COMMUN. MATH. SCI.
Vol. 14, No. 3, pp. 799–829

THE ADAPTIVE PATCHED CUBATURE FILTER AND ITS IMPLEMENTATION*

WONJUNG LEE† AND TERRY LYONS‡

Abstract. There are numerous contexts where one wishes to describe the state of a randomly evolving system. Effective solutions combine models that quantify the underlying uncertainty with available observational data to form scientifically reasonable estimates for the uncertainty in the system state. Stochastic differential equations are often used to mathematically model the underlying system.

The Kusuoka–Lyons–Victoir (KLV) approach is a higher-order particle method for approximating the weak solution of a stochastic differential equation that uses a weighted set of scenarios to approximate the evolving probability distribution to a high-order of accuracy. The algorithm can be performed by integrating along a number of carefully selected bounded variation paths. The iterated application of the KLV method has a tendency for the number of particles to increase. This can be addressed and, together with local dynamic recombination, which simplifies the support of discrete measure without harming the accuracy of the approximation, the KLV method becomes eligible to solve the filtering problem in contexts where one desires to maintain an accurate description of the ever-evolving conditioned measure.

In addition to the alternate application of the KLV method and recombination, we make use of the smooth nature of the likelihood function and high order accuracy of the approximations to lead some of the particles immediately to the next observation time and to build into the algorithm a form of automatic high order adaptive importance sampling.

We perform numerical simulations to evaluate the efficiency and accuracy of the proposed approaches in the example of the linear stochastic differential equation driven by three-dimensional Brownian motion. Our numerical simulations show that, even when the sequential Monte-Carlo methods poorly perform, the KLV method and recombination can together be used to approximate higher-order moments of the filtering solution in a moderate dimension with high accuracy and efficiency.

Key words. Bayesian statistics, particle filter, cubature on Wiener space, recombination.

AMS subject classifications. 60G17, 60G35, 94A12, 94A20.

1. Introduction

Filtering is an approach for calculating the probability distribution of an evolving system in the presence of noisy observations. The problem has many significant and practical applications in science and engineering, such as satellite and airplane orbit determination, the spread of hazardous plumes or pollutants, and prediction of weather and climate in atmosphere-ocean dynamics [1, 15–17, 19–21, 23]. If both the underlying system and the observation process satisfy linear equations, the solution of the filtering problem can be obtained from the Kalman filter [20,21]. For nonlinear filtering problems in finite dimension, there occasionally exist analytic solutions but the results are too narrow in applicability [2]. As a result, a number of numerical schemes have been developed with an aim to accurately describe the fundamental object of interest in filtering, i.e. the conditioned measure, in terms of collection of weighted Dirac masses [12, 15, 17].

When the underlying dynamics is a continuous process and the available observations are intermittent in time, the standard approach of filtering is to perform a forward

*Received: November 2, 2014; accepted (in revised form): July 6, 2015. Communicated by David Anderson.

†Stochastic Analysis Group and Oxford Centre for Collaborative Applied Mathematics (OCCAM), Mathematical Institute and Oxford-Man Institute of Quantitative Finance, University of Oxford, U.K. (leew@maths.ox.ac.uk).

‡Stochastic Analysis Group, Mathematical Institute and Oxford-Man Institute of Quantitative Finance, University of Oxford, U.K. (tlyons@maths.ox.ac.uk).

uncertainty quantification and then to incorporate data into this predicted measure using Bayes' rule in a sequential fashion. The former prediction step corresponds to solving the Kolmogorov forward equation when the system is driven by Brownian motions. For the numerical integration of a stochastic differential equation, the sequential Monte-Carlo method uses sampling from random variables whose distribution agrees with the law of the truncated strong Taylor expansion of the solution of an Ito diffusion. The algorithm usually gives lower-order strong convergence of the probability distribution [22].

Instead of randomly simulating Wiener measure as in the sequential Monte-Carlo method, the KLV (Kusuoka–Lyons–Victoir) method at the path level replaces Brownian motion by a weighted combination of bounded variation paths while making sure that expectations of the iterated integrals with respect to these two measures on Wiener space agree up to a certain degree. Then the particles are pushed forward along the deterministically chosen paths to yield a weighted discrete measure. The KLV method is of higher order with effective and transparent error bounds obtained from the Stratonovich–Taylor expansion of the solution of a stochastic differential equation [31].

It is intrinsic to the KLV method that the number of particles increases when the algorithm is iterated. Therefore its successive application without an efficient suppression of the growth of the number of particles cannot be used to filter the ever-evolving dynamics. Given a family of test functions, one can replace the original discrete measure by a simpler measure with smaller support whose integrations against these test functions agree with those against the original measure. Recombination achieves the reduction of particles in this way using the polynomials as test functions [29]. One advantage of recombination is its local applicability in space. Therefore one can divide the set of particles into a number of disjoint subsets and recombine each subset of discrete measure separately, a process which we call patched recombination. The dynamic property of patched recombination, if an efficient classification method is provided, leads to a competitive high-order reduction algorithm whose error bound can be obtained from the Taylor expansion of the test function.

One can use the alternate application of the KLV method and patched recombination as an algorithm for the prediction step in filtering. However the cost of this non-adaptive method would become extremely high, particularly in high dimension. Therefore we further modify the algorithm so that it can significantly reduce the computational efforts. More precisely, we exploit the internal smoothness of the likelihood to allow some particles to immediately leap to the next observation time provided certain conditions are fulfilled. The bootstrap reweighting is subsequently applied to obtain our non-Monte-Carlo particle approximation of the optimal filter.

The paper is organised as follows. Section 2 introduces the filtering problem and the Bayesian filter as its formal solution. In Section 3, a prototypical sequential Monte-Carlo filtering algorithm and one of its clever variants that adapts importance sampling are described. The rest of the paper is devoted to developing two non-Monte-Carlo particle filtering algorithms that retain the strengths and mitigate the weaknesses of existing Monte-Carlo methods. In order to do that, two essential building blocks, cubature measure on (infinite dimensional) Wiener space and cubature measure on a finite dimensional space, are introduced in sections 4 and 5, respectively. In Section 6, we define the main algorithms, and in Section 7 we perform numerical simulations to validate the algorithms. Concluding discussions are in Section 8.

2. Bayesian filter

Suppose that the N-dimensional underlying Markov process $X(t)$, $t \in \mathbb{R}^+ \cup \{0\}$ and

the N'-dimensional observation process Y_n, $n \in \mathbb{N}$ associated with $X_n = X(nT)$ are given for some inter-observation time $T > 0$. Let $Y_{1:n'} \equiv \{Y_1, \ldots, Y_{n'}\}$ be the path of the observation process and $y_{1:n'} \equiv \{y_1, \cdots, y_{n'}\}$ be a generic point in the space of paths. We define the measure of the conditioned variable $X_n | Y_{1:n'}$ by $\pi_{n|n'}(dx_n) = \mathbb{P}(X_n \in dx_n | Y_{1:n'} = y_{1:n'})$. Assuming the law of $X(0)$ is given, filtering is to find $\pi_{n|n}$ for all $n \geq 1$.

This intermittent data assimilation problem can in principle be solved by the alternate application of the prediction, to obtain the prior measure $\pi_{n|n-1}$ from $\pi_{n-1|n-1}$, and the updating, to obtain the posterior measure $\pi_{n|n}$ from $\pi_{n|n-1}$. If the transition kernel $K(dx_n|x_{n-1})$ and the likelihood function $g(y_n|x_n)$ satisfying

$$\mathbb{P}(X_n \in A | X_{n-1} = x_{n-1}) = \int_A K(dx_n|x_{n-1}),$$
$$\mathbb{P}(Y_n \in B | X_n = x_n) = \int_B g(y_n|x_n)\, dy_n$$

for all $A \in \mathcal{B}(\mathbb{R}^N)$, the Borel σ-algebra, and $B \in \mathcal{B}(\mathbb{R}^{N'})$, are given, the prediction and the updating are achieved by

$$\pi_{n|n-1}(dx_n) = \int K(dx_n|x_{n-1}) \pi_{n-1|n-1}(dx_{n-1}), \tag{2.1}$$

$$\pi_{n|n}(dx_n) = \frac{g(y_n|x_n)\, \pi_{n|n-1}(dx_n)}{\int g(y_n|x_n)\, \pi_{n|n-1}(dx_n)}, \tag{2.2}$$

respectively. Equation (2.2) is Bayes' rule, and the recursive scheme (2.1), (2.2) is called a Bayesian filter.

3. Particle filtering

3.1. Weak approximation. The closed form of $\pi_{n|n'}$ is in general not available. In many cases the essential properties of a probability measure we are interested in can accurately be described by the expectation of test functions. If the class of test functions is specified, we can replace the original measure with a simpler measure that integrates the test functions correctly and hence still keeps the right properties of the original measure. Therefore efforts have been devoted to weakly approximating $\pi_{n|n'}$ by finding an efficient way to compute $\mathbb{E}(f(X_n)|Y_{1:n'}) = \int f(x_n) \pi_{n|n'}(dx_n)$ accurately for a sufficiently large class of $f : \mathbb{R}^N \to \mathbb{R}$. We mention that the class of test functions is not given in the filtering problem. However their choice is quite critical as it affects the notion of an optimal algorithm and controls the detailed description of the conditioned measure.

One of the methodologies for the weak approximation is to employ *particles* whose locations and weights characterise the approximation of the conditioned measure. More precisely, a particle filter is a recursive algorithm that produces

$$\pi^{\mathrm{PF}}_{n|n'} = \sum_{i=1}^{M_{n|n'}} \lambda^i_{n|n'} \delta_{x^i_{n|n'}} \tag{3.1}$$

approximating $\pi_{n|n'}$, where δ_x denotes the Dirac mass centred at x. One approximates $(\pi_{n|n'}, f)$ by $(\pi^{\mathrm{PF}}_{n|n'}, f) = \sum_{i=1}^{M_{n|n'}} \lambda^i_{n|n'} f(x^i_{n|n'})$, where the notation $(\pi, f) = \int f(x) \pi(dx)$ is used.

