE over the time interval  $[0, K]$ .

In order to describe the Euler discretization of the two processes for a given accuracy parameter  $h_l$ , we need some definitions. The meaning of the subscript  $l$  will become clear in the next section. Let  $\delta_l > 0$  denote a jump threshold parameter in the sense that jumps which are smaller than  $\delta_l$  will be ignored. Let  $B_{\delta_l} = \{x \in \mathbb{R}^d : |x| < \delta_l\}$ . Define  $\lambda_l = \nu(B_{\delta_l}^c) < \infty$ , that is the Lévy measure outside of the ball of radius  $\delta_l$ . We assume that the Lévy component of the process is nontrivial so that  $\nu(B_1) = \infty$ , which assures that  $\lambda_l > 0$  for  $\delta_l$  sufficiently small. Note that  $\lambda_l$  will increase as  $\delta_l$  decreases and, assuming  $\delta_\infty = 0$ , then  $\lambda_\infty = \infty$ . Therefore, the following procedure is logical. First  $h_l$  will be chosen and then the parameter  $\delta_l$  will be chosen such that the step-size of the time-stepping method is  $h_l = 1/\lambda_l$ . The jump time increments larger than  $\delta_l$  are exponentially distributed with parameter  $\lambda_l$  so that the number of jumps larger than  $\delta_l$  before time  $t$  is a Poisson process  $N^l(t)$  with intensity  $\lambda_l$ . The jump times will be denoted by  $\tilde{T}_j^l$ . The jump heights  $\Delta L_{\tilde{T}_j^l}^l$  are distributed according to

$$\mu^l(dx) := \frac{1}{\lambda_l} \mathbb{1}_{B_{\delta_l}^c}(x) \nu(dx).$$

Define

$$F_0^l = \int_{B_{\delta_l}^c} x \nu(dx). \tag{4}$$

The expected total length of all jumps before time  $t$  is  $F_0^l t$ , and the compensated compound Poisson process  $L^{\delta_l}$  defined by

$$L_t^{\delta_l} = \sum_{j=1}^{N^l(t)} \Delta L_{\tilde{T}_j^l}^l - F_0^l t$$

is an  $L^2$  martingale which converges in  $L^2$  to the Lévy process  $L$  as  $\delta_l \rightarrow 0$  (Applebaum 2004; Dereich and Heidenreich 2011).

The Euler discretization of the Lévy process and the Lévy driven SDE is given by Algorithm 1. Appropriate refinement of the original jump times  $\{\tilde{T}_j^l\}$  to new jump times  $\{T_j^l\}$  is necessary to control the discretization error arising from the

Brownian motion component, the original drift process, and the drift component of the compound Poisson process. Note that the  $\Delta L_{T_j^l}^l$  is non-zero only when  $T_j^l$  corresponds with  $\tilde{T}_m^l$  for some  $m$ , as a consequence of the construction presented above.

**Algorithm 1 : Discretization of Lévy process**

Initialization: Let  $\tilde{T}_0^l = 0$  and  $j = 1$ ;

(A) Generate jump times:  
 $\tilde{T}_j^l = \min\{1, \tilde{T}_{j-1}^l + \xi_j^l\}$ ,  $\xi_j^l \sim Exp(\lambda_l)$  ;  
 If  $\tilde{T}_j^l = 1$ ,  $\tilde{k}_l = j$ ; Go to (B);  
 Otherwise  $j = j + 1$ ; Go to start of (A).

(B) Generate jump heights:  
 For  $j \in \{1, \dots, \tilde{k}_l - 1\}$ ,  $z_j^l \sim \mu^l$ ;  
 $\Delta L_{\tilde{T}_j^l}^l = z_j^l$  and  $\Delta L_{\tilde{T}_{\tilde{k}_l}^l}^l = 0$ ;  
 Set  $j = 1$ ,  $T_0^l = 0$ .

(C) Refinement of original jump times:  
 $T_j^l = \min \left\{ T_{j-1}^l + h_l, \min \left\{ \tilde{T}_k^l > T_{j-1}^l; k \in \{1, \dots, \tilde{k}_l\} \right\} \right\}$  ;  
 If  $T_j^l = \tilde{T}_k^l$  for some  $k \in \{1, \dots, \tilde{k}_l\}$ , then  $\Delta L_{T_j^l}^l = \Delta L_{\tilde{T}_j^l}^l$ ;  
 otherwise  $\Delta L_{T_j^l}^l = 0$  ;  
 If  $T_j^l = 1$ ,  $k_l = j$ ; Go to (D);  
 Otherwise  $j = j + 1$ ; Go to start of (C).

(D) Recursion of the process:

For  $m \in \{0, \dots, k_l - 1\}$ ,  $X_0^l = x_0$ ;

$$X_{m+1}^l = X_m^l + \sqrt{\Sigma} (W_{T_{m+1}^l} - W_{T_m^l}) + \Delta L_{T_{m+1}^l}^l + (b - F_0^l)(T_{m+1}^l - T_m^l). \tag{5}$$

The numerical approximation of the Lévy process described in Algorithm 1 gives rise to an approximation of the Lévy-driven SDE as follows. Given  $Y_{T_0^l}^l$ , for  $m = 0, \dots, k_l - 1$

$$Y_{T_{m+1}^l}^l = Y_{T_m^l}^l + a(Y_{T_m^l}^l)(\Delta X)_{T_{m+1}^l}^l, \tag{6}$$

where  $(\Delta X)_{T_{m+1}^l}^l = X_{T_{m+1}^l}^l - X_{T_m^l}^l$  is given by (5). In particular the recursion in (6) gives rise to a transition kernel, denoted by  $Q^l(u, dy)$ , between observation times  $t \in \{0, 1, \dots, K\}$ . This kernel is the conditional distribution of  $Y_{T_{k_l}^l}^l$  given initial condition  $Y_{T_0^l}^l = u$ . Observe that the initial condition for  $X$  is irrelevant for simulation of  $Y$ , since only the increments  $(\Delta X)_{T_{m+1}^l}^l$  are required, which are simulated independently by adding a realization of  $N((b - F_0^l)(T_{m+1}^l - T_m^l), (T_{m+1}^l - T_m^l)\Sigma)$  to  $\Delta L_{T_{m+1}^l}^l$ .

**Remark 2.1** The numerical approximation of the Lévy process and hence Lévy-driven SDE (1) in Algorithm 1 is the

single-level version of a more general coupled discretization (Dereich and Heidenreich 2011) which will be described shortly in Sect. 2.5. This procedure will be used to obtain samples for the plain particle filter algorithm.

**2.4 Multilevel Monte Carlo method**

Suppose one aims to approximate the expectation of functions of the solution of the Lévy-driven SDE in (1) at time 1, that is  $\mathbb{E}[f(Y_1)]$ , where  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  is a bounded and measurable function. Typically, one is interested in the expectation w.r.t. the law of the exact solution of SDE (1), but this is not always possible in practice. Suppose that the law associated with (1) with no discretization is  $\pi_1$ . Since we cannot sample from  $\pi_1$ , we use a biased version  $\pi_1^L$  associated with a given level of discretization of SDE (1) at time 1. Given  $L \geq 1$ , define  $\pi_1^L(f) := \mathbb{E}[f(Y_1^L)]$ , the expectation with respect to the law associated with the Euler discretization (5) at level  $L$ . The standard Monte Carlo (MC) approximation at time 1 consists in obtaining i.i.d. samples  $(Y_1^{L,(i)})_{i=1}^{N_L}$  from the law  $\pi_1^L$  and approximating  $\pi_1^L(f)$  by its empirical average

$$\pi_1^{L,N_L}(f) := \frac{1}{N_L} \sum_{i=1}^{N_L} f(Y_1^{L,(i)}).$$

The mean square error of the estimator is

$$e(\pi_1^{L,N_L}(f))^2 := \mathbb{E} \left[ \left( \pi_1^{L,N}(f) - \pi_1(f) \right)^2 \right].$$

Since the MC estimator  $\pi_1^{L,N_L}(f)$  is an unbiased estimator for  $\pi_1^L(f)$ , this can further be decomposed into

$$e(\pi_1^{L,N_L}(f))^2 = \underbrace{N_L^{-1} \mathbb{V}[f(Y_1^L)]}_{\text{variance}} + \underbrace{(\pi_1^L(f) - \pi_1(f))^2}_{\text{bias}}. \tag{7}$$

The first term in the right hand side of the decomposition is the variance of MC simulation and the second term is the bias arising from discretization. If we want (7) to be  $\mathcal{O}(\epsilon^2)$ , then it is clearly necessary to choose  $N_L \propto \epsilon^{-2}$ , and then the total cost is  $N_L \times \text{Cost}(Y_1^{L,(i)}) \propto \epsilon^{-2-1}$ , where it is assumed that  $\text{Cost}(Y_1^{L,(i)}) \propto \epsilon^{-1}$  is the cost to ensure the bias is  $\mathcal{O}(\epsilon)$ .

Now, in the multilevel Monte Carlo (MLMC) settings, one can observe that the expectation of the finest approximation  $\pi_1^L(f)$  can be written as a telescopic sum starting from a coarser approximation  $\pi_1^0(f)$ , and the intermediate ones:

$$\pi_1^L(f) := \pi_1^0(f) + \sum_{l=1}^L (\pi_1^l(f) - \pi_1^{l-1}(f)). \tag{8}$$

Now it is our hope that the variance of the increments decays with  $l$ , which is reasonable in the present scenario where they are finite resolution approximations of a limiting process. The idea of the MLMC method is to approximate the multilevel (ML) identity (8) by independently computing each of the expectations in the telescopic sum by a standard MC method. This is possible by obtaining i.i.d. pairs of samples  $(Y_1^{l,(i)}, Y_1^{l-1,(i)})_{i=1}^{N_l}$  for each  $l$ , from a suitably coupled joint measure  $\bar{\pi}_1^l$  with the appropriate marginals  $\pi_1^l$  and  $\pi_1^{l-1}$ , for example generated from a coupled simulation the Euler discretization of SDE (1) at successive refinements. The construction of such a coupled kernel is detailed in Sect. 2.5. Suppose it is possible to obtain such coupled samples at time 1. Then for  $l = 0, \dots, L$ , one has independent MC estimates. Let

$$\begin{aligned} \pi_1^{N_{0:L}}(f) &:= \frac{1}{N_0} \sum_{i=1}^{N_0} f(Y_1^{1,(i)}) \\ &+ \sum_{l=1}^L \frac{1}{N_l} \sum_{i=1}^{N_l} (f(Y_1^{l,(i)}) - f(Y_1^{l-1,(i)})), \end{aligned} \tag{9}$$

where  $N_{0:L} := \{N_l\}_{l=0}^L$ . Analogously to the single level Monte Carlo method, the mean square error for the multilevel estimator (9) can be expanded to obtain

$$\begin{aligned} e(\pi_1^{N_{0:L}}(f))^2 &:= \underbrace{\sum_{l=0}^L N_l^{-1} \mathbb{V}[f(Y_1^l) - f(Y_1^{l-1})]}_{\text{variance}} + \underbrace{(\pi_1^L(f) - \pi_1(f))^2}_{\text{bias}}, \end{aligned} \tag{10}$$

with the convention that  $f(Y_1^{-1}) \equiv 0$ . It is observed that the bias term remains the same; that is we have not introduced any additional bias. However, by an optimal choice of  $N_{0:L}$ , one can possibly reduce the computational cost for any pre-selected tolerance of the variance of the estimator, or conversely reduce the variance of the estimator for a given computational effort.