3.2. Sequential Monte-Carlo methods. Particle approximation is widely used in Monte-Carlo frameworks. We here introduce two representative algorithms, the sampling importance resampling (SIR) suggested in [17] and the sequential importance sampling and resampling (SISR) algorithm [13, 30, 35]. The number of particles does not have to be equal in each step, but it is here fixed by $M_{n|n'} = M$ for simplicity [10].

3.2.1. Sampling importance resampling (SIR). The prediction step is achieved by using $(\pi_{n|n-1}, f) = (\pi_{n-1|n-1}, Kf)$ from equation (2.1). Given the empirical measure $\pi^{\mathrm{SIR}}_{n-1|n-1} = \frac{1}{M}\sum_{i=1}^{M} \delta_{x^i_{n-1|n-1}}$ approximating $\pi_{n-1|n-1}$, one performs independent and identically distributed (i.i.d.) sampling $\bar{x}^i_{n|n-1}$ drawn from $K(dx_n|x^i_{n-1|n-1})$. Then $\pi^{\mathrm{SIR}}_{n|n-1} = \frac{1}{M}\sum_{i=1}^{M} \delta_{\bar{x}^i_{n|n-1}}$ is an empirical measure with respect to $\pi_{n|n-1}$.

For the updating step, equation (2.2) implies

$$(\pi_{n|n}, f) = (\pi_{n|n-1}, fg^{y_n})/(\pi_{n|n-1}, g^{y_n}),$$

where the notation $g^{y_n}(\cdot) \equiv g(y_n|\cdot)$ is used. Define the bootstrap reweighting operator

$$\mathrm{REW}\left(\sum_{i=1}^{n} \kappa_i \delta_{x^i}, g^{y_n}\right) \equiv \frac{\sum_{i=1}^{n} \kappa_i g^{y_n}(x^i)\delta_{x^i}}{\sum_{i=1}^{n} \kappa_i g^{y_n}(x^i)} \tag{3.2}$$

and $\bar{\pi}^{\mathrm{SIR}}_{n|n} = \mathrm{REW}\left(\pi^{\mathrm{SIR}}_{n|n-1}, g^{y_n}\right)$ is an approximation of $\pi_{n|n}$.

In order to prevent degeneracy in the weights caused by a successive application of equation (3.2), one approximates the weighted discrete measure $\bar{\pi}^{\mathrm{SIR}}_{n|n}$ by an equally weighted discrete measure [12]. Random resampling performs M independent samples $\{x^i_{n|n}\}_{i=1}^{M}$ from $\bar{\pi}^{\mathrm{SIR}}_{n|n}$. This process might introduce a large Monte-Carlo variation and work has been done to reduce the variance [4, 7]. The resulting one $\pi^{\mathrm{SIR}}_{n|n} = \frac{1}{M}\sum_{i=1}^{M} \delta_{x^i_{n|n}}$ is an empirical measure with respect to $\pi_{n|n}$.

The SIR algorithm can be displayed by

$$\pi^{\mathrm{SIR}}_{n-1|n-1} \mapsto \pi^{\mathrm{SIR}}_{n|n-1} \Rightarrow \bar{\pi}^{\mathrm{SIR}}_{n|n} \to \pi^{\mathrm{SIR}}_{n|n} \tag{3.3}$$

where the notation $\mapsto$ is used for moving particles forward in time, $\Rightarrow$ for reweighting and $\to$ for random resampling. The algorithm is very intuitive and straightforward to implement. Furthermore, it produces an approximation that converges toward to the truth posterior measure as the number of particles increases [5]. However, SIR might be inaccurate when $\pi^{\mathrm{SIR}}_{n|n-1}$ is far from $\pi_{n|n}$ in the sense that bootstrap reweighting generates importance weights distributed with a high variance. The following algorithm modifies SIR to get over this degeneracy problem to some extent.

3.2.2. Sequential importance sampling and resampling (SISR). Given the unweighted measure $\pi^{\mathrm{SISR}}_{n-1|n-1} = \frac{1}{M}\sum_{i=1}^{M} \delta_{x^i_{n-1|n-1}}$ that approximates $\pi_{n-1|n-1}$, one performs i.i.d. sampling $\widetilde{x}^i_{n|n-1} \sim \widetilde{K}(dx_n|x^i_{n-1|n-1}, y_n)$ instead of $\bar{x}^i_{n|n-1} \sim K(dx_n|x^i_{n-1|n-1})$. Here the new transition kernel $\widetilde{K}$ can depend on the instance y_n and should be chosen in such a way that the distribution of $\pi^{\mathrm{SISR}}_{n|n-1} = \frac{1}{M}\sum_{i=1}^{M} \delta_{\widetilde{x}^i_{n|n-1}}$ is closer to $\pi_{n|n}$ than $\pi^{\mathrm{SIR}}_{n|n-1}$ in the above-mentioned sense [13].

Note that $\pi^{\mathrm{SISR}}_{n|n-1}$ is not distributed according to $\pi_{n|n-1}$. To account for the effect

of this discrepancy, the expression

$$\begin{aligned}&\mathbb{P}(X_{n-1}\in dx_{n-1},X_n\in dx_n|Y_{1:n}=y_{1:n})\\&=\frac{w(x_{n-1},x_n,y_n)\widetilde{K}(dx_n|x_{n-1},y_n)\pi_{n-1|n-1}(dx_{n-1})}{\int w(x_{n-1},x_n,y_n)\widetilde{K}(dx_n|x_{n-1},y_n)\pi_{n-1|n-1}(dx_{n-1})},\end{aligned}\tag{3.4}$$

where

$$w(x_{n-1},x_n,y_n)\propto\frac{g(y_n|x_n)K(dx_n|x_{n-1})}{\widetilde{K}(dx_n|x_{n-1},y_n)}$$

is used. Replacing $\widetilde{K}(dx_n|x_{n-1},y_n)\pi_{n-1|n-1}(dx_{n-1})$ in equation (3.4) by its empirical approximation and integrating over x_{n-1}, one obtains $\widetilde{\pi}^{\mathrm{SISR}}_{n|n}=\sum_{i=1}^M w^i\delta_{\widetilde{x}^i_{n|n-1}}$ where $w^i\propto w(x^i_{n-1|n-1},\widetilde{x}^i_{n|n-1},y_n)$. A random resampling from $\widetilde{\pi}^{\mathrm{SISR}}_{n|n}$ yields the empirical measure with respect to $\pi_{n|n}$, denoted by $\pi^{\mathrm{SISR}}_{n|n}$.

If $\widetilde{K}(dx_n|x_{n-1},y_n)$ and $w(x_{n-1},x_n,y_n)$ have better theoretical properties than $K(dx_n|x_{n-1})$ and $g(y_n|x_n)$ such as better mixing properties of $\widetilde{K}(dx_n|x_{n-1},y_n)$ or flatter likelihood, then the algorithm can produce a better approximation. Because one needs to integrate an evolution equation of a Markov process with transition kernel $\widetilde{K}$ in any practical implementation, designing efficient particle filtering methods is equivalent to building an appropriate dynamic model that has good theoretical properties while keeping the same filtering distributions. The SISR algorithm

$$\pi^{\mathrm{SISR}}_{n-1|n-1}\mapsto\pi^{\mathrm{SISR}}_{n|n-1}\Rightarrow\widetilde{\pi}^{\mathrm{SISR}}_{n|n}\to\pi^{\mathrm{SISR}}_{n|n}\tag{3.5}$$

might use fewer particles than SIR to achieve a similar accuracy [40]. One can find a considerable study illustrating the difference in performance of SISR using different proposal distributions in [11].

4. Kusuoka–Lyons–Victoir (KLV) method

Suppose that a random vector $X(t)\in\mathbb{R}^N$ evolves according to a Stratonovich stochastic differential equation (SDE)

$$dX(t)=V_0(X(t))\,dt+\sum_{i=1}^d V_i(X(t))\circ dW_i(t),\tag{4.1}$$

where $\{V_i\in C_b^\infty(\mathbb{R}^N,\mathbb{R}^N)\}_{i=0}^d$ is a family of smooth vector fields from $\mathbb{R}^N$ to $\mathbb{R}^N$ with bounded derivatives of all orders, and $W=(W_1,\cdots,W_d)$ denotes a set of Brownian motions, independent of one another. The KLV method enables us to deterministically approximate the law of $X(T)$ in terms of discrete measure.

4.1. Cubature on Wiener space on path level. Let us use the notations $W_0(t)=t$, $\omega_{T,0}(t)=t$ and $I=(i_1,\ldots,i_l)\in\{0,\ldots,d\}^l$. Consider the iterated integral with respect to $W=(W_1,\ldots,W_d)$,

$$\mathcal{J}^I_{0,T}(\circ W)\equiv\int_{0<t_1<\cdots<t_l<T}\circ dW_{i_1}(t_1)\cdots\circ dW_{i_l}(t_l),$$

and the iterated integral with respect to a continuous path of bounded variation $\omega_T=(\omega_{T,1},\ldots,\omega_{T,d}):[0,T]\to\mathbb{R}^d$,

$$\mathcal{J}^I_{0,T}(\omega_T)\equiv\int_{0<t_1<\cdots<t_l<T}d\omega_{T,i_1}(t_1)\cdots d\omega_{T,i_l}(t_l).$$

Recall that Wiener space $C_0^0\left([0,T],\mathbb{R}^d\right)$ is the set of continuous functions starting at zero. We define a discrete measure $\mathbb{Q}_T^m=\sum_{j=1}^{n_m}\lambda_j\delta_{\omega_T^j}$ supported on continuous paths of bounded variation to be a cubature on Wiener space on path level of degree m with respect to the Wiener measure $\mathbb{P}$, provided the equation

$$\begin{aligned}\mathbb{E}_{\mathbb{P}}\left(\mathcal{J}_{0,T}^I(\circ W)\right)&=\mathbb{E}_{\mathbb{Q}_T^m}\left(\mathcal{J}_{0,T}^I(\circ W)\right)\\&=\sum_{j=1}^{n_m}\lambda_j\mathcal{J}_{0,T}^I(\omega_T^j)\end{aligned}\tag{4.2}$$

holds for all I satisfying $||I||\equiv l+\mathrm{card}\{j,i_j=0\}\le m$. Note that $\mathbb{Q}_T^m$ is obtained from $\mathbb{Q}_1^m$ via a suitable rescaling and that the existence of $\mathbb{Q}_1^m$ with $n_m\le\mathrm{card}\{I:\|I\|\le m\}$ is proved in [31].

The cubature measure on Wiener space can be used to approximate $\mathbb{E}_{\mathbb{P}}(f(X_T^x))$ for the random process X_t^x in N dimension satisfying

$$dX_t^x=V_0(X_t^x)\,dt+\sum_{i=1}^{d}V_i(X_t^x)\circ dW_i(t)\tag{4.3}$$

and $X_0^x=x$. The expectation of $f(X_T^x)$ against Wiener measure can be viewed as an integral with respect to infinite dimensional Wiener space.

Let $t\mapsto X_t^{x,\omega_\Delta^j}$ for $t\in[0,\Delta]$ be the deterministic process satisfying

$$dX_t^{x,\omega_\Delta^j}=\sum_{i=0}^{d}V_i(X_t^{x,\omega_\Delta^j})\,d\omega_{\Delta,i}^j(t)\tag{4.4}$$

and $X_0^{x,\omega_\Delta^j}=x$. The ordinary differential equations (ODEs) of equation (4.4) are obtained from replacing the Brownian motions W in equation (4.3) by the bounded variation path ω_Δ^j. The measure $\sum_{j=1}^{n_m}\lambda_j\delta_{X_T^{x,\omega_T^j}}$ on $\mathbb{R}^N$ is called the cubature approximation of the law of X_T^x at the path level.

An error estimate for the weak approximation of this particle method can be derived from the Stratonovich–Taylor expansion of a smooth function f,

$$f(X_T^x)=\sum_{||I||\le m}V_If(x)\mathcal{J}_{0,T}^I(\circ W)+R_m(x,T,f),\tag{4.5}$$

where the remainder $R_m(x,T,f)$ satisfies

$$\sup_{x\in\mathbb{R}^N}\sqrt{\mathbb{E}_{\mathbb{P}}(R_m(x,T,f)^2)}\le C\sum_{i=m+1}^{m+2}T^{i/2}\sup_{\|I\|=i}\|\,V_If\,\|_\infty\tag{4.6}$$

for a constant C depending on d and m [22]. Here the vector field $V_i=(V_{i,1},\ldots,V_{i,N})$ is used as the differential operator $V_i\equiv\sum_{j=1}^{N}V_{i,j}\partial x_j$ and V_I denotes $V_{i_1}\cdots V_{i_k}$.

The process $R_m(x,T,f)$ further satisfies

$$\sup_{x\in\mathbb{R}^N}\mathbb{E}_{\mathbb{Q}_T^m}(|R_m(x,T,f)|)\le C\sum_{i=m+1}^{m+2}T^{i/2}\sup_{\|I\|=i}\|\,V_If\,\|_\infty\tag{4.7}$$

for a constant C depending on d, m, and $\mathbb{Q}_1^m$ [31]. Let the operators P_T and Q_T^m be defined by $P_T f(x) \equiv \mathbb{E}_{\mathbb{P}}(f(X_T^x))$ and $Q_T^m f(x) \equiv \mathbb{E}_{\mathbb{Q}_T^m}(f(X_T^x))$. Then the error bound of the cubature approximation at the path level is given by

$$\sup_{x\in\mathbb{R}^N}\left|\mathbb{E}_{\mathbb{P}}(f(X_T^x)) - \sum_{j=1}^{n_m}\lambda_j f(X_T^{x,\omega_T^j})\right| = \| (P_T - Q_T^m) f \|_\infty$$
$$\leq C \sum_{i=m+1}^{m+2} T^{i/2} \sup_{\|I\|=i} \| V_I f \|_\infty \tag{4.8}$$

for smooth f, from equation (4.2) and equations (4.5), (4.6), and (4.7).

The algorithm was developed by Lyons, and Victoir [31] following the work of Kusuoka [24, 26], so it is referred to as the KLV method. Equation (4.8) leads us to define

$$\mathrm{KLV}^{(m)}\left(\sum_{i=1}^{n}\kappa_i\delta_{x^i}, \Delta\right) \equiv \sum_{i=1}^{n}\sum_{j=1}^{n_m}\kappa_i\lambda_j\delta_{X_\Delta^{x^i,\omega_\Delta^j}} \tag{4.9}$$

that may be interpreted as a Markov operator acting on discrete measure on $\mathbb{R}^N$.

In the following, assume $T \in (0,1)$ for simplicity. One may take a higher degree m to achieve a given degree of accuracy in equation (4.8). An alternative method to improve the accuracy of the particle approximation is a successive application of the KLV operator. Let $\mathcal{D} = \{0 = t_0 < t_1 < \cdots < t_k = T\}$ be a partition of $[0,T]$ with $s_j = t_j - t_{j-1}$. Instead of $Q_T^m f(x) = (\mathrm{KLV}^{(m)}(\delta_x, T), f)$, the value of $P_T f(x) = P_{s_1} P_{s_2} \cdots P_{s_k} f(x)$ can accurately be approximated by a multiple-step algorithm $Q_{s_1}^m Q_{s_2}^m \cdots Q_{s_k}^m f(x)$.

Given a discrete measure μ^0, we define a sequence of discrete measure by

$$\begin{aligned}&\Phi_{\mathcal{D}}^{m,0}(\mu^0) = \mu^0,\\ &\Phi_{\mathcal{D}}^{m,j}(\mu^0) = \mathrm{KLV}^{(m)}(\Phi_{\mathcal{D}}^{m,j-1}(\mu^0), s_j) \quad 1 \leq j \leq k\end{aligned} \tag{4.10}$$

that can be viewed as a Markov chain. The inequality

$$\begin{aligned}\left|P_T f(x) - (\Phi_{\mathcal{D}}^{m,k}(\delta_x), f)\right| &= \left|\sum_{j=1}^{k}\left(\Phi_{\mathcal{D}}^{m,j-1}(\delta_x), P_{T-t_{j-1}} f\right) - \left(\Phi_{\mathcal{D}}^{m,j}(\delta_x), P_{T-t_j} f\right)\right|\\ &= \left|\sum_{j=1}^{k}\left(\Phi_{\mathcal{D}}^{m,j-1}(\delta_x), (P_{s_j} - Q_{s_j}^m) P_{T-t_j} f\right)\right|\\ &\leq \sum_{j=1}^{k} \| (P_{s_j} - Q_{s_j}^m) P_{T-t_j} f \|_\infty\end{aligned} \tag{4.11}$$

obtained from the Markovian property of the KLV operator shows that the total error of the repeated KLV application is bounded above by the sum of the errors over the subintervals in the partition. Applying equation (4.8) to estimate the upper bound of equation (4.11), we need $P_{T-t_j} f$ to be smooth. When f is smooth, this is true and the error bound

$$\sup_{x\in\mathbb{R}^N}\left|P_T f(x) - (\Phi_{\mathcal{D}}^{m,k}(\delta_x), f)\right| \leq C \sum_{i=m+1}^{m+2}\sum_{j=1}^{k} s_j^{i/2} \sup_{\|I\|=i} \| V_I P_{T-t_j} f \|_\infty$$

is obtained from equations (4.8), and (4.11).

The case of Lipschitz continuous f is of particular interest because $P_t f$ is indeed smooth in the direction of $\{V_i\}_{i=0}^d$ with additional conditions for these vector fields [6]. In this case, the regularity estimate

$$\| V_I P_t f \|_\infty \leq \frac{C}{t^{(\|I\|-1)/2}} \| \nabla f \|_\infty \tag{4.12}$$

holds for all $t \in (0,1]$, where C is a constant independent of f [25, 27]. Combining equations (4.8), (4.11), and equation (4.12), we obtain an error estimate for the KLV method in terms of the gradient of the Lipschitz continuous f:

$$\sup_{x\in\mathbb{R}^N} \left| P_T f(x) - (\Phi_{\mathcal{D}}^{m,k}(\delta_x), f) \right| \leq C \| \nabla f \|_\infty \left(s_k^{1/2} + \sum_{i=m+1}^{m+2} \sum_{j=1}^{k-1} \frac{s_j^{i/2}}{(T-t_j)^{(i-1)/2}} \right), \tag{4.13}$$

where C is a constant independent of k. Here the final term in the upper bound of equation (4.11) is estimated by $\| (P_{s_k} - Q_{s_k}^m) f \|_\infty \leq \| P_{s_k} f - f \|_\infty + \| f - Q_{s_k}^m f \|_\infty \leq C s_k^{1/2} \| \nabla f \|_\infty$ using the boundedness of $\{V_i\}_{i=0}^d$.

Let $\mathcal{D}(\gamma) = \{t_j\}_{j=0}^k$ be the Kusuoka partition [24] given by

$$t_j = T\left(1 - \left(1 - \frac{j}{k}\right)^\gamma\right). \tag{4.14}$$

Then the error estimate

$$\sup_{x\in\mathbb{R}^N} \left| P_T f(x) - (\Phi_{\mathcal{D}(\gamma)}^{m,k}(\delta_x), f) \right| \leq C \| \nabla f \|_\infty T^{1/2} k^{-(m-1)/2} \tag{4.15}$$

is satisfied for a Lipschitz continuous f when $\gamma > m-1$.

Equation (4.15) is obtained from substituting the non-equidistant time discretisation $\mathcal{D}(\gamma)$ into equation (4.13). Using this particular choice of partition ensures that the bound of the KLV error is of high order in the number of iterations k.

Before concluding this subsection, we here mention that $u(x,t) \equiv \mathbb{E}_{\mathbb{P}}(f(X_{T-t}^x))$ satisfies the partial differential equation (PDE)

$$\begin{aligned} \frac{\partial}{\partial t} u(x,t) &= -\left(V_0 + \frac{1}{2}\sum_{i=1}^d V_i^2\right) u(x,t), \\ u(x,T) &= f(x). \end{aligned} \tag{4.16}$$

where $\{V_i\}_{i=0}^d$ are used as differential operators [41]. Therefore $P_T f(x)$, the heat kernel applied to f, is equal to the solution $u(x,0)$ of equation (4.16). Due to this inherent relationship between parabolic PDEs and SDEs, one can apply any well-known algorithm for the solution of equation (4.16) to the prediction step of the filtering problem where the underlying system is given by equation (4.1). However it is very important to understand the critical difference between these two problems. One needs to weakly approximate the law of $X(T)$, when $X(0)$ is given by δ_x, that accurately integrate the test function f for the PDE problem while the filtering problem requires one to approximate the conditioned measure of $X_n | Y_{1:n}$ for all $n \geq 1$, in which the test function is not at all specified.