In particular, for a given user specified error tolerance  $\epsilon$  measured in the root mean square error, the highest level  $L$  and the replication numbers  $N_{0:L}$  are derived as follows. We make the following assumptions about the bias, variance and computational cost based on the observation that there is an exponential decay of bias and variance as  $L$  increases.

Suppose that there exist some constants  $\alpha, \beta$  and an accuracy parameter  $h_l$  associated with the discretization of SDE (1) at level  $l$  such that

$$(B_l) \quad |\mathbb{E}[f(Y_1^l) - f(Y_1^{l-1})]| = \mathcal{O}(h_l^\alpha),$$

$$(V_l) \quad \mathbb{E}[|f(Y_1^l) - f(Y_1^{l-1})|^2] = \mathcal{O}(h_l^\beta),$$

$$(C_l) \quad \text{cost}(Y_1^l, Y_1^{l-1}) \propto h_l^{-1},$$

where  $\alpha, \beta$  are related to the particular choice of the discretization method and cost is the computational effort to obtain one sample  $(Y^l, Y^{l-1})$ . For example, Euler–Maruyama discretization method for the solution of SDEs driven by Brownian motion, under Assumption 2.1 on the coefficient of the SDE, and for Lipschitz  $f$ , gives orders  $\alpha = \beta = 1$  (see Kloeden and Platen 1992, Theorem 10.2.2). The accuracy parameter  $h_l$  typically takes the form  $h_l = S_0^{-l}$  for some integer  $S_0 \in \mathbb{N}$ . Such estimates can be obtained for Lévy driven SDE and this point will be revisited in detail below. For the time being we take this as an assumption.

The key observation from the mean-square error of the multilevel estimator (9)–(10) is that the bias is given by the finest level, while the variance is decomposed into a sum of variances of the  $l$ th increments. Thus the total variance is of the form  $\mathcal{V} = \sum_{l=0}^L V_l N_l^{-1}$  and by condition (V<sub>l</sub>) above, the variance of the  $l$ th increment is of the form  $V_l N_l^{-1}$ . The total computational cost takes the form  $\mathcal{C} = \sum_{l=0}^L C_l N_l$ . In order to minimize the effort to obtain a given mean square error (MSE), one must balance the terms in (10). Based on the condition (B<sub>l</sub>) above, a bias error proportional to  $\epsilon$  will require the highest level

$$L \propto \frac{-\log(\epsilon)}{\log(S_0)\alpha}. \tag{11}$$

In order to obtain optimal allocation of resources  $N_{0:L}$ , one needs to solve a constrained optimization problem: minimize the total cost  $\mathcal{C} = \sum_{l=0}^L C_l N_l$  for a given fixed total variance  $\mathcal{V} = \sum_{l=0}^L V_l N_l^{-1}$  or vice versa. Based on the conditions (V<sub>l</sub>) and (C<sub>l</sub>) above, one obtains via the Lagrange multiplier method the optimal allocation  $N_l \propto V_l^{1/2} C_l^{-1/2} \propto h_l^{(\beta+1)/2}$ .

Now targeting an error of size  $\mathcal{O}(\epsilon)$ , one sets  $N_l \propto \epsilon^{-2} h_l^{(\beta+1)/2} K(\epsilon)$ , where  $K(\epsilon)$  is chosen to control the total error for increasing  $L$ . Thus, for the multilevel estimator we obtained:

$$\text{variance} : \mathcal{V} = \sum_{l=0}^L V_l N_l^{-1} = \epsilon^2 K(\epsilon)^{-1} \sum_{l=0}^L h_l^{(\beta-1)/2}$$

$$\text{cost} : \mathcal{C} = \sum_{l=0}^L C_l N_l = \epsilon^{-2} K(\epsilon)^2.$$

One then sets  $K(\epsilon) = \sum_{l=0}^L h_l^{(\beta-1)/2}$  in order to have variance of  $\mathcal{O}(\epsilon^2)$ . We can identify three distinct cases

- (i) If  $\beta = 1$ , which corresponds to the Euler–Maruyama scheme, then  $K(\epsilon) = L$ . One can clearly see from the expression in (11) that  $L = \mathcal{O}(|\log(\epsilon)|)$ . Then the

total cost is  $\mathcal{O}(\epsilon^{-2} \log(\epsilon)^2)$  compared with single level  $\mathcal{O}(\epsilon^{-3})$ .

- (ii) If  $\beta > 1$ , which correspond to the Milstein scheme, then  $K(\epsilon) \equiv 1$ , and hence the optimal computational cost is  $\mathcal{O}(\epsilon^{-2})$ .
- (iii) If  $\beta < 1$ , which is the worst case scenario, then it is sufficient to choose  $K(\epsilon) = K_L(\epsilon) = h_L^{(\beta-1)/2}$ . In this scenario, one can easily deduce that the total cost is  $\mathcal{O}(\epsilon^{-(1/\alpha+\kappa)})$ , where  $\kappa = 2 - \beta/\alpha$ , using the fact that  $h_L \propto \epsilon^{1/\alpha}$ .

One of the defining features of the multilevel method is that the realizations  $(Y_1^l, Y_1^{l-1})$  for a given increment must be sufficiently coupled in order to obtain decaying variances  $(V_l)$ . It is clear how to accomplish this in the context of stochastic differential equations driven by Brownian motion introduced in Giles (2008) [see also Jasra et al. (2017)], where coarse increments are obtained by summing the fine increments, but it is non-trivial how to proceed in the context of SDEs purely driven by general Lévy processes. A technique based on Poisson thinning has been suggested by Giles and Xia (2012) for pure-jump diffusion and by Ferreira-Castilla et al. (2014) for general Lévy processes. In the next section, we explain an alternative construction of a coupled kernel based on the Lévy–Ito decomposition, in the same spirit as in Dereich and Heidenreich (2011).

### 2.5 Coupled sampling for Levy-driven SDEs

The ML methodology described in Sect. 2.4 works by obtaining samples from some coupled-kernel associated with discretization of (1). We now describe how one can construct such a kernel associated with the discretization of the Lévy-driven SDE. Let  $(y, y') \in \mathbb{R}^{2d}$ . For  $l \geq 1$ , let  $\check{Q}^{l,l-1}((y, y'), \cdot)$  be a coupling of the kernels  $Q^l(y, \cdot)$  and  $Q^{l-1}(y', \cdot)$ . For  $\varphi \in \mathcal{B}_b(\mathbb{R}^{2d})$  we use the notation for  $(y, y') \in \mathbb{R}^{2d}$ :

$$\begin{aligned} &\check{Q}^{l,l-1}(\varphi)(y, y') \\ &:= \int_{\mathbb{R}^{2d}} \varphi(y^l, y^{l-1}) \check{Q}^{l,l-1}((y, y'), d(y^l, y^{l-1})). \end{aligned}$$

Coupling means that for  $\varphi \in \mathcal{B}_b(\mathbb{R}^d)$

$$\begin{aligned} \check{Q}^{l,l-1}(\varphi \otimes 1)(y, y') &= Q^l(\varphi)(y), \\ \check{Q}^{l,l-1}(1 \otimes \varphi)(y, y') &= Q^{l-1}(\varphi)(y') \end{aligned}$$

where  $\otimes$  denotes the tensor product of functions, e.g.  $\varphi \otimes 1$  denotes  $\varphi(y^l)$  in the integrand associated to  $\check{Q}^{l,l-1}((y, y'), d(y^l, y^{l-1}))$ .

The coupled kernel  $\check{Q}^{l,l-1}$  can be constructed using the following strategy. Using the same definitions as in Sect. 2.3,

let  $\delta_l$  and  $\delta_{l-1}$  be user specified jump-thresholds for the fine and coarse approximation, respectively. Define

$$F_0^l = \int_{B_{\delta_l}^c} x \nu(dx) \quad \text{and} \quad F_0^{l-1} = \int_{B_{\delta_{l-1}}^c} x \nu(dx). \tag{12}$$

The objective is to generate a coupled pair  $(Y_1^{l,l}, Y_1^{l,l-1})$  given  $(y_0^l, y_0^{l-1})$ ,  $h_l, h_{l-1}$  with  $h_l < h_{l-1}$ . The parameter  $\delta_\ell(h_\ell)$  will be chosen such that  $h_\ell^{-1} = \nu(B_{\delta_\ell}^c)$ , and these determine the value of  $F_0^\ell$  in (12), for  $\ell \in \{l, l-1\}$ . We now describe the construction of the coupled kernel  $\check{Q}^{l,l-1}$  and thus obtain the coupled pair in Algorithm 2, which is the same as the one presented in Dereich and Heidenreich (2011).