4.2. Cubature on Wiener space on flow level. We here study the construction of cubature formula $\mathbb{Q}_T^m$. Meanwhile cubature on Wiener space on flow level is defined in terms of Lie polynomial and used to develop an approximation based on the autonomous ODEs at flow level.

Let $\{e_i\}_{i=0}^d$ be the standard basis of $\mathbb{R}\oplus\mathbb{R}^d$. Let $\mathcal{T}$ denote the associative and noncommutative tensor algebra of polynomials generated by $\{e_i\}_{i=0}^d$. The exponential and logarithm on $\mathcal{T}$ are defined by

$$\begin{aligned}\exp(a) &\equiv \sum_{i=0}^{\infty} \frac{a^{\otimes i}}{i!},\\ \log(a) &\equiv \log(a_0) + \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i} \left(\frac{a}{a_0} - 1\right)^{\otimes i},\end{aligned} \tag{4.17}$$

where $a = \sum_I a_I e_I$ and $e_I = e_{i_1} \otimes \cdots \otimes e_{i_l}$ for a multi-index $I = (i_1, \ldots, i_l) \in \{0, \ldots, d\}^l$. Here $\otimes$ denotes the tensor product. Let the operators $\exp^{(m)}(\cdot)$ and $\log^{(m)}(\cdot)$ be defined by the truncation of equation (4.17) leaving the case $\| I \| \le m$.

The signature of a continuous path of bounded variation $\omega_T : [0,T] \to \mathbb{R}^d$ is defined by

$$\begin{aligned}\mathcal{S}_{0,T}(\omega_T) &\equiv \sum_{l=0}^{\infty} \int_{0<t_1<\cdots<t_l<T} d\omega_T(t_1) \otimes \cdots \otimes d\omega_T(t_l)\\ &= \sum_I \mathcal{J}_{0,T}^I(\omega_T)\, e_I\end{aligned}$$

and similarly the signature of a Brownian motion W by

$$\mathcal{S}_{0,T}(\circ W) \equiv \sum_I \mathcal{J}_{0,T}^I(\circ W)\, e_I.$$

In view of equation (4.2), the definition of cubature on Wiener space of degree m can be rephrased by

$$\mathbb{E}_{\mathbb{P}}\left(\mathcal{S}_{0,T}^{(m)}(\circ W)\right) = \mathbb{E}_{\mathbb{Q}_T^m}\left(\mathcal{S}_{0,T}^{(m)}(\circ W)\right), \tag{4.18}$$

where $\mathcal{S}_{0,T}^{(m)}(\cdot)$ is the truncation of $\mathcal{S}_{0,T}(\cdot)$ to the case $\| I \| \le m$.

Define $\mathcal{L}$ to be the space of Lie polynomials, i.e. linear combinations of finite sequences of Lie brackets $[e_i, e_j] = e_i \otimes e_j - e_j \otimes e_i$. Because Chen's theorem ensures that the logarithm of signature is a Lie series [37], its truncation

$$\mathcal{L}_T^j \equiv \log^{(m)}(\mathcal{S}_{0,T}(\omega_T^j)) \tag{4.19}$$

is a Lie polynomial and an element of $\mathcal{L}$. Then the measure $\widetilde{\mathbb{Q}}_T^m = \sum_{j=1}^{n_m} \lambda_j \delta_{\mathcal{L}_T^j}$ supported on Lie polynomials satisfies

$$\begin{aligned}\mathbb{E}_{\mathbb{P}}\left(S_{0,T}^{(m)}(\circ W)\right) &= \mathbb{E}_{\widetilde{\mathbb{Q}}_T^m}\left(\exp^{(m)}(\mathcal{L})\right)\\ &= \sum_{j=1}^{n_m} \lambda_j \exp^{(m)}(\mathcal{L}_T^j).\end{aligned} \tag{4.20}$$

Conversely, for any Lie polynomials $\mathcal{L}_T^j$, there exist continuous bounded variation paths ω_T^j whose truncated logarithmic signature is $\mathcal{L}_T^j$. Moreover if $\widetilde{\mathbb{Q}}_T^m$ satisfies equation (4.20), then $\mathbb{Q}_T^m$ satisfies equation (4.18). Therefore equation (4.18) and equation (4.20) are equivalent. The discrete measure $\widetilde{\mathbb{Q}}_T^m$ is defined as cubature on Wiener space on flow level.

The expectation of the truncated Brownian signature is

$$\mathbb{E}_{\mathbb{P}}\left(\mathcal{S}_{0,1}^{(m)}(\circ W)\right) = \exp^{(m)}\left(e_0 + \frac{1}{2}\sum_{i=1}^{d} e_i \otimes e_i\right), \tag{4.21}$$

which is proved in [31]. It is immediate from equation (4.21) that cubature formulae on Wiener space for $m = 2n-1$ and $m = 2n$ are equal to one another. A formula $\{\lambda_j, \mathcal{L}_1^j\}_{j=1}^{n_m}$ satisfying equation (4.20) is found when $m = 3$ and $m = 5$ for any d [31]. In some cases of $m \geq 7$, cubature formula of Lie polynomial is available when $d = 1, 2$ [18].

From this $\widetilde{\mathbb{Q}}_1^m$ and equation (4.19), one can construct $\mathbb{Q}_1^m$. It follows from the scaling property of the Brownian motion that $\omega_{T,0}^j(t) = \omega_{1,0}^j(t)$ and $\omega_{T,i}^j(t) = \sqrt{T}\omega_{1,i}^j(t/T)$ for $1 \leq i \leq d$. The paths define a cubature formula $\widetilde{\mathbb{Q}}_T^m$. Using $\mathcal{J}_{0,T}^I(\circ W) \triangleq T^{\|I\|/2}\mathcal{J}_{0,1}^I(\circ W)$ and equation (4.19), the scaling of the Lie polynomial is $\mathcal{L}_T^j = \langle T, \mathcal{L}_1^j \rangle$ where $\langle t, \sum_I a_I e_I \rangle \equiv \sum_I t^{\|I\|/2} a_I e_I$. The Lie polynomials define a cubature formula $\mathbb{Q}_T^m$.

Next, we study the approximation based on the flows of autonomous ODEs. It is in fact corresponds to a version of Kusuoka's algorithm [24]. Let Γ denote the algebra homomorphism generated by $\Gamma(e_i) = V_i$ for $i = 0, \ldots, d$. For a vector field $V \in C_b^\infty(\mathbb{R}^N, \mathbb{R}^N)$, we define the flow $\mathrm{Exp}(tV)(x) \equiv \xi_t^x$ to be the solution of the ODE $d\xi_t^x = V(\xi_t^x)dt$ with $\xi_0^x = x$. By interchanging the algebra homomorphism Γ with the exponentiation (so far taken in the tensor algebra), we arrive at an approximation operator in which the exponentiation is understood as taking the flow of autonomous ODEs. More precisely, one has

$$\begin{aligned}\mathbb{E}_{\mathbb{P}}\left(\Gamma\left(S_{0,T}^{(m)}(\circ W)\right)\right) f(x) &= \sum_{j=1}^{n_m} \lambda_j \Gamma\left(\exp^{(m)}(\mathcal{L}_T^j)\right) f(x) \\ &\simeq \sum_{j=1}^{n_m} \lambda_j f\left(\mathrm{Exp}\left(\Gamma(\mathcal{L}_T^j)\right)(x)\right)\end{aligned}$$

using equation (4.20). The error introduced while interchanging exp and Γ operators turns out to be of the similar order with the error in the cubature approximation of the path level, as shown below.

Formally the cubature approximation at the flow level is defined as follows. Let $t \mapsto X_t^{x,\mathcal{L}_\Delta^j}$ for $t \in [0,1]$ be the deterministic process satisfying

$$dX_t^{x,\mathcal{L}_\Delta^j} = \Gamma(\mathcal{L}_\Delta^j)(X_t^{x,\mathcal{L}_\Delta^j})dt \tag{4.22}$$

and $X_0^{x,\mathcal{L}_\Delta^j} = x$. Define the operator

$$\widetilde{\mathrm{KLV}}^{(m)}\left(\sum_{i=1}^{n} \kappa_i \delta_{x^i}, \Delta\right) \equiv \sum_{i=1}^{n}\sum_{j=1}^{n_m} \kappa_i \lambda_j \delta_{X_1^{x^i,\mathcal{L}_\Delta^j}} \tag{4.23}$$

and a sequence of discrete measure

$$\begin{aligned}\widetilde{\Phi}_{\mathcal{D}}^{m,0}(\mu^0) &= \mu^0, \\ \widetilde{\Phi}_{\mathcal{D}}^{m,j}(\mu^0) &= \widetilde{\mathrm{KLV}}^{(m)}(\widetilde{\Phi}_{\mathcal{D}}^{m,j-1}(\mu^0), s_j)\end{aligned}$$

for $1 \le j \le k$.

Let $\widetilde{Q}_T^m f(x) \equiv (\widetilde{\mathrm{KLV}}^{(m)}(\delta_x, T), f)$ be a flow level cubature approximation. Then the Taylor expansions of equation (4.4) and equation (4.22) lead to

$$\| (Q_T^m - \widetilde{Q}_T^m) f \|_\infty \le C \sum_{m+1 \le \|I\| \le 2m} T^{\|I\|/2} \| V_I f \|_\infty \tag{4.24}$$

for a smooth f, where C is a constant depending on m, d, $\mathbb{Q}_1^m$, and $\widetilde{\mathbb{Q}}_1^m$ [24].

The error estimate

$$\sup_{x \in \mathbb{R}^N} \left| P_T f(x) - (\widetilde{\Phi}^{m,k}_{\mathcal{D}(\gamma)}(\delta_x), f) \right| \le C \| \nabla f \|_\infty T^{1/2} k^{-(m-1)/2} \tag{4.25}$$

is satisfied for a Lipschitz continuous f when $\gamma > m-1$.