---

#### Algorithm 2 : Coupled kernel $\check{Q}^{l,l-1}$ for Lévy-driven SDE

---

- (1) Generate fine process: Use parts (A) to (C) of Algorithm 1 to generate fine process yielding  $(\Delta L_{T_1^l}^{l,l}, \dots, \Delta L_{T_{k_l^l}^l}^{l,l})$  and  $(T_1^{l,l}, \dots, T_{k_l^l}^{l,l})$
- (2) Generate coarse jump times and heights: for  $j_l \in \{1, \dots, k_l^l\}$ , If  $\Delta L_{T_{j_l}^l}^{l,l} \geq \delta_{l-1}$ , then  $\Delta L_{\tilde{T}_{j_l-1}^{l,l-1}} = \Delta L_{T_{j_l}^l}^{l,l}$  and  $\tilde{T}_{j_l-1}^{l,l-1} = T_{j_l}^{l,l}$ ;  $j_{l-1} = j_l - 1 + 1$ ;
- (3) Refine jump times: Set  $j_{l-1} = j_l = 1$  and  $T_0^{l,l-1} = \bar{T}_0^{l,l} = 0$ ,
  - (i)  $T_{j_{l-1}}^{l,l-1} = \min \left\{ T_{j_{l-1}-1}^{l,l-1} + h_{l-1}, \min \left\{ \tilde{T}_k^{l,l-1} \geq T_{j_{l-1}-1}^{l,l-1}; k \in \{1, \dots, \tilde{k}_{l-1}^l\} \right\} \right\}$ .  
If  $T_{j_{l-1}}^{l,l-1} = 1$ , set  $k_{l-1}^l = j_{l-1}$ ; else  $j_{l-1} = j_{l-1} + 1$  and Go to (i).
  - (ii)  $\bar{T}_{j_l}^{l,l} = \min \left\{ T \geq \bar{T}_{j_{l-1}}^{l,l}; T \in \{T_k^{l,l-1}\}_{k=1}^{k_{l-1}^l} \cup \{T_k^{l,l}\}_{k=1}^{k_l^l} \right\}$ .  
If  $\bar{T}_{j_l}^{l,l} = 1$ , set  $k_l^l = j_l$ , and redefine  $T_i^{l,l} := \bar{T}_i^{l,l}$  for  $i = 1, \dots, k_l^l$ ;  
Else  $j_l = j_l + 1$  and Go to (ii).
- (4) Recursion of the process: sample  $W_{T_1^{l,l}}, \dots, W_{T_{k_l^l}^{l,l}}$  (noting  $\{T_k^{l,l-1}\}_{k=1}^{k_{l-1}^l} \subset \{T_k^{l,l}\}_{k=1}^{k_l^l}$ );  
Let  $m_l \in \{0, \dots, k_l^l - 1\}$ ,  $m_{l-1} \in \{0, \dots, k_{l-1}^l - 1\}$ ,  $Y_0^l = Y_0^l$ , and  $Y_0^{l-1} = Y_0^{l-1}$ ;

$$Y_{T_{m_l+1}^{l,l}}^{l,l} = Y_{T_{m_l}^{l,l}}^{l,l} + a\left(Y_{T_{m_l}^{l,l}}^{l,l}\right) \left( \sqrt{\Sigma} \Delta W_{T_{m_l+1}^{l,l}} + \Delta L_{T_{m_l+1}^{l,l}}^{l,l} + (b - F_0^l) \Delta T_{m_l+1}^{l,l} \right), \tag{13}$$

$$\begin{aligned} Y_{T_{m_{l-1}+1}^{l,l-1}}^{l,l-1} &= Y_{T_{m_{l-1}}^{l,l-1}}^{l,l-1} + a\left(Y_{T_{m_{l-1}}^{l,l-1}}^{l,l-1}\right) \\ &\times \left( \sqrt{\Sigma} \Delta W_{T_{m_{l-1}+1}^{l,l-1}} + \Delta L_{T_{m_{l-1}+1}^{l,l-1}}^{l,l-1} + (b - F_0^{l-1}) \Delta T_{m_{l-1}+1}^{l,l-1} \right), \end{aligned} \tag{14}$$

where  $\Delta W_{T_{m_\ell+1}^{l,\ell}} = W_{T_{m_\ell+1}^{l,\ell}} - W_{T_{m_\ell}^{l,\ell}}$  and  $\Delta T_{m_\ell+1}^{l,\ell} = T_{m_\ell+1}^{l,\ell} - T_{m_\ell}^{l,\ell}$ , for  $\ell \in \{l, l-1\}$ .

---

### 3 Multilevel particle filter for Lévy-driven SDEs

In this section, the multilevel particle filter will be discussed for sampling from certain types of measures which have a density with respect to a Lévy process. We will begin by briefly reviewing the general framework and standard particle filter, and then we will extend these ideas into the multilevel particle filtering framework.

#### 3.1 Filtering and normalizing constant estimation for Lévy-driven SDEs

Recall the Lévy-driven SDE (1). We will use the following notation here:  $y_{1:n} = [y_1, y_2, \dots, y_n]$ . It will be assumed that the general probability measure of interest is of the form

$$\hat{\eta}_n^\infty(dy_{1:n}) \propto \left[ \prod_{i=1}^n G_i(y_i) Q^\infty(y_{i-1}, dy_i) \right], \tag{15}$$

for  $n \geq 1$  and for some given  $y_0$ . Here  $Q^\infty(y_{i-1}, dy)$  is the transition kernel of the process (1), i.e. the probability measure of solution  $Y_1$  at time point 1 given initial condition  $Y_0 = y_{i-1}$ . It is assumed that observations  $Z_i$  are regularly observed at times 1, 2, . . . . The conditional density of an observation  $Z_i$ , given  $y_i$ , is known and denoted  $g(z_i|y_i)$ . Assume the observations are fixed. Since this will always only be considered as a function of  $y_i$ , following standard practice, we introduce the shorthand notation  $G_i(y_i) = g(z_i|y_i)$ , where the subscript  $i$  encapsulates the dependence on  $z_i$ . Note that the formulation discussed here, that is for  $\hat{\eta}_n^\infty$ , also allows one to consider general Feynman–Kac models (of the form (15)), rather than just the filters that are focussed upon in this section. The following assumptions will be made on the likelihood functions  $\{G_i\}$ , which are sufficient for the mathematical results in this paper.

**Assumption 3.1** There are  $c > 1$  and  $C > 0$ , such that for all  $n > 0$ , and  $v, v' \in \mathbb{R}^d$ ,  $G_n$  satisfies

- (i)  $c^{-1} < G_n(v) < c$ ;
- (ii)  $|G_n(v) - G_n(v')| \leq C|v - v'|$ .

In practice, as discussed earlier on  $Q^\infty$  is typically analytically intractable, in the sense that we cannot sample from it exactly, and we do not know how to evaluate a non-negative and un-biased estimate of it, or even an un-normalized version of it. As a result, we will focus upon targets associated to a discretization, i.e. of the type

$$\hat{\eta}_n^l(dy_{1:n}) \propto \left[ \prod_{i=1}^n G_i(y_i) Q^l(y_{i-1}, dy_i) \right], \tag{16}$$

for  $l < \infty$ , where  $Q^l$  is defined by  $k_l$  iterates of the recursion in (6). Note that we will use  $\hat{\eta}_n^l$  as the notation for measure and density, with the use clear from the context, where  $l = 0, 1, \dots$

The objective is to compute the expectation of functionals with respect to this measure, particularly at the last coordinate. For any bounded and measurable function  $f : \mathbb{R}^d \rightarrow \mathbb{R}$ ,  $n \geq 1$ , we will use the notation

$$\hat{\eta}_n^l(f) := \int_{\mathbb{R}^{dn}} f(y_n) \hat{\eta}_n^l(dy_{1:n}). \tag{17}$$

Often of interest is the computation of the unnormalized measure. That is, for any bounded and measurable function  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  define, for  $n \geq 1$

$$\hat{\zeta}_n^l(f) := \int_{\mathbb{R}^{dn}} f(y_n) \left[ \prod_{i=1}^n G_i(y_i) Q^l(y_{i-1}, dy_i) \right]. \tag{18}$$

In the context of the model under study,  $\hat{\zeta}_n^l(1)$  is the marginal likelihood.

Henceforth  $Y_{1:n}^l$  will be used to denote a draw from  $\hat{\eta}_n^l$ . The vanilla case described earlier can be viewed as the special example in which  $G_i \equiv 1$  for all  $i$ . Following standard practice, realizations of random variables will be denoted with small letters. The randomness of the samples will be recalled again for MSE calculations, over potential realizations.

#### 3.2 Particle filtering

We will now describe the particle filter, which is capable of consistently estimating terms of the form (17) and (18), for any fixed  $l$ . Consistent means it is asymptotically exact in the limit of infinite particles. The particle filter has been studied and used extensively (see for example Moral 2004; Doucet et al. 2001) in many practical applications of interest.

For a given level  $l$ , Algorithm 3 gives the standard particle filter. The weights are defined as for  $k \geq 1$

$$w_k^{l,(i)} = w_{k-1}^{l,(i)} \frac{G_k(y_k^{l,(i)})}{\sum_{j=1}^{N_l} w_{k-1}^{l,(j)} G_k(y_k^{l,(j)})} \tag{19}$$

with the convention that  $w_0^{l,(i)} = 1$ . Note that the abbreviation *ESS* stands for effective sample size, which measures the variability of weights at time  $k$  of the algorithm. In the analysis to follow  $H = 1$  in Algorithm 3 (or rather it’s extension in the next section), but this is not the case in our numerical implementations.

Moral (2004) (along with many other authors) have shown that for upper-bounded and non-negative,  $G_i$ , and suitable assumptions on the dynamics (for example Assumption 2.1 is sufficient), for  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  bounded and measurable

(these conditions can be relaxed), at step 3 of Algorithm 3, the estimate

$$\sum_{i=1}^{N_l} w_n^{l,(i)} f\left(y_n^{l,(i)}\right)$$

will converge almost surely to (17). In addition, if  $H = 1$  in Algorithm 3,

$$\left[ \prod_{i=1}^{n-1} \frac{1}{N_l} \sum_{j=1}^{N_l} G_i(y_i^{l,(j)}) \right] \frac{1}{N_l} \sum_{j=1}^{N_l} G_n\left(y_n^{l,(j)}\right) f\left(y_n^{l,(j)}\right)$$

will converge almost surely to (18).

**Algorithm 3 : Particle filter**

0. Set  $k = 1$ ; for  $i = 1, \dots, N_l$ , draw  $Y_1^{l,(i)} \sim \mathcal{Q}^l(y_0, \cdot)$
1. Compute weights  $\{w_1^{l,(i)}\}_{i=1}^{N_l}$  using (19)
2. Compute  $ESS = \left(\sum_{i=1}^{N_l} (w_k^{l,(i)})^2\right)^{-1}$ .  
If  $ESS/N_l < H$  (for some threshold  $H$ ), resample the particles  $\{Y_k^{l,(i)}\}_{i=1}^{N_l}$  and set all weights to  $w_k^{l,(i)} = 1/N_l$ . Denote the resampled particles  $\{\hat{Y}_k^{l,(i)}\}_{i=1}^{N_l}$ .  
Else set  $\{\hat{Y}_k^{l,(i)}\}_{i=1}^{N_l} = \{Y_k^{l,(i)}\}_{i=1}^{N_l}$ .
3. Set  $k = k + 1$ ; if  $k = n + 1$  stop;  
for  $i = 1 \dots, N_l$ , draw  $Y_k^{l,(i)} \sim \mathcal{Q}^l(\hat{y}_{k-1}^{l,(i)}, \cdot)$ ;  
compute weights  $\{w_k^{l,(i)}\}_{i=1}^{N_l}$  by using (19). Go to 2.

**3.3 Multilevel particle filter**

We now describe the multilevel particle filter of Jasra et al. (2017) for the context considered here. The basic idea is to run  $L + 1$  independent algorithms, the first a particle filter as in the previous section and the remaining, coupled particle filters. The particle filter will sequentially (in time) approximate  $\hat{\eta}_k^0$  and the coupled filters will sequentially approximate the couples  $(\hat{\eta}_k^0, \hat{\eta}_k^1, \dots, (\hat{\eta}_k^{L-1}, \hat{\eta}_k^L)$ . Each (coupled) particle filter will be run with  $N_l$  particles.

The most important step in the MLPF is the coupled resampling step, which maximizes the probability of resampled indices being the same at the coarse and fine levels. Denote the coarse and fine particles at level  $l \geq 1$  and step  $k \geq 1$  as  $(Y_k^{l,(i)}(l), Y_k^{l-1,(i)}(l))$ , for  $i = 1, \dots, N_l$ . Equation (19) is replaced by the following, for  $k \geq 1$

$$w_k^{l,(i)}(l) = w_{k-1}^{l,(i)}(l) \frac{G_k(y_k^{l,(i)}(l))}{\sum_{j=1}^{N_l} w_{k-1}^{l,(j)}(l) G_k(y_k^{l,(j)}(l))} \tag{20}$$

$$w_k^{l-1,(i)}(l) = w_{k-1}^{l-1,(i)}(l) \frac{G_k(y_k^{l-1,(i)}(l))}{\sum_{j=1}^{N_l} w_{k-1}^{l-1,(j)}(l) G_k(y_k^{l-1,(j)}(l))} \tag{21}$$

with the convention that  $w_0^{l,(i)}(l) = w_0^{l-1,(i)}(l) = 1$ .

**Algorithm 4 Coupled Resampling Procedure**

- For  $\ell = 1, \dots, N_l$   
 With probability  $\sum_{i=1}^{N_l} \min\{w_k^{l,(i)}(l), w_k^{l-1,(i)}(l)\}$ ,
- (i) Sample  $J$  with probability proportional to  $\min\{w_k^{l,(i)}(l), w_k^{l-1,(i)}(l)\}$  for  $i = 1, \dots, N_l$ , where the weights are computed according to (20).
  - (ii) Set  $(\hat{Y}_k^{l,(\ell)}(l), \hat{Y}_k^{l-1,(\ell)}(l)) = (Y_k^{l,(j)}(l), Y_k^{l-1,(j)}(l))$ .
- Else, with probability  $1 - \sum_{i=1}^{N_l} \min\{w_k^{l,(i)}(l), w_k^{l-1,(i)}(l)\}$ ,
- (i) Sample  $J_l$  with probability proportional to  $w_k^{l,(i)}(l) - \min\{w_k^{l,(i)}(l), w_k^{l-1,(i)}(l)\}$  for  $i = 1, \dots, N_l$ ,
  - (ii) Sample  $J_{l-1} \perp J_l$  with probability proportional to  $w_k^{l-1,(i)}(l) - \min\{w_k^{l,(i)}(l), w_k^{l-1,(i)}(l)\}$  for  $i = 1, \dots, N_l$ ,
  - (iii) Set  $\hat{Y}_k^{l,(\ell)}(l) = Y_k^{l,(j_l)}(l)$ , and  $\hat{Y}_k^{l-1,(\ell)}(l) = Y_k^{l-1,(j_{l-1})}(l)$ .

We set  $H = 1$  (as in Algorithm 3), but it need not be the case. Recall that the case  $l = 0$  is just a particle filter. For each  $1 \leq l \leq L$  the following procedure is run independently.

**Algorithm 5 Multilevel Particle filter**

0. Set  $k = 1$ ; for  $i = 1, \dots, N_l$ , draw  $(Y_1^{l,(i)}(l), Y_1^{l-1,(i)}(l)) \sim M^l((y_0, y_0), \cdot)$ .
1. Compute weights  $\{(w_1^{l,(i)}(l), w_1^{l-1,(i)}(l))\}_{i=1}^{N_l}$  using (20)
2. Compute  $ESS = \min\left\{\left(\sum_{i=1}^{N_l} (w_k^{l,(i)}(l))^2\right)^{-1}, \left(\sum_{i=1}^{N_l} (w_k^{l-1,(i)}(l))^2\right)^{-1}\right\}$ .  
If  $ESS/N_l < H$ , resample the particles  $\left\{\left(\hat{Y}_k^{l,(i)}(l), \hat{Y}_k^{l-1,(i)}(l)\right)\right\}_{i=1}^{N_l}$  according to Algorithm 4, and set all weights to  $w_k^{l,(i)}(l) = w_k^{l-1,(i)}(l) = 1/N_l$ .  
Else set  $\left\{\left(\hat{Y}_k^{l,(i)}(l), \hat{Y}_k^{l-1,(i)}(l)\right)\right\}_{i=1}^{N_l} = \left\{\left(Y_k^{l,(i)}(l), Y_k^{l-1,(i)}(l)\right)\right\}_{i=1}^{N_l}$ .
3. Set  $k = k + 1$ ; if  $k = n + 1$  stop;  
for  $i = 1 \dots, N_l$ , draw  $(Y_k^{l,(i)}(l), Y_k^{l-1,(i)}(l)) \sim M^l\left(\hat{y}_{k-1}^{l,(i)}(l), \hat{y}_{k-1}^{l-1,(i)}(l), \cdot\right)$ ;  
compute weights  $\{(w_k^{l,(i)}(l), w_k^{l-1,(i)}(l))\}_{i=1}^{N_l}$  by using (20). Go to 2.

The samples generated by the particle filter for  $l = 0$  at time  $k$  are denoted  $Y_k^{0,(i)}(0)$ ,  $i \in \{1, \dots, N_0\}$  (we are assuming  $H = 1$ ).

To estimate the quantities (17) and (18) (with  $l = L$ ) (Jasra et al. 2017, 2018) show that in the case of discretized diffusion processes

$$\hat{\eta}_n^{ML,L}(f) = \sum_{l=1}^L \left( \frac{\sum_{i=1}^{N_l} G_n(y_n^{l,(i)}(l)) f(y_n^{l,(i)}(l))}{\sum_{i=1}^{N_l} G_n(y_n^{l,(i)}(l))} \right)$$



$$\begin{aligned}
 & - \frac{\sum_{i=1}^{N_l} G_n(y_n^{l-1,(i)}(l)) f(y_n^{l-1,(i)}(l))}{\sum_{i=1}^{N_l} G_n(y_n^{l-1,(i)}(l))} \\
 & + \frac{\sum_{i=1}^{N_0} G_n(y_n^{0,(i)}(0)) f(y_n^{0,(i)}(0))}{\sum_{i=1}^{N_0} G_n(y_n^{0,(i)}(0))}
 \end{aligned}$$

and

$$\begin{aligned}
 \hat{\xi}_n^{\text{ML,L}}(f) &= \sum_{l=1}^L \left( \left[ \prod_{i=1}^{n-1} \frac{1}{N_l} \sum_{j=1}^{N_l} G_i(y_i^{l,(j)}(l)) \right] \right. \\
 & \frac{1}{N_l} \sum_{j=1}^{N_l} G_n(y_n^{l,(j)}(l)) f(y_n^{l,(j)}(l)) \\
 & \left. - \left[ \prod_{i=1}^{n-1} \frac{1}{N_l} \sum_{j=1}^{N_l} G_i(y_i^{l-1,(j)}(l)) \right] \right. \\
 & \frac{1}{N_l} \sum_{j=1}^{N_l} G_n(y_n^{l-1,(j)}(l)) f(y_n^{l-1,(j)}(l)) \\
 & \left. + \left[ \prod_{i=1}^{n-1} \frac{1}{N_0} \sum_{j=1}^{N_0} G_i(y_i^{0,(j)}(0)) \right] \right. \\
 & \left. \frac{1}{N_0} \sum_{j=1}^{N_0} G_n(y_n^{0,(j)}(0)) f(y_n^{0,(j)}(0)) \right) \quad (22)
 \end{aligned}$$

converge almost surely to  $\hat{\eta}_n^L(f)$  and  $\hat{\xi}_n^L(f)$  respectively, as  $\min\{N_l\} \rightarrow \infty$ . Furthermore, both can significantly improve over the particle filter, for  $L$  and  $\{N_l\}_{l=1}^L$  appropriately chosen to depend upon a target mean square error (MSE). By improve, we mean that the work is less than the particle filter to achieve a given MSE with respect to the continuous time limit, under appropriate assumptions on the diffusion. We show how the  $N_0, \dots, N_L$  can be chosen in Sect. 3.3.1. Note that for positive  $f$  the estimator above  $\hat{\xi}_n^{\text{ML,L}}(f)$  can take negative values with positive probability.

We remark that the coupled resampling method can be improved as in Sen et al. (2018). We also remark that the approaches of Houssineau et al. (2018) and Jacob et al. (2016) could potentially be used here. However, none of these articles has sufficient supporting theory to verify a reduction in cost of the ML procedure.

### 3.3.1 Theoretical result

We conclude this section with a technical theorem. We consider only  $\hat{\eta}_n^{\text{ML,L}}(f)$ , but this can be extended to  $\hat{\xi}_n^{\text{ML,L}}(f)$ , similarly to Jasra et al. (2018). The proofs are given in “Appendix”.