Equation (4.25) is obtained using equation (4.24) and demonstrates that for a suitable partition the bounds for the approximation at flow and path level have the same rate of convergence in view of equation (4.15).

5. Simplification of particle approximation

A successive application of the KLV operator gives rise to geometric growth of the number of particles in view of equations (4.9) and (4.23). Except some cases of PDE problems in which the KLV method can produce an accurate approximation with small number of iterations, this geometric growth of particle number prohibits an application of the KLV method, particularly to the filtering problem where, to maintain an accurate description of the ever-evolving measure with reasonable computational, cost is the key requirement. It is therefore necessary to add a simplification algorithm between two consecutive iterations, which suppresses the growth of the number of particles in the KLV framework. Though it is possible to achieve the simplification through one of several Monte-Carlo methods, we here make use of cubature measure on a finite-dimensional space to efficiently reduce the support of discrete measure. This will let the entire algorithm consistently step outside of the Monte-Carlo paradigm. Furthermore, its proper applications never harm the accuracy of the KLV approximation, as we shall see.

5.1. Cubature on a finite-dimensional space. Let ν be a (possibly unnormalised) positive measure on $\mathbb{R}^N$. A discrete measure $\widehat{\nu}^{(r)} = \sum_{j=1}^{n_r} w_j \delta_{y^j}$ is called a cubature (quadrature when $N=1$) of degree r with respect to ν provided $\mathrm{supp}(\widehat{\nu}^{(r)}) \subseteq \mathrm{supp}(\nu)$ and (ν, q) equals $(\widehat{\nu}^{(r)}, q) = \sum_{j=1}^{n_r} w_j q(y^j)$ for all polynomials q whose total degree is less than or equal to r. It is proved that a cubature $\widehat{\nu}^{(r)}$ with respect to an arbitrary positive measure ν satisfying $n_r \le \binom{N+r}{r}$ exists [36]. As a result, one can adopt a cubature measure on $\mathbb{R}^N$ with respect to the original measure as the reduced measure.

Importantly an error bound of $(\nu, F) - (\widehat{\nu}^{(r)}, F) \equiv (\nu - \widehat{\nu}^{(r)}, F)$ for a smooth function $F : \mathbb{R}^N \to \mathbb{R}$ can be obtained from the Taylor expansion. The value of F at $x = (x_1, \ldots, x_N)$ is written as

$$F(x) = \sum_{|\alpha| \le r} \frac{D^\alpha F(x_0)}{\alpha!} (x - x_0)^\alpha + R^r(x, x_0, F) \tag{5.1}$$

where $\alpha \equiv (\alpha_1, \cdots, \alpha_N)$, $|\alpha| \equiv \alpha_1 + \cdots + \alpha_N$, $\alpha! \equiv \alpha_1! \cdots \alpha_N!$, $D^\alpha \equiv \partial x_1^{\alpha_1} \cdots \partial x_N^{\alpha_N}$, $x^\alpha \equiv$

$x_1^{\alpha_1}\cdots x_N^{\alpha_N}$ and

$$R^r(x,x_0,F)=\sum_{|\alpha|=r+1}\frac{D^\alpha F(x^*)}{\alpha!}(x-x_0)^\alpha \tag{5.2}$$

for some $x^*\in\mathbb{R}^N$. If the support of ν is in a closed ball of centre x_0 and radius u, denoted by $B(x_0,u)$, then we have

$$\begin{aligned}|(\nu-\widehat{\nu}^{(r)},F)|=|(\nu-\widehat{\nu}^{(r)},R^r)|&\le 2(\nu,1)\,\|\,R^r\,\|_{L_\infty(B(x_0,u))}\\&\le\frac{Cu^{r+1}}{(r+1)!}\sup_{|\alpha|=r+1}\|\,D^\alpha F\,\|_{L_\infty(B(x_0,u))}.\end{aligned} \tag{5.3}$$

Equation (5.3) reveals that cubature on a finite-dimensional space is an approach for the numerical integration of functions on finite-dimensional space with a clear error bound.

5.2. Local dynamic recombination. Instead of using a cubature of higher degree to reduce the entire family of particles all at once, we improve the performance by dividing a given discrete measure into locally supported unnormalised positive measures and replacing each separated measure by the cubature of lower degree [29]. This so-called local dynamic recombination can be a competitive algorithm because each reduction can be performed in a parallel manner to save computational time and because the error bound from the Taylor approximation remains of higher order.

Let $U=(U_i)_{i=1}^R$ be a collection of balls of radius u that covers the support of discrete measure μ on $\mathbb{R}^N$. Then one can find unnormalised measures $(\mu_i)_{i=1}^R$ such that $\mu=\bigsqcup_{i=1}^R\mu_i$ (μ_i and μ_j have disjoint support for $i\neq j$) and $\operatorname{supp}(\mu_i)\subseteq U_i\cap\operatorname{supp}(\mu)$. In this case, we define the patched recombination operator by

$$\mathrm{REC}^{(u,r)}(\mu)\equiv\bigsqcup_{i=1}^R\widehat{\mu}_i^{(r)}, \tag{5.4}$$

where $\widehat{\mu}_i^{(r)}$ denotes a cubature of degree r with respect to μ_i.

Given a discrete measure μ^0, we define a sequence of discrete measure by

$$\begin{aligned}&\Phi^{m,0}_{\mathcal{D},(u,r)}(\mu^0)=\mu^0,\\&\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\mu^0)=\mathrm{REC}^{(u_{j-1},r_{j-1})}\left(\Phi^{m,j-1}_{\mathcal{D},(u,r)}(\mu^0)\right),\\&\Phi^{m,j}_{\mathcal{D},(u,r)}(\mu^0)=\mathrm{KLV}^{(m)}\left(\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\mu^0),s_j\right),\end{aligned} \tag{5.5}$$

for $1\le j\le k$. An application of equation (5.5) with initial condition δ_x yields a weak approximation for the law of X_T^x. One obtains the estimate

$$\begin{aligned}&\left|P_Tf(x)-(\Phi^{m,k}_{\mathcal{D},(u,r)}(\delta_x),f)\right|\\=&\left|\sum_{j=1}^k\left(\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_{j-1}}f\right)-\left(\Phi^{m,j}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_j}f\right)\right.\\&\left.+\left(\Phi^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_{j-1}}f\right)-\left(\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_{j-1}}f\right)\right|\end{aligned}$$

$$=\left|\sum_{j=1}^{k}\left(\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x),(P_{s_j}-Q^m_{s_j})P_{T-t_j}f\right)+\left(\Phi^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x)-\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_{j-1}}f\right)\right|$$

$$\leq\sum_{j=1}^{k}\|(P_{s_j}-Q^m_{s_j})P_{T-t_j}f\|_\infty+\sum_{j=0}^{k-1}\left|\left(\Phi^{m,j}_{\mathcal{D},(u,r)}(\delta_x)-\widehat{\Phi}^{m,j}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_j}f\right)\right|, \tag{5.6}$$

where the first sum of the upper bound is due to the KLV approximation. The second sum is the error caused by the recombination.

Suppose that f is Lipschitz continuous. The smoothness of P_tf leads to

$$\sup_{x\in\mathbb{R}^N}\left|\left(\Phi^{m,j}_{\mathcal{D},(u,r)}(\delta_x)-\widehat{\Phi}^{m,j}_{\mathcal{D},(u,r)}(\delta_x),P_{T-t_j}f\right)\right|\leq Cu_j^{r_j+1}\sup_{|\alpha|=r_j+1}\|D^\alpha P_{T-t_j}f\|_\infty \tag{5.7}$$

for $0\leq j\leq k-1$, where equation (5.3) and the triangle inequality are used. Like the case of equation (4.12), a suitable condition on $\{V_i\}_{i=0}^d$ ensures there exists a positive integer $p\in\mathbb{N}$ such that

$$\sup_{|\alpha|=r+1}\|D^\alpha P_tf\|_\infty\leq Ct^{-rp/2}\|\nabla f\|_\infty \tag{5.8}$$

for all $t\in(0,1]$. When equations (4.12) and (5.8) are satisfied [6, 25, 29], one obtains

$$\sup_{x\in\mathbb{R}^N}\left|P_Tf(x)-\left(\Phi^{m,k}_{\mathcal{D},(u,r)}(\delta_x),f\right)\right|$$
$$\leq\left(C_1\left(s_k^{1/2}+\sum_{i=m+1}^{m+2}\sum_{j=1}^{k-1}\frac{s_j^{i/2}}{(T-t_j)^{(i-1)/2}}\right)+C_2\sum_{j=1}^{k-1}\frac{u_j^{r_j+1}}{(T-t_j)^{r_jp/2}}\right)\|\nabla f\|_\infty \tag{5.9}$$

from equations (4.13) and (5.7). Here C_1 and C_2 are constants.

The recombination error can be controlled by the radius of the ball u_j and the cubature on $\mathbb{R}^N$ degree r_j. By choosing an appropriate pair (u_j,r_j), one can make the order of the recombination error bound not dominant over the order of the error bound in the KLV method. For example, in the case of $(u_j,r_j)=(s_j^{p/2-a},\lceil m/p\rceil)$ where $a=(p-1)/(2(\lceil m/p\rceil+1))$ ($\lceil x\rceil$ denotes the smallest integer greater than or equal to x) or $(u_j,r_j)=((s_j^{m+1}/(T-t_j)^{m-rp})^{1/2(r+1)},m)$, the error estimate

$$\sup_{x\in\mathbb{R}^N}\left|P_Tf(x)-\left(\Phi^{m,k}_{\mathcal{D}(\gamma),(u,r)}(\delta_x),f\right)\right|\leq C\|\nabla f\|_\infty T^{1/2}k^{-(m-1)/2} \tag{5.10}$$

is satisfied for a Lipschitz continuous f when $\gamma>m-1$. Equation (5.10) is obtained from substituting the partition defined in equation (4.14) into equation (5.9) and shows the same convergence rate as the ones without recombination, equations (4.15) and (4.25).