Define  $\mathcal{B}_b(\mathbb{R}^d)$  as the bounded, measurable and real-valued functions on  $\mathbb{R}^d$  and  $\text{Lip}(\mathbb{R}^d)$  as the globally Lipschitz real-valued functions on  $\mathbb{R}^d$ . Define the space  $\mathcal{A} =$

**Table 1** The three cases of MLPF, and associated cost  $\mathcal{C}(\varepsilon)$ .  $\beta$  is as Lemma A3

CASE	$\mathcal{C}(\varepsilon)$
$\beta > 2$	$\mathcal{O}(\varepsilon^{-2})$
$\beta = 2$	$\mathcal{O}(\varepsilon^{-2} \log(\varepsilon)^2)$
$\beta < 2$	$\mathcal{O}(\varepsilon^{-2+(\beta-2)/(\beta)})$

$\mathcal{B}_b(\mathbb{R}^d) \cap \text{Lip}(\mathbb{R}^d)$  with the norm  $\|\varphi\| = \sup_{x \in \mathbb{R}^d} |\varphi(x)| + \sup_{x, y \in \mathbb{R}^d} \frac{|\varphi(x) - \varphi(y)|}{|x - y|}$ .

The following assumptions will be required.

**Assumption 3.2** There exists  $C, \beta_1 > 0$ , such that for all  $h_l > 0$ , there exists a  $\delta_l(h_l) > 0$  such that  $\delta_l(h_l) \leq Ch_l^{\beta_1}$  and  $h_l = 1/\nu(B_{\delta_l(h_l)}^c)$ .

Denote by  $\check{Q}^{l,l-1}((y, y'), \cdot)$  the coupling of the Markov transitions  $Q^l(y, \cdot)$  and  $Q^{l-1}(y', \cdot)$ ,  $(y, y') \in \mathbb{R}^{2d}$  as in Algorithm 2.

**Assumption 3.3**  $\mathbb{E}[\text{COST}(\check{Q}^{l,l-1})] = \mathcal{O}(h_l^{-1})$ , where  $\mathbb{E}[\text{COST}(\check{Q}^{l,l-1})]$  is the expected cost to simulate one sample from the kernel  $\check{Q}^{l,l-1}$ .

Below  $\mathbb{E}$  denotes expectation w.r.t. the law of the particle system.

**Theorem 1** Assume (2.1, 3.1,3.2, 3.3). Then for any  $n \geq 0, f \in \mathcal{A}$ , there exists a  $C > 0$  such that for  $\varepsilon > 0$  given and  $L > 0, \{N_l\}_{l=0}^L$  depending only upon  $\varepsilon$ , the following estimate holds

$$\mathbb{E} \left[ \left( \hat{\eta}_n^{\text{ML,L}}(f) - \hat{\eta}_n^\infty(f) \right)^2 \right] \leq C\varepsilon^2,$$

for the cost  $\mathcal{C}(\varepsilon) := \mathbb{E}[\text{COST}(\varepsilon)]$  given in the second column of Table 1.

**Proof** The proof is essentially identical to Jasra et al. (2017, Theorem 4.3). The only difference is to establish analogous results to Jasra et al. (2017, Appendix D); this is done in the appendix of this article.  $\square$

## 4 Numerical examples

In this section, we compare our proposed multilevel particle filter method with the vanilla particle filter method. A target accuracy parameter  $\varepsilon$  will be specified and the cost to achieve an error below this target accuracy will be estimated. The performance of the two algorithms will be compared in two applications of SDEs driven by general Lévy process: filtering of a partially observed Lévy process (S&P 500 stock price data) and pricing of a path dependent option. In each of these two applications, we let  $X = \{X_t\}_{t \in [0, K]}$  denote a symmetric stable Lévy process, i.e.  $X$  is a  $(\nu, \Sigma, b)$ -Lévy

process, and the Lebesgue density of the Lévy measure is given by

$$\nu(dx) = c|x|^{-1-\phi} \mathbb{1}_{[-x^*,0)}(x)dx + c|x|^{-1-\phi} \mathbb{1}_{(0,x^*]}(x)dx, \quad x \in \mathbb{R} \setminus \{0\}, \tag{23}$$

with  $c > 0$ ,  $x^* \geq 1$  (the truncation threshold) and index  $\phi \in (0, 2)$ . The parameters  $c$  and  $x^*$  are both 1 for all the examples considered. Notice Assumption 2.1(ii) is satisfied. The Lévy-driven SDE considered here has the form

$$dY_t = a(Y_{t-})dX_t, \quad Y_0 = y_0, \tag{24}$$

with  $y_0$  assumed known. In the examples illustrated below, we take  $a(Y_t) = Y_t$ ,  $y_0 = 1$ , and  $\phi = 0.5$ . The first condition of Assumption 2.1(i) is satisfied for this choice of  $a$ , but the second condition is not, without modifying  $a$  outside of a compact set.

**Remark 1** (Symmetric stable Lévy process of index  $\phi \in (0, 2)$ )

In approximating the Lévy-driven SDE (24), Theorem 2 of Dereich and Heidenreich (2011) provided asymptotic error bounds for the strong approximation by the Euler scheme. If the driving Lévy process  $X_t$  has no Brownian component, that is  $\Sigma = 0$ , then the  $L^2$ -error, denoted  $\sigma_{h_l}^2$ , is bounded by

$$\sigma_{h_l}^2 \leq C(\sigma^2(\delta_l) + |b - F_0^l|^2 h_l^2),$$

and for  $\Sigma > 0$ ,

$$\sigma_{h_l}^2 \leq C(\sigma^2(\delta_l) + h_l |\log(h_l)|),$$

for a fixed constant  $C < \infty$  (that is the Lipschitz constant), where  $\sigma^2(\delta_l) := \int_{B_{\delta_l}} |x|^2 \nu(dx)$ . Recall that  $\delta_l(h_l)$  is chosen such that  $h_l = 1/\nu(B_{\delta_l}^c)$ . Observe that for  $\delta_l \geq x^*$  one has  $\nu(B_{\delta_l}^c) = 0$  and  $h_l = \infty$ . As mentioned earlier, we are concerned with asymptotically small values of  $h_l$  and  $\delta_l$  here, and so we let  $\delta_l < x^*$ , without any loss of generality. Then one obtains the analytical expression

$$\sigma^2(\delta_l) = \frac{2c}{2-\phi} \delta_l (h_l)^{2-\phi} \leq C \delta_l^{2-\phi}, \tag{25}$$

for some constant  $C > 0$ .

One can also analytically compute

$$\nu(B_{\delta_l}^c) = \frac{2c(\delta_l^{-\phi} - x^{*- \phi})}{\phi}.$$

Now one can solve  $\nu(B_{\delta_l}^c) = h_l$ , and obtain

$$\delta_l = \left( \frac{2ch_l}{\phi + x^{*- \phi} h_l} \right)^{1/\phi}. \tag{26}$$

Then, one finds that

$$\delta_l = \mathcal{O}(h_l^{1/\phi}),$$

hence verifying Assumption 3.2 for this example. Using (25)–(26) and the error bounds for  $\Sigma = 0$ , one can straightforwardly obtain strong error rates for the approximation of an SDE driven by a stable Lévy process in terms of the single accuracy parameter  $h_l$ . This is given by

$$\sigma_{h_l}^2 \leq C(h_l^{(2-\phi)/\phi} + |b - F_0^l|^2 h_l^2).$$

Thus, if  $b - F_0^l \neq 0$ , the strong error rate  $\beta$  of Lemma A3 associated with a particular discretization level  $h_l$ , is given by

$$\beta = \min \left( \frac{2-\phi}{\phi}, 2 \right). \tag{27}$$

Otherwise it is just given by  $(2 - \phi)/\phi$ .

In the examples considered below, the original Lévy process has no drift and Brownian motion components, that is  $\Sigma = b = 0$ . Due to the linear drift correction  $F_0^l$  in the compensated compound Poisson process, the random jump times are refined such that the time differences between successive jumps are bounded by the accuracy parameter  $h_l$  associated with the Euler discretization approximation methods in (5) and (13)–(14). However, since  $F_0^l = 0$  here, due to symmetry, this does not affect the rate, as described in Remark 1.

We start with verification of the weak and strong error convergence rates,  $\alpha$  and  $\beta$  for the forward model. To this end the quantities  $|\mathbb{E}[Y_1^l - Y_1^{l-1}]|$  and  $\mathbb{E}[|Y_1^l - Y_1^{l-1}|^2]$  are computed over increasing levels  $l$ . Figure 1 shows these computed values plotted against  $h_l$  on base-2 logarithmic scales. A fit of a linear model gives rate  $\alpha = 1.3797$ , and similar simulation experiment gives  $\beta = 2.7377$ . This is consistent with the rate  $\beta = 3$  and  $\alpha = \beta/2$  from Remark 1 (27).

We begin our comparison of the MLPF and PF algorithms starting with the filtering of a partially observed Lévy-driven SDE and then consider the knock out barrier call option pricing problem.

### 4.1 Partially observed data

In this section we consider filtering a partially observed Lévy process. Recall that the Lévy-driven SDE takes the form (24). In addition, partial observations  $\{z_1, \dots, z_n\}$  are available with  $Z_k$  obtained at time  $k$  and  $Z_k | (Y_k = y_k)$  has a density function  $G_k(y_k)$  (with observation is omitted from the

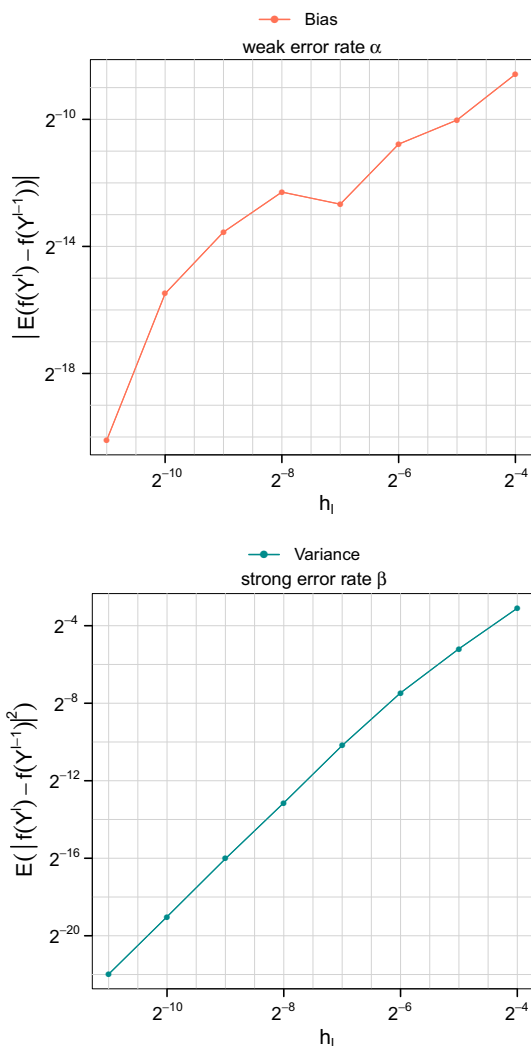


Fig. 1 Empirical weak and strong error rates estimates

notation and appearing only as subscript  $k$ ). The observation density is Gaussian with mean  $y_k$  and variance 1. We aim to estimate  $\mathbb{E}[f(Y_k)|z_{1:k}]$  for some test function  $f(y)$ . In this application, we consider the real daily S&P 500 log return data (from August 3, 2011 to July 24, 2015, normalized to unity variance). We shall take the test function  $f(y) = e^y$  for the example considered below, which we note does not satisfy the assumptions of Theorem 1, and hence challenges the theory. In fact the results are roughly equivalent to the case  $f(y) = e^y \mathbb{I}_{\{|y| < 10\}}$ , where  $\mathbb{I}_A$  is the indicator function on the set  $A$ , which was also considered and does satisfy the required assumptions.

The error-versus-cost plots on base 10 logarithmic scales for PF and MLPF are shown in Fig. 2. The fitted linear model of log MSE against log Cost has a slope of  $-0.667$  and  $-0.859$  for PF and MLPF respectively. These results again verify numerically the expected theoretical asymptotic

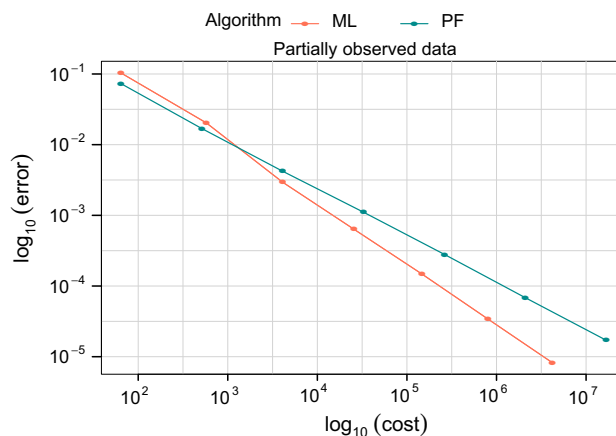


Fig. 2 Mean square error against computational cost for filtering with partially observed data

behaviour of computational cost as a function of MSE for both standard cost and ML cost.

### 4.2 Barrier option

Here we consider computing the value of a discretely monitored knock out barrier option (see e.g. Glasserman (2004) and the references therein). Let  $Y_0 \in [a, b]$  for some  $0 < a < b < +\infty$  known and let  $Q^\infty(y_{i-1}, dy)$  be the transition kernel of the process as in (24). Then the value of the barrier option (up-to a known constant) is

$$\int_{\mathbb{R}^n} \max\{y_n - S, 0\} \prod_{i=1}^n \mathbb{I}_{[a,b]}(y_i) Q^\infty(y_{i-1}, dy_i),$$

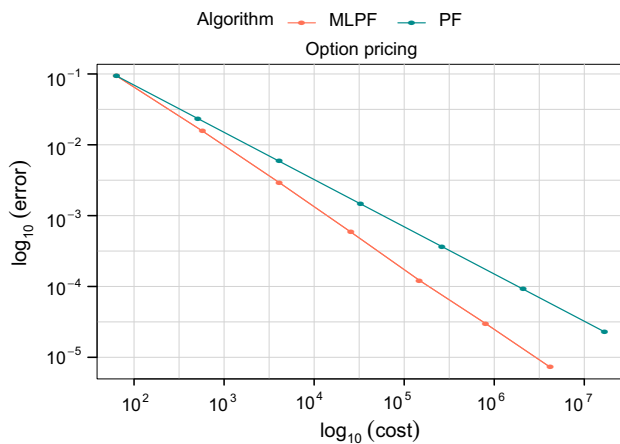
for  $S > 0$  given. As seen in Jasra and Moral (2011) the calculation of the barrier option is non-trivial, in the sense that even importance sampling may not work well. We consider the (time) discretized version

$$\int_{\mathbb{R}^n} \max\{y_n - S, 0\} \prod_{i=1}^n \mathbb{I}_{[a,b]}(y_i) Q^l(y_{i-1}, dy_i).$$

Define a sequence of probabilities,  $k \in \{1, \dots, n\}$ ,

$$\begin{aligned} \hat{\eta}_k^l(dy_{1:k}) &\propto \tilde{G}_k(y_k) \prod_{i=1}^k \mathbb{I}_{[a,b]}(y_i) Q^l(y_{i-1}, dy_i) \\ &= \prod_{i=1}^k \left( \frac{\tilde{G}_i(y_i)}{\tilde{G}_{i-1}(y_{i-1})} \right) \mathbb{I}_{[a,b]}(y_i) Q^l(y_{i-1}, dy_i) \end{aligned} \quad (28)$$

for some non-negative collection of functions  $\tilde{G}_k(y_k)$ ,  $k \in \{1, \dots, n\}$  to be specified. Recall that  $\hat{\xi}_n^l$  denotes the unnormalized density associated to  $\hat{\eta}_n^l$ . Then the value of the time discretized barrier option is exactly



**Fig. 3** Mean square error against computational cost for the knock out barrier option example

$$\hat{\zeta}_n^l \left( \frac{f}{\tilde{G}_n} \right) = \int_{\mathbb{R}^n} \max\{y_n - S, 0\} \prod_{i=1}^n \mathbb{I}_{[a,b]}(y_i) Q^l(y_{i-1}, dy_i), \tag{29}$$

where  $f(y_n) = \max\{y_n - S, 0\}$ . Thus, we can apply the MLPF targeting the sequence  $\{\hat{\eta}_k^l\}_{k \in \{1, \dots, n\}, l \in \{0, \dots, L\}}$  and use our normalizing constant estimator (22) to estimate (29). If  $\tilde{G}_n = |f|$ , then we have an optimal importance distribution, in the sense that we are estimating the integral of the constant function 1 and the variance is minimal (Rubinstein and Kroese 2016). However, noting the form of the effective potential above (28), this can result in infinite weights (with adaptive resampling as done here), and so some regularization is necessary. We bypass this issue by choosing  $\tilde{G}_k(y_k) = |y_k - S|^{\kappa_k}$ , where  $\kappa_k$  is an annealing parameter with  $\kappa_0 = 0$  and  $\kappa_n = 1$ . We make no claim that this is the best option, but it guides us to something reminiscent of the optimal thing, and with well-behaved weights, in practice. We tried also  $|\max\{y_k - S, \varepsilon\}|^{\kappa_k}$ , with  $\varepsilon = 0.001$ , and the results are almost identical.

For this example we choose  $S = 1.25, a = 0, b = 5, y_0 = 1, n = 100$ . The  $N_l$  are chosen as in the previous example. The error-versus-cost plots for PF and MLPF are shown in Fig. 3. Note that the bullets in the graph correspond to different choices of  $L$  (for both PF and MLPF,  $2 \leq L \leq 8$ ). The fitted linear model of log MSE against log cost has a slope of  $-0.6667$  and  $-0.859$  for PF and MLPF respectively. These numerical results are consistent with the expected theoretical asymptotic behaviour of  $MSE \propto Cost^{-1}$  for the multilevel method. The single level particle filter achieves the asymptotic behaviour of the standard Monte Carlo method with  $MSE \propto Cost^{-2/3}$ .

**Acknowledgements** AJ was supported by Singapore ministry of education AcRF tier 2 Grant R-155-00-161-112 and he is affiliated with the CQF, RMI and ORA cluster at NUS. He was also supported by a King

Abdullah University of Science and Technology Competitive Research Grant round 4, Ref:2584. KJHL was sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U. S. Department of Energy, as well as the School of Mathematics at the University of Manchester. We would also like to thank the anonymous referees for numerous suggestions and corrections which have greatly improved the manuscript.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

### Appendix: Theoretical results

Our proof consists of following the proof of Jasra et al. (2017). To that end all the proofs of Jasra et al. (2017, Appendices A–C) are the same for the approach in this article (note that one needs Lemma A2 of this article along the way). One must verify the analogous results of Jasra et al. (2017, Appendix D), which is what is done in this appendix.

The predictor at time  $n$ , level  $l$ , is denoted as  $\eta_n^l$ . Denote the total variation norm as  $\|\cdot\|_{TV}$ . For  $\varphi \in \text{Lip}(\mathbb{R}^d)$ ,  $\|\varphi\|_{\text{Lip}} := \sup_{x,y \in \mathbb{R}^d} \frac{|\varphi(x) - \varphi(y)|}{|x - y|}$  is the Lipschitz constant. For ease (in abuse) of notation,  $Q^l$  defined by  $k_l$  iterates of the recursion in (6) is used as a Markov kernel below. We set for  $\varphi \in \mathcal{B}_b(\mathbb{R}^d)$ ,  $y \in \mathbb{R}^d$

$$Q^l(\varphi)(y) := \int_{\mathbb{R}^d} \varphi(y') Q^l(y, y') dy'$$

Recall, for  $l \geq 1$ ,  $\check{Q}^{l,l-1}((y, y'), \cdot)$  is the coupling of the kernels  $Q^l(y, \cdot)$  and  $Q^{l-1}(y', \cdot)$  as in Algorithm 2. For  $\varphi \in \mathcal{B}_b(\mathbb{R}^{2d})$  we use the notation for  $(y, y') \in \mathbb{R}^{2d}$ :

$$\begin{aligned} &\check{Q}^{l,l-1}(\varphi)(y, y') \\ &:= \int_{\mathbb{R}^{2d}} \varphi(y^l, y^{l-1}) \check{Q}^{l,l-1}((y, y'), d(y^l, y^{l-1})) \end{aligned}$$

and note that for  $\varphi \in \mathcal{B}_b(\mathbb{R}^d)$

$$\begin{aligned} \check{Q}^{l,l-1}(\varphi \otimes 1)(y, y') &= Q^l(\varphi)(y), \\ \check{Q}^{l,l-1}(1 \otimes \varphi)(y, y') &= Q^{l-1}(\varphi)(y') \end{aligned}$$

where  $\otimes$  denotes the tensor product of functions, e.g.  $\varphi \otimes 1$  denotes  $\varphi(y^l)$  in the integrand associated to  $\check{Q}^{l,l-1}((y, y'), d(y^l, y^{l-1}))$ .