6. Patched cubature filter and adaptive patched cubature filter

Recall that $X(t)\in\mathbb{R}^N$ is governed by

$$dX(t)=V_0(X(t))\,dt+\sum_{i=1}^{d}V_i(X(t))\circ dW_i(t). \tag{6.1}$$

Let the noisy observations Y_n associated with $X_n=X(nT)$ satisfy

$$Y_n=\varphi(X_n)+\eta_n,\quad \eta_n\sim\mathcal{N}(0,R_n), \tag{6.2}$$

where $\varphi \in C_b^\infty(\mathbb{R}^N, \mathbb{R}^{N'})$ and realisations of the noise η_n are i.i.d. random vectors in $\mathbb{R}^{N'}$.

For a deterministic particle approximation of the optimal filtering solution of equations (6.1) and (6.2), we employ the KLV method and recombination to define the patched cubature filter (PCF) in Subsection 6.1 and and the adaptive patched cubature filter (APCF) in Subsection 6.2. We address several issues encountered during their practical implementations in Subsection 6.3.

6.1. Patched cubature filter (PCF). Let $\pi_{n|n'}$ be the law of the conditioned variable $X_n|Y_{1:n'}$ and $\pi_{0|0}^{\mathrm{PCF}}$ be a discrete measure approximation of the law of $X(0)$. We define the *patched cubature filter (PCF) at the path level* by the recursive algorithm

$$\begin{aligned} \pi_{n|n-1}^{\mathrm{PCF}} &= \Phi_{\mathcal{D},(u,r)}^{m,k}(\pi_{n-1|n-1}^{\mathrm{PCF}}), \\ \pi_{n|n}^{\mathrm{PCF}} &= \mathrm{REW}\left(\pi_{n|n-1}^{\mathrm{PCF}}, g^{y_n}\right), \end{aligned} \tag{6.3}$$

for $n \geq 1$. The algorithm can be stated as the following.

1. One breaks the measure into patches and performs individual recombination for each one.
2. One moves the given discrete measure forward in time using the KLV method.
3. One performs data assimilation via bootstrap reweighting at every inter-observation time which might differ from the time step for the numerical integration.
4. One again applies the patched recombination.

Using $\pi_{n-1|n-1}^{\mathrm{PCF}}$ in place of δ_x in equation (5.6), an error bound of the prior approximation of the PCF is given by

$$\begin{aligned} \left|(\pi_{n|n-1} - \pi_{n|n-1}^{\mathrm{PCF}}, f)\right| &\leq \left|(\pi_{n-1|n-1}, P_T f) - (\pi_{n-1|n-1}^{\mathrm{PCF}}, P_T f)\right| \\ &\quad + \left|(\pi_{n-1|n-1}^{\mathrm{PCF}}, P_T f) - (\Phi_{\mathcal{D},(u,r)}^{m,k}(\pi_{n-1|n-1}^{\mathrm{PCF}}), f)\right| \\ &\leq \left|(\pi_{n-1|n-1} - \pi_{n-1|n-1}^{\mathrm{PCF}}, P_T f)\right| + \sum_{j=1}^{k} \| (P_{s_j} - Q_{s_j}^m) P_{T-t_j} f \|_\infty \\ &\quad + \sum_{j=0}^{k-1} \left|\left(\Phi_{\mathcal{D},(u,r)}^{m,j}(\pi_{n-1|n-1}^{\mathrm{PCF}}) - \widehat{\Phi}_{\mathcal{D},(u,r)}^{m,j}(\pi_{n-1|n-1}^{\mathrm{PCF}}), P_{T-t_j} f\right)\right|. \end{aligned} \tag{6.4}$$

One can use the same argument with the case of the PDE problem to obtain a higher order estimate of the PCF. An error bound of the posterior approximation

$$\begin{aligned} &\left|(\pi_{n|n} - \pi_{n|n}^{\mathrm{PCF}}, f)\right| \\ &= \left| \frac{(\pi_{n|n-1}, f g^{y_n})}{(\pi_{n|n-1}, g^{y_n})} - \frac{(\pi_{n|n-1}^{\mathrm{PCF}}, f g^{y_n})}{(\pi_{n|n-1}, g^{y_n})} + \frac{(\pi_{n|n-1}^{\mathrm{PCF}}, f g^{y_n})}{(\pi_{n|n-1}, g^{y_n})} - \frac{(\pi_{n|n-1}^{\mathrm{PCF}}, f g^{y_n})}{(\pi_{n|n-1}^{\mathrm{PCF}}, g^{y_n})} \right| \\ &\leq \frac{1}{(\pi_{n|n-1}, g^{y_n})} \left|(\pi_{n|n-1} - \pi_{n|n-1}^{\mathrm{PCF}}, f g^{y_n})\right| + \frac{\| f \|_\infty}{(\pi_{n|n-1}, g^{y_n})} \left|(\pi_{n|n-1} - \pi_{n|n-1}^{\mathrm{PCF}}, g^{y_n})\right| \end{aligned} \tag{6.5}$$

is given in terms of an error estimate of the prior approximation. We stress that PCF does not include any Monte-Carlo subroutine and therefore its error estimate for the weak approximation can be of high order with respect to the number of iterations k in view of equations (6.4) and (6.5).

Recall that the path level operator $\mathrm{KLV}^{(m)}$ can be replaced by the flow level operator $\widetilde{\mathrm{KLV}}^{(m)}$ without harming the order of accuracy. By doing this in the PCF at the path level, we define the *PCF at the flow level* by the successive algorithm that produces $\widetilde{\pi}^{\mathrm{PCF}}_{n|n-1}$ and $\widetilde{\pi}^{\mathrm{PCF}}_{n|n}$.

6.2. Adaptive patched cubature filter (APCF). It would be worthwhile to mention that PCF (6.3), like SIR (3.3), performs the prior approximation without using the next time observational data. This naturally leads us to develop a variant of PCF that will share the common essential feature with SISR (3.5) in the aspect that the observation process is involved in moving particles forward in time.

In order to do that, we first consider a modification of the standard KLV scheme in which some particles are adaptively accelerated when it causes no significant difference in the integration of the test function. If the smoothness of the test function is not known in advance, the accuracy requirement of the KLV numerical approach leaves no choice other than to let the family of particles forward following the pre-specified partition until the next observation time. This is because, for truly irregular test functions, accurate integration would require exploration of the irregularities. However if the test function is smooth enough and the less regular set is of significantly lower dimension than the main part of the smoothness, then we are allowed to let the particles go straight to the next observation time from some considerable distance back instead of the step predicted in the worst case which we would otherwise have used to terminate the algorithm.

We build this insight into the practical algorithm. At each application of the KLV operator, the algorithm evaluates the test function using a one step prediction straight to the next observation time and compares this with the evaluation using a two step (one next step and the rest step to the next observation time) prediction. If two evaluations agree within the error tolerance, then the particles immediately leap to the next observation time. Otherwise the prediction will follow the original partition.

In terms of accuracy, the approach is pragmatically rather successful because the opportunities for two (or three to break certain pathological symmetries) step prediction to produce consistent answers by chance are essentially negligible. Furthermore, the adaptive switch for which the KLV is employed to move the prediction measure forward but move a part of it straight to the observation time whenever the relevant part of the test function is smooth enough has a very significant effect of pruning the computation and speeding up the algorithm due to the reduction of particles to be recombined at each iteration.

This adaptive KLV method of course cannot be applied without a test function. Differently from the PDE problem, the test function is not specified in the filtering problem. Therefore in practice we take the smooth likelihood as test function to lead the adaptation.

Recall, $\mathcal{D}=\{0=t_0<t_1<\cdots<t_k=T\}$ is a partition of $[0,T]$ with $s_j=t_j-t_{j-1}$. We use the likelihood g^{y_n} to define the splitting operator acting on a discrete measure $\mu^{j-1}=\sum_{i=1}^{n}\kappa_i\delta_{x^i}$ at time t_{j-1}. Let $\mu^{j-1}_{i,21}=\mathrm{KLV}(\delta_{x^i},t_j-t_{j-1})$, $\mu^{j-1}_{i,22}=\mathrm{KLV}(\mu_{i,21},t_k-t_j)$, and $\mu^{j-1}_{i,1}=\mathrm{KLV}(\delta_{x^i},t_k-t_{j-1})$. Let I_τ be the collection of index i satisfying $|(\mu^{j-1}_{i,1}-\mu^{j-1}_{i,22},g^{y_n})|<\tau$. Then the discrete measure μ^{j-1} is the union of two discrete measure $\mu^{j-1}=\mu^{j-1,<\tau}\sqcup\mu^{j-1,\geq\tau}$ where $\mu^{j-1,<\tau}=\sum_{i\in I_\tau}\kappa_i\delta_{x^i}$. For simplicity, $\mu^{k-1,\geq\tau}$ is defined

to be the null set. The process defines the splitting operator

$$\mathrm{SPL}^{(\tau)}\left(\mu^{j-1}, g^{y_n}\right) \equiv \mu^{j-1,<\tau} \tag{6.6}$$

for $1 \le j \le k$.

Define a sequence of discrete measures as follows:

$$\begin{aligned}
&\Phi^{m,0}_{\mathcal{D},(u,r),\tau}(\mu^0) = \mu^0,\\
&\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r),\tau}(\mu^0) = \mathrm{REC}^{(u_{j-1},r_{j-1})}\left(\Phi^{m,j-1}_{\mathcal{D},(u,r),\tau}(\mu^0)\right),\\
&\widehat{\Phi}^{m,j-1,<\tau}_{\mathcal{D},(u,r),\tau}(\mu^0) = \mathrm{SPL}^{(\tau)}\left(\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r),\tau}(\mu^0), g^{y_n}\right),\\
&\Phi^{m,j}_{\mathcal{D},(u,r),\tau}(\mu^0) = \mathrm{KLV}^{(m)}\left(\widehat{\Phi}^{m,j-1,<\tau}_{\mathcal{D},(u,r),\tau}(\mu^0), s_j\right),
\end{aligned} \tag{6.7}$$

for $1 \le j \le k$. Let $\widehat{\Phi}^{m,j-1}_{\mathcal{D},(u,r),\tau}(\mu^0) = \widehat{\Phi}^{m,j-1,<\tau}_{\mathcal{D},(u,r),\tau}(\mu^0) \sqcup \widehat{\Phi}^{m,j-1,\ge\tau}_{\mathcal{D},(u,r),\tau}(\mu^0)$ and

$$\Psi^{m,j-1,k}_{\mathcal{D},(u,r),\tau}(\mu^0) = \mathrm{KLV}^{(m)}\left(\mathrm{KLV}^{(m)}\left(\widehat{\Phi}^{m,j-1,\ge\tau}_{\mathcal{D},(u,r),\tau}(\mu^0), t_j - t_{j-1}\right), T - t_j\right), \tag{6.8}$$

for $1 \le j \le k-1$.