Let  $T_j(t) = \max\{T_j \in \mathbb{T}; T_j < t\}$ , and let  $(\Delta X)_t^l = X_t^l - X_{T_j(t)}^l$ , where  $X_t^l = X_{T_j(t)}^l + (b - F_0^l)(t - T_j(t)) + (W_t - W_{T_j(t)})$  is a continuation of the discretized Lévy process (5).

Define the continuation of the discretized driven process by

$$Y_t^l = Y_{T_j(t)}^l + a(Y_{T_j(t)}^l)(\Delta X)_t^l.$$

Let  $Y_1^l \sim Q^l(y, \cdot)$  and independently  $Y_1^{l'} \sim Q^l(y', \cdot)$ . We denote expectations w.r.t. these random variables as  $\mathbb{E}$ .

**Lemma A1** Assume (2.1). Then there exists a  $C > 0$  such that for any  $L \geq l \geq 0$ , and  $(y, y') \in \mathbb{R}^{2d}$

$$\mathbb{E}|Y_1^l - Y_1^{l'}|^2 \leq C|y - y'|^2.$$

**Proof** Let  $t \in [0, 1]$ . We have

$$\begin{aligned} |Y_t^l - Y_t^{l'}|^2 &= |Y_{T_j(t)}^l - Y_{T_j(t)}^{l'}|^2 + 2 \left( Y_{T_j(t)}^l - Y_{T_j(t)}^{l'} \right)^T \\ &\quad \left( a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right) (\Delta X)_t^l \\ &\quad + \left| \left( a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right) (\Delta X)_t^l \right|^2. \end{aligned}$$

Let  $N = \#\{T_j \leq 1\}$  be the number of time-steps before time 1, and denote  $\mathbb{T} = \{\tilde{T}_1, \dots, \tilde{T}_N, N\}$ , where  $\tilde{T}_j$  and  $T_j$  are generated by Algorithm 1. The sigma algebra generated by these random variables is denoted  $\sigma(\mathbb{T})$ .

Following from the independence of  $Y_{T_j(t)}^l$  and  $(\Delta X)_t^l$  conditioned on  $\sigma(\mathbb{T})$ , we have

$$\begin{aligned} \mathbb{E} \left[ |Y_t^l - Y_t^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] &\leq \mathbb{E} \left[ |Y_{T_j(t)}^l - Y_{T_j(t)}^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] \\ &\quad + 2\mathbb{E} \left[ \left( Y_{T_j(t)}^l - Y_{T_j(t)}^{l'} \right)^T \left( a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right) \middle| \sigma(\mathbb{T}) \right] \\ &\quad \mathbb{E} \left[ (\Delta X)_t^l \middle| \sigma(\mathbb{T}) \right] \\ &\quad + \mathbb{E} \left[ \left| a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right|^2 \middle| \sigma(\mathbb{T}) \right] \mathbb{E} \left[ |(\Delta X)_t^l|^2 \middle| \sigma(\mathbb{T}) \right]. \end{aligned} \tag{30}$$

The inequality is a result of the last term which uses the definition of the matrix 2 norm. Note that  $\mathbb{E}[W_t] = \mathbb{E}[L_t] = 0$ , so that  $\mathbb{E}[(\Delta X)_t^l \middle| \sigma(\mathbb{T})] = (b - F_0^l)(t - T_j(t))$ . In addition, (4), Jensen’s inequality and the fact that  $B_{\delta_l}^c \subset \mathbb{R}^d$ , together imply that

$$|F_0^l|^2 \leq \int_{B_{\delta_l}^c} |x|^2 \nu(dx) \leq \int_{\mathbb{R}^d} |x|^2 \nu(dx). \tag{31}$$

We have

$$\begin{aligned} &\mathbb{E} \left[ \left( Y_{T_j(t)}^l - Y_{T_j(t)}^{l'} \right)^T \left( a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right) \middle| \sigma(\mathbb{T}) \right] \mathbb{E} \left[ (\Delta X)_t^l \middle| \sigma(\mathbb{T}) \right] \\ &= \mathbb{E} \left[ \left( Y_{T_j(t)}^l - Y_{T_j(t)}^{l'} \right)^T \left( a(Y_{T_j(t)}^l) - a(Y_{T_j(t)}^{l'}) \right) \middle| \sigma(\mathbb{T}) \right] (b - F_0^l)(t - T_j(t)) \\ &\leq C^2 h_l \mathbb{E} \left[ |Y_{T_j(t)}^l - Y_{T_j(t)}^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] \end{aligned} \tag{32}$$

The inequality follows from Cauchy-Schwarz, definition of the matrix 2 norm, Assumption 2.1(i) and (ii) in connection with (31) and the definition of the construction of  $\{T_j\}$  in Algorithm 1, so that  $|t - T_j(t)| \leq h_l$ .

Note also

$$\begin{aligned} \mathbb{E} \left[ |(\Delta X)_t^l|^2 \middle| \sigma(\mathbb{T}) \right] &\leq C^2(|t - T_j(t)| \\ &\quad + |t - T_j(t)|^2) \leq C^2 h_l, \end{aligned} \tag{33}$$

by Assumption 2.1 (ii), and since  $h_l \leq 1$  by definition. Returning to (30), and using (33) and (32), and Assumption 2.1 (i) again on the last term, we have

$$\begin{aligned} \mathbb{E} \left[ |Y_t^l - Y_t^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] &\leq \mathbb{E} \left[ |Y_{T_j(t)}^l - Y_{T_j(t)}^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] \\ &\quad (1 + Ch_l), \end{aligned} \tag{34}$$

where the value of the constant is different.

Therefore, in particular

$$\begin{aligned} \mathbb{E} \left[ |Y_{T_{j+1}}^l - Y_{T_{j+1}}^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] &\leq \mathbb{E} \left[ |Y_{T_j}^l - Y_{T_j}^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] \\ &\quad (1 + Ch_l). \end{aligned}$$

By applying (34) recursively, we have

$$\mathbb{E} \left[ |Y_1^l - Y_1^{l'}|^2 \middle| \sigma(\mathbb{T}) \right] \leq |y - y'|^2 (1 + Ch_l)^N.$$

Note that  $\mathbb{P}(N = n) = \frac{(\lambda^{\delta_l})^n}{n!} e^{-\lambda^{\delta_l}}$ , and  $\lambda^{\delta_l} = h_l^{-1}$  by design, as described in Sect. 2.3. Taking expectation with respect to  $\sigma(\mathbb{T})$  gives

$$\begin{aligned} \mathbb{E}|Y_1^l - Y_1^{l'}|^2 &\leq \left( \sum_{n \geq 0} \frac{(h_l^{-1}(1 + Ch_l))^n}{n!} e^{-h_l^{-1}} \right) |y - y'|^2 \\ &= e^C |y - y'|^2. \end{aligned}$$

The result follows by redefining  $C$ . □

**Lemma A2** Assume (2.1). Then there exists a  $C > 0$  such that for any  $L \geq l \geq 0$ ,  $(y, y') \in \mathbb{R}^{2d}$ , and  $\varphi \in \mathcal{B}_b(\mathbb{R}^d) \cap \text{Lip}(\mathbb{R}^d)$

$$|Q^l(\varphi)(y) - Q^l(\varphi)(y')| \leq C \|\varphi\|_{\text{Lip}} |y - y'|.$$

**Proof** We have

$$\begin{aligned} |Q^l(\varphi)(y) - Q^l(\varphi)(y')| &= |\mathbb{E}(\varphi(Y_1^l) - \varphi(Y_1^{l'}))| \\ &\leq (\mathbb{E}|\varphi(Y_1^l) - \varphi(Y_1^{l'})|^2)^{1/2} \\ &\leq \|\varphi\|_{\text{Lip}} (\mathbb{E}|Y_1^l - Y_1^{l'}|^2)^{1/2} \end{aligned}$$

where Jensen has been applied to go to the second line and that  $\varphi \in \text{Lip}(\mathbb{R}^d)$  to the third. The proof is concluded via Lemma A1. □

**Lemma A3** Assume (2.1, 3.2). Then there exists  $C > 0$  such that for any  $L \geq l \geq 1$

$$\sup_{\varphi \in \mathcal{A}} \sup_{y \in \mathbb{R}^d} |\mathcal{Q}^l(\varphi)(y) - \mathcal{Q}^{l-1}(\varphi)(y)| \leq Ch_l^{\frac{\beta}{2}}.$$

**Proof** We have

$$\begin{aligned} & |\mathcal{Q}^l(\varphi)(y) - \mathcal{Q}^{l-1}(\varphi)(y)| \\ &= \left| \int_{\mathbb{R}^{2d}} \varphi(y^l) \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \right. \\ & \quad \left. - \int_{\mathbb{R}^{2d}} \varphi(y^{l-1}) \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \right|. \end{aligned}$$

Using Jensen’s inequality yields

$$\begin{aligned} & |\mathcal{Q}^l(\varphi)(y) - \mathcal{Q}^{l-1}(\varphi)(y)| \\ & \leq \left( \int_{\mathbb{R}^{2d}} (\varphi(y^l) - \varphi(y^{l-1}))^2 \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \right)^{1/2}. \end{aligned}$$

Recall for  $\varphi \in \mathcal{A}$  there exists  $C > 0$  such that  $|\varphi(y^l) - \varphi(y^{l-1})| \leq C|y^l - y^{l-1}|$ . By Dereich and Heidenreich (2011, Theorem 2), there exists  $C > 0$  such that for any  $y \in \mathbb{R}^d$ ,  $l \geq 1$

$$\int_{\mathbb{R}^{2d}} |y^l - y^{l-1}|^2 \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \leq Ch_l^\beta \quad (35)$$

The proof is then easily concluded.  $\square$

**Remark A1** To verify (35) note that  $\delta_l(h_l)$  is chosen as a function of  $h_l$  here for simplicity, and by Assumption 3.2 it can be bounded by  $Ch_l^{\beta_1}$  for some  $\beta_1$ . The bounds in Theorem 2 of Dereich and Heidenreich (2011) can therefore be written as the sum of two terms  $C(h_l^{\beta_1} + h_l^{\beta_2})$ , and  $\beta = \min\{\beta_1, \beta_2\}$ . See remark 1 for calculation of  $\beta$  in the example considered in this paper.