We define the *adaptive patched cubature filter (APCF) at the path level* by

$$\begin{aligned}
\pi^{\mathrm{APCF}}_{n|n-1} &= \left(\bigsqcup_{j=1}^{k-1} \Psi^{m,j-1,k}_{\mathcal{D},(u,r),\tau}(\pi^{\mathrm{APCF}}_{n-1|n-1})\right) \sqcup \Phi^{m,k}_{\mathcal{D},(u,r),\tau}(\pi^{\mathrm{APCF}}_{n-1|n-1}),\\
\pi^{\mathrm{APCF}}_{n|n} &= \mathrm{REW}\left(\pi^{\mathrm{APCF}}_{n|n-1}, g^{y_n}\right),
\end{aligned} \tag{6.9}$$

for $n \ge 1$. The algorithm can be stated as the following.

1. One breaks the measure into patches and performs individual recombination for each one.
2. One splits given discrete measure to lead some of the particles to the next observation time and the rest of the particles to the next iteration time using the KLV method.
3. One performs data assimilation via bootstrap reweighting at every inter-observation time which might differ from the time step for the numerical integration.
4. One again applies the patched recombination.

By replacing $\mathrm{KLV}^{(m)}$ by $\widetilde{\mathrm{KLV}}^{(m)}$, we define the *APCF at the flow level* that produces $\widetilde{\pi}^{\mathrm{APCF}}_{n|n-1}$ and $\widetilde{\pi}^{\mathrm{APCF}}_{n|n}$ instead of $\pi^{\mathrm{APCF}}_{n|n-1}$ and $\pi^{\mathrm{APCF}}_{n|n}$.

In view of equation (6.5), the likelihood is indeed a natural choice for the filtering problem in which the posterior measure is of primary interest. One can apply g^{y_n} and fg^{y_n} simultaneously as the test function for the SPL operator in equation (6.7) if one would like to obtain a posterior approximation that accurately integrates f.

Note that both SISR (3.5) and APCF (6.9) are built upon the same philosophy – making use of the observational information to lead the particles for a more accurate approximation of the posterior possibly at the expense of the accuracy of the corresponding prior approximation. However the way of modifying the basis algorithm is different from each other. In particular, while SISR leads the particles only using the instance of the observation y_n, APCF fully uses the likelihood g^{y_n} to achieve the adaptation. Furthermore APCF cares about the domain of importance without introducing a new dynamics.

6.3. Practical implementation. One has to specify the time partition and the way of patched recombination for the implementation of PCF and APCF. We here present adaptive partition and adaptive recombination as alternatives to the Kusuoka partition and the covering with fixed-size balls, respectively. Differently from prior suggestions, ours are subject to the test function and thus called adaptive.

Before doing that, we at this point mention that the work in [8] also employs cubature on Wiener space to solve the nonlinear filtering problem. Comparing these two kinds of cubature filters, one major difference is how to simplify the support of discrete measure between successive KLV applications to control computational cost. The one developed in [8] makes use of the Monte-Carlo scheme based on a branching and pruning mechanism. The algorithm looks for a reduced measure whose distance from the original measure is minimised in some sense. Therefore the simplification procedure should be applied to the whole discrete measure all at once. On the contrary, PCF and APCF take the deterministic moment-matching recombination strategy, which can be applied locally in the support of measure for an enhanced efficiency.

In addition to the algorithm characteristics, the problem setting in [8] is rather different from the current paper as the observation process is assumed to be not discrete but continuous (for more details, we refer the reader to [28]). In this case, the time integration of the KLV method is performed along with even partition of small intervals. However, in case of sparse observations, the numerical integration until the next observation time requires multiple steps, preferably with uneven partition of decreasing intervals rather than even partition. For PCF and APCF, the likelihood can serve as the test function, and we can further utilise the presence of this test function to determine time partition. This is clearly one additional degree of freedom allowed in the cubature filter under the scenario of intermittent observations.

6.3.1. Adaptive partition. For a given test function f, one can make use of the heat kernel P_t as well as f to evolve the set of particles so that one step error is within a given degree of accuracy, i.e.,

$$\| (P_{s_j} - Q^m_{s_j}) P_{T-t_j} f \|_\infty < \epsilon \tag{6.10}$$

for some $\epsilon > 0$. We define an adaptive partition $\mathcal{D}(\epsilon, f) = \{t_j\}_{j=0}^k$ to be a time discretisation for which each $s_j = t_j - t_{j-1}$ is the supremum among the ones satisfying equation (6.10). Because $P_t f$ becomes smoother as t increases, the sequence $\{s_j\}_{j=1}^k$ tends to decrease monotonically, i.e. $s_1 \geq s_2 \geq \cdots \geq s_k$. The upper bound of the total error along with the adaptive partition is given by

$$\sup_x \left| P_T f(x) - (\Phi^{m,k}_{\mathcal{D}}(\delta_x), f) \right| < k\epsilon \tag{6.11}$$

from equation (4.11).

6.3.2. Adaptive recombination. Consider the condition

$$\left| \left(\Phi^{m,j}_{\mathcal{D},(u,r)}(\mu^0) - \widehat{\Phi}^{m,j}_{\mathcal{D},(u,r)}(\mu^0), P_{T-t_j} f \right) \right| < \theta \tag{6.12}$$

given some $\theta > 0$. We define the adaptive recombination by the algorithm that uses as large of a value of u as possible among the ones satisfying equation (6.12) for a fixed recombination degree r. The algorithm again makes use of the heat kernel P_t as well as the test function f. When the adaptive partition and the adaptive recombination are

simultaneously used, the combination of equations (6.10) and (6.12) yields

$$\sup_x \left| P_T f(x) - (\Phi^{m,k}_{\mathcal{D},(u,r)}(\delta_x), f) \right| < k(\epsilon+\theta) \tag{6.13}$$

from equation (5.6). Notice, unlike the case of equation (5.10) where the constant C is not specified, the upper bound of equation (6.13) is explicitly under the control.

It deserves to be mentioned that the application of the adaptive recombination does not require us to determine the size (and even topology) of patches in advance. Given the recombination degree, it suffices to keep shrinking the size of patches until equation (6.12) is met. Due to this feature, the adaptive recombination can practically be useful in achieving the error bound of equation (6.13) when it is accompanied with an efficient algorithm that divides the support of discrete measure into local disjoint subsets. Because the detailed algorithm of the recombination can be found in [29], we conclude this section with one way to achieve the adaptive recombination utilising the Morton ordering [32]. The methodology adopts boxes, instead of balls, as patches to locally cover the particles. The algorithm is advantageous particularly in the case of high dimension.

Given a number of particles in N dimension, we perform an affine transformation to map the particles into the ones in the box $[0.5,1)^N$. In the following, we evenly divide each edge of the box by 2^n to get 2^{nN} sub-boxes and assign the particles to these sub-boxes. We use the double-precision floating-point format in scientific computing: any number $z^i \in [0.5,1)$ is saved in terms of $\{b^i_j\}_{j=1}^{52}$, where b^i_j is either 0 or 1 in a way that $z^i = (1/2)\times(1+\sum_{j=1}^{52} b^i_j 2^{-j})$, (almost all numbers in $[0.5,1)$ have a binary expansion of more than 52 digits, but this reduced information is quite enough for our purpose). In this way, the point $(z^1,\ldots,z^N)$ in N-dimension can be expressed by $52\times N$ binary numbers. Interleaving the binary coordinate values yields binary values. Connecting the binary values in their numerical order produces the Morton ordering. Then an appropriate coarse-graining leads to the subdivision of a box. For example, when $N=2$, the binary value corresponding (z^1,z^2) is $b^1_1 b^2_1 b^1_2 b^2_2 \cdots b^1_{52} b^2_{52}$. The point is in first quadrant if $(b^1_1, b^2_1) = (1,1)$, in second quadrant if $(b^1_1, b^2_1) = (0,1)$, in third quadrant if $(b^1_1, b^2_1) = (0,0)$ and in fourth quadrant if $(b^1_1, b^2_1) = (1,0)$. Applying this classification to a number of particles produces 2^2 disjoint subsets of classified particles. Similarly, using $b^1_1 b^2_1 b^1_2 b^2_2$ and ignoring the remaining subgrid scales gives 4^2 subsets when $N=2$. Taking the inverse affine transformation, a classification of the particles has been achieved.

The crucial point is that, by sorting the one dimensional transformed points, one keeps points in a box together without ever needing to introduce the boxes, and particularly empty boxes. The complexity of the clustering is no worse than $MN\log M$ in the number of points. Here M is number of particles, N is dimension, and $MN\log M$ is the cost of patching. Note $N\log M$ is the cost of getting those points in a patch.

7. Numerical simulations

We perform numerical simulations to examine the efficiency and accuracy of the proposed filtering approaches. We introduce the test model in Subsection 7.1 and obtain the reference solutions in Subsection 7.2. We implement the PCF and APCF with cubature on Wiener space of degree $m=5$ in Subsection 7.3. Finally, in Subsection 7.4, we investigate the prospective performance of PCF and APCF with cubature on Wiener space of degree $m=7$.

7.1. Test model. It is very important to select a good example to examine the performance of the algorithms we have developed. Here we choose a forward model and

observation process for which the analytic solution of the filtering problem is known and can be used to measure the accuracy of the various particle approximations.

Our test model is the Ornstein–Uhlenbeck process [39] in three dimension:

$$dX = -\Lambda X\,dt + gI_3 dW, \tag{7.1}$$

where $X=(x^1\,x^2\,x^3)^t$, $\Lambda=\begin{pmatrix}\sigma & -\sigma & 0\\ -\rho & 1 & 0\\ 0 & 0 & \beta\end{pmatrix}$, $dW=(dW_1\,dW_2\,dW_3)^t$, and I_3 denotes the 3×3 identity matrix. Here the superscript t denotes the transpose. The parameter values $\sigma=1$, $\rho=0.28$, $\beta=8/3$, and $g=0.5$ are chosen. The observations

$$Y_n = X_n + \eta_n, \quad \eta_n \sim \mathcal{N}(0, R_n) \tag{7.2}$$

are available at every inter-observation time $T=0.5$. We study the cases in which the covariance of observation noise is $R_n = R\times I_3$ for the values $R=10^{-1},10^{-2}$, and 10^{-3}.