**Lemma A4** Assume (2.1,3,2). Then there exists  $C > 0$  and  $\beta > 0$  such that for any  $L \geq l \geq 1$ , and  $(y, y') \in \mathbb{R}^{2d}$ ,

$$\left( \int_{\mathbb{R}^{2d}} |y^l - y'^l|^2 \check{\mathcal{Q}}^{l,l-1}((y, y'), d(y^l, y'^l)) \right)^{1/2} \leq C(|y - y'| + h_l^{\beta/2})$$

where  $\beta$  is as in Lemma A3.

**Proof** We have

$$\begin{aligned} & \left( \int_{\mathbb{R}^{2d}} |y^l - y'^l|^2 \check{\mathcal{Q}}^{l,l-1}((y, y'), d(y^l, y'^l)) \right)^{1/2} \\ &= \left( \int_{\mathbb{R}^{3d}} |y^l - \tilde{y}^l + \tilde{y}^l - y'^l|^2 \check{\mathcal{Q}}^{l,l-1}((y, y'), d(y^l, y'^l)) \mathcal{Q}^l(y', d\tilde{y}^l) \right)^{1/2} \\ & \leq \left( \int_{\mathbb{R}^{2d}} |y^l - \tilde{y}^l|^2 \mathcal{Q}^l(y, d\tilde{y}^l) \mathcal{Q}^l(y', d\tilde{y}^l) \right)^{1/2} \\ & \quad + \left( \int_{\mathbb{R}^{2d}} |\tilde{y}^l - y'^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^{l-1}(y', d\tilde{y}^{l-1}) \right)^{1/2} \\ & \leq C|y - y'| + \left( \int_{\mathbb{R}^{2d}} |\tilde{y}^l - y'^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^{l-1}(y', d\tilde{y}^{l-1}) \right)^{1/2} \end{aligned}$$

where we have applied Minkowski’s inequality to go to the third line and Lemma A1 to go to the final line. Now

$$\begin{aligned} & \left( \int_{\mathbb{R}^{2d}} |y^l - y'^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^{l-1}(y', d\tilde{y}^{l-1}) \right)^{1/2} \\ &= \left( \int_{\mathbb{R}^{3d}} |y^l - \tilde{y}^l + \tilde{y}^l - y'^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \check{\mathcal{Q}}^{l,l-1}((y', y'), d(\tilde{y}^l, y'^{l-1})) \right)^{1/2} \\ & \leq \left( \int_{\mathbb{R}^{2d}} |y^l - \tilde{y}^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^l(y', d\tilde{y}^l) \right)^{1/2} \\ & \quad + \left( \int_{\mathbb{R}^{2d}} |\tilde{y}^l - y'^l|^2 \check{\mathcal{Q}}^{l,l-1}((y', y'), d(\tilde{y}^l, y'^{l-1})) \right)^{1/2} \end{aligned}$$

where again we have applied Minkowski’s inequality to go to the third line. Then

$$\int_{\mathbb{R}^{2d}} |y^l - \tilde{y}^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^l(y', d\tilde{y}^l) = 0$$

and by (35) we have

$$\left( \int_{\mathbb{R}^{2d}} |y^l - y'^l|^2 \mathcal{Q}^l(y', d\tilde{y}^l) \mathcal{Q}^{l-1}(y', d\tilde{y}^{l-1}) \right)^{1/2} \leq Ch_l^{\frac{\beta}{2}}.$$

The argument is then easily concluded.  $\square$

**Proposition 1** Assume (2.1,3,1,3,2). Then there exists a  $C > 0$  such that for any  $L \geq l \geq 1, n \geq 0$ ,

$$\|\eta_n^l - \eta_n^{l-1}\|_{\text{tv}} \leq Ch_l^{\frac{\beta}{2}}. \quad (36)$$

where  $\beta$  is as Lemma A3.

**Proof** The result follows from the same calculations of the proof of Jasra et al. (2017, Lemma D.2) along with our Lemma A3, which we note is analogous to (32) in Jasra et al. (2017) with  $\alpha = \beta/2$   $\square$

It is remarked that, given our above results, Lemmata D.3 and D.4 as well as Theorem D.5 (all of Jasra et al. (2017)) can be proved for our algorithm by the same arguments as in Jasra et al. (2017) and are hence omitted.

Note that we have proved that:

$$\begin{aligned} & \sup_{\varphi \in \mathcal{A}} \sup_{y \in \mathbb{R}^d} |\mathcal{Q}^l(\varphi)(y) - \mathcal{Q}^{l-1}(\varphi)(y)| \leq Ch_l^{\frac{\beta}{2}}, \\ & \left| \int_{\mathbb{R}^{2d}} \varphi(y^l) \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \right. \\ & \quad \left. - \int_{\mathbb{R}^{2d}} \varphi(y^{l-1}) \check{\mathcal{Q}}^{l,l-1}((y, y), d(y^l, y^{l-1})) \right| \leq Ch_l^{\frac{\beta}{2}}, \\ & \int_{\mathbb{R}^{2d}} (\varphi(y^l) - \varphi(y^{l-1}))^2 \check{\mathcal{Q}}^{l,l-1}((y, y'), d(y^l, y^{l-1})) \leq Ch_l^\beta, \end{aligned}$$

for all  $\varphi \in \mathcal{A}$ . This provides Jasra et al. (2017, Assumption 4.2. (i) & (ii)), with  $\alpha$  (as in Jasra et al. (2017)) equal to  $\beta/2$ .

## References

- Applebaum, D.: Lévy Processes and Stochastic Calculus. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge (2004)
- Asmussen, S., Rosiński, J.: Approximations of small jumps of Lévy processes with a view towards simulation. *J. Appl. Prob.* **38**(2), 482–493 (2001)
- Bertoin, J.: Lévy Processes. Cambridge Tracts in Mathematics, vol. 121. Cambridge University Press, Cambridge (1996)
- Cont, R., Tankov, P.: Financial Modelling with Jump Processes. Financial Mathematics Series. Chapman & Hall, Boca Raton (2004)
- Del Moral, P.: Feynman–Kac Formulae: Genealogical and Interacting Particle Systems with Applications. Springer, New York (2004)
- Dereich, S.: Multilevel Monte Carlo algorithms for Lévy-driven SDEs with Gaussian correction. *Ann. Appl. Probab.* **21**(1), 283–311 (2011)
- Dereich, S., Heidenreich, F.: A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations. *Stoch. Proc. Appl.* **121**(7), 1565–1587 (2011)
- Doucet, A., De Freitas, N., Gordon, N.J.: Sequential Monte Carlo Methods in Practice. Springer, New York (2001)
- Ferreiro-Castilla, A., Kyprianou, A.E., Scheichl, R., Suryanarayana, G.: Multilevel Monte Carlo simulation for Lévy processes based on the Wiener–Hopf factorisation. *Stoch. Proc. Appl.* **124**(2), 985–1010 (2014)
- Giles, M.B.: Multilevel Monte Carlo path simulation. *Oper. Res.* **56**(3), 607–617 (2008)
- Giles, M.B., Xia, Y.: Multilevel path simulation for jump-diffusion SDEs. In: Plaskota, L., Woźniakowski, H. (eds.) Monte Carlo and Quasi-Monte Carlo Methods 2010. Springer, Berlin (2012)
- Glasserman, P.: Monte Carlo Methods in Financial Engineering. Springer, New York (2004)
- Heinrich, S.: Multilevel Monte Carlo methods. In: Margenov, S., Wasniewski, J., Yalamov, P. (eds.) Large-Scale Scientific Computing. Berlin, Springer (2001)
- Houssineau, J., Jasra, A., Singh, S.S.: Multilevel transport for smoothing. *SIAM J. Sci. Comp.* **40**, A2315–A2335 (2018)
- Jacob, P.E., Lindsten, F., Schön, T.B.: Coupling of Particle Filters. [arXiv:1606.01156](https://arxiv.org/abs/1606.01156) (2016)
- Jacod, J., Kurtz, T.G., Méléard, S., Protter, P.: The approximate Euler method for Lévy driven stochastic differential equations. *Ann. Inst. Henri Poincaré Probab. Stat.* **41**(3), 523–558 (2005)
- Jasra, A., Doucet, A.: Sequential Monte Carlo methods for diffusion processes. *Proc. R. Soc. A* **465**, 3709–3727 (2009)
- Jasra, A., Del Moral, P.: Sequential Monte Carlo methods for option pricing. *Stoch. Anal. Appl.* **29**, 292–316 (2011)
- Jasra, A., Kamatani, K., Law, K.J.H., Zhou, Y.: Multilevel particle filters. *SIAM J. Numer. Anal.* **55**, 3068–3096 (2017)
- Jasra, A., Kamatani, K., Osei, P.P., Zhou, Y.: Multilevel particle filters: normalizing constant estimation. *Stat. Comput.* **28**, 47–60 (2018)
- Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)
- Madan, D., Seneta, E.: The variance gamma (V.G.) model for share market returns. *J. Bus.* **63**(4), 511–524 (1990)
- Protter, P.: Stochastic Integration and Differential Equations. Stochastic Modelling and Applied Probability, vol. 21, 2nd edn. Springer, Berlin (2004)
- Rubenthaler, S.: Numerical simulation of the solution of a stochastic differential equation driven by a Lévy process. *Stoch. Process. Appl.* **103**(2), 311–349 (2003)
- Rubinstein, R.Y., Kroese, D.P.: Simulation and the Monte Carlo Method, vol. 10. Wiley, New York (2016)
- Sato, K.: Lévy Processes and Infinitely Divisible Distributions. Cambridge studies in Advanced Mathematics, vol. 68. Cambridge University Press, Cambridge (1999)
- Schoutens, W.: Lévy Processes in Finance: Pricing Financial Derivatives. Wiley, Chichester (2003)
- Sen, D., Thiery, A., Jasra, A.: On coupling particle filters. *Stat. Comput.* **28**, 461–475 (2018)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.