7.2. Reference Solutions.

7.2.1. Kalman filter. The conditioned measure for equations (7.1), (7.2) is Gaussian and $\pi_{n|n'}=\mathcal{N}(M_{n|n'},C_{n|n'})$ can be obtained from the Kalman filter. In this case, the prior covariance $C_{n|n-1}$ satisfies the Riccati difference equation and its solution converges as n increases [3]. We take the covariance of the initial condition $X(0)$ as the one step prediction from the limit of the Riccati equation solution so that $C_{n|n-1}$ and $C_{n|n}$ do not depend on n (but $M_{n|n-1}$ and $M_{n|n}$ depend on n). We see that the diagonal elements of $C_{n|n-1}$ are about 10^{-1} for all cases of $R=10^{-1},10^{-2},10^{-3}$. The diagonal elements of $C_{n|n}$ are about 10^{-1} when $R=10^{-1}$, about 10^{-2} when $R=10^{-2}$ and about 10^{-3} when $R=10^{-3}$.

In this filtering problem, we first investigate *where are the observations*. We apply the Kalman filter for $1\le n\le 10^8$ and calculate the values of D_1, D_2, and D_3 satisfying $y_n = M_{n|n-1} + (D_1\,D_2\,D_3)^t\cdot\sqrt{\mathrm{diag}(C_{n|n-1})}$, where y_n is determined by one trajectory of the dynamics (7.1) together with a realisation of the observation noise η_n. The histograms in Figure 7.1 show the distribution of these normalised distances between the observation and the prior mean when $R=10^{-2}$ (the cases of $R=10^{-1}$ and $R=10^{-3}$ are similar and not shown). One can see that most of the observations are within two standard deviations from the prior mean in each coordinate. Among the cases of 10^8, there are 4592208 cases for which $|D_i|>1$ for all $i=1,2,3$ at the same time. There are 37574 cases for which $|D_i|>2$ for all i at the same time, and 60 cases for which $|D_i|>3$ for all i at the same time. From the simulation, we understand the three cases in which the parameter value of $D\equiv D_1=D_2=D_3$ is 1, 2, and 3, are normal, exceptional, and rare event, respectively.

7.2.2. L^2 norm of the higher-order central moments. Here we aim to investigate the parameter regimes under which our cubature filters are likely (or unlikely) to outperform. In order to evaluate the computational error, one needs to define an error criterion relevant to the approximations. We realise that unfortunately a comparison between the evolving single trajectory and the corresponding posterior mean approximation, which is commonly used in the filtering context, is highly inappropriate for our purpose. This is because the cubature approximation is basically superior within approximating the tail behaviour or higher-order moments of the probability distribution. Therefore we instead use the L^2 norm of the central moment to quantify the accuracy of the approximation obtained in the form of discrete measure.

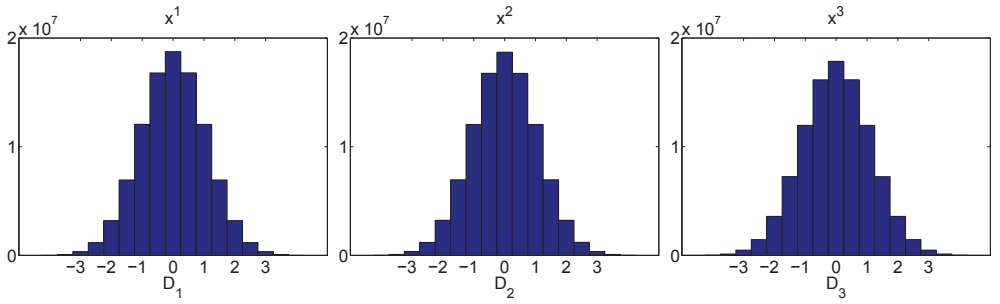


FIG. 7.1. *The distribution of normalised distances between the observation and the prior mean when the noise covariance is* $R_n = 10^{-2} \times I_3$.

Let $\mathcal{C}^p$ be the pth central moment of $X = (x^1\, x^2\, x^3)^t$, i.e.

$$\mathcal{C}^p_{i_1,\cdots,i_p} = \mathbb{E}\left(\prod_{j=1}^{p}\left(x^{i_j} - \mathbb{E}(x^{i_j})\right)\right)$$

where $i_j = 1,2,3$. The L^2 norm of $\mathcal{C}^p$ is defined by

$$\| \mathcal{C}^p \|_2 \equiv \left(\sum_{i_1,\cdots,i_p=1}^{3} |\mathcal{C}^p_{i_1,\cdots,i_p}|^2\right)^{1/2}. \tag{7.3}$$

When $p = 1$, equation (7.3) is the Euclidean norm of the vector. When $p = 2$, it is equivalent with the Frobenius norm of the matrix. Let $\widehat{\mathcal{C}}^p$ be the pth central moment of a particle approximation. Then the relative root mean square error (RMSE)

$$\text{RMSE}\% \equiv \| \mathcal{C}^p - \widehat{\mathcal{C}}^p \|_2 \,/\, \| \mathcal{C}^p \|_2 \tag{7.4}$$

will be calculated to measure the accuracy of the moment approximations.

7.2.3. Monte-Carlo Gaussian samples. In our problem setting, the RMSE errors are insensitive to the specific time interval between successive observations. Taking one arbitrary time interval, we study the cases of $D = 1,2,3$, which correspond to a normal, an exceptional, and a rare event. The scenario initially may look somewhat artificial because, unlike the filtering in practice, the observational data is not generated from realisations. However, we emphasise that it has been carefully designed, while keeping the practical relevance, in order to find the parameter regimes under which our approaches outperform Monte-Carlo methods, and this will eventually turn out to be extremely helpful for a deeper understanding of the filtering problem.

We perform Gaussian sampling to obtain three different Monte-Carlo approximations of the posterior measure. For the first one, we draw samples from the prior measure and subsequently apply the bootstrap reweighting to obtain the posterior approximation. One can regard these bootstrap reweighted samples from the prior as the SIR result. The second one is from the SISR algorithm under the transition kernel $\widetilde{K}(dx_n|x_{n-1},y_n) = \mathbb{P}(dx_n|x_{n-1},y_n)$, which is the optimal proposal in the sense of minimising the variance of the importance weights [14]. Finally, we draw samples directly from the posterior measures as the third one. Note, in all Monte-Carlo approximations,

	$\epsilon=10^{-2}$	$\epsilon=10^{-3}$	$\epsilon=10^{-4}$	$\epsilon=10^{-5}$
$R=10^{-1}$	7	31	102	344
$R=10^{-2}$	10	29	101	330
$R=10^{-3}$	20	48	120	329

TABLE 7.1. *The number of adaptive partition k for KLV with* $m=5$

neither truncation error due to numerical integration nor resampling error is induced for a fair comparison. The RMSE errors (7.4) of these Gaussian samples are depicted in Figure 7.4 when $R=10^{-2}$, $D=1,2,3$ and in Figure 7.5 when $D=1$, $R=10^{-1},10^{-2},10^{-3}$. These results will be compared with the cubature filters.

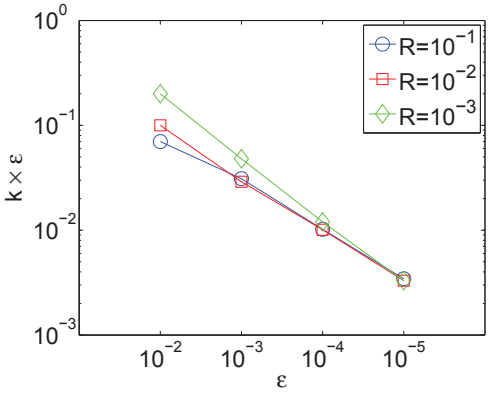
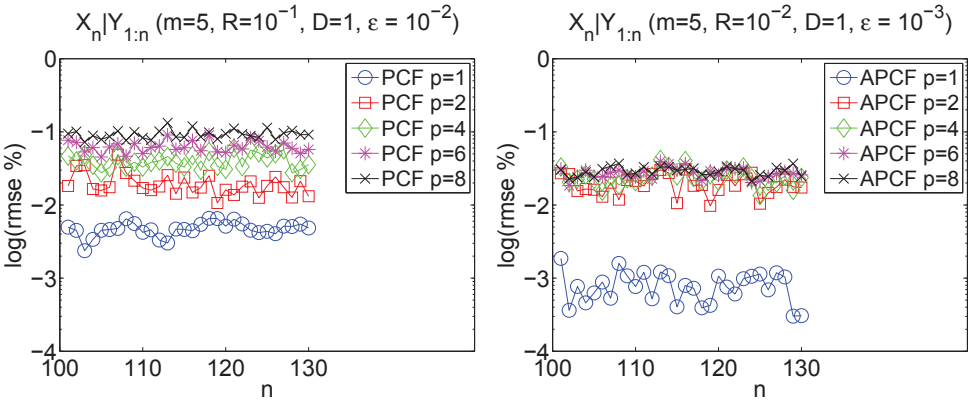


FIG. 7.2. *The upper bound of the total error along with the adaptive partition when* $m=5$.



(a) PCF using adaptive partition with $\epsilon=10^{-2}$ (b) APCF using adaptive partition with $\epsilon=10^{-3}$

FIG. 7.3. *The relative* L^2 *errors for the pth moments of the evolutionary posterior.*

7.3. PCF and APCF with cubature on Wiener space of degree 5.

7.3.1. Choice of parameters. Here, we implement the PCF and APCF at the flow level. In the case of $d=3$, i.e. when the system is driven by three independent white noises, cubature on Wiener space of degree $m=3$ and $m=5$ (with support size $n_m=6$ and $n_m=28$ respectively) are available. We apply the KLV operator with degree $m=5$.

Using the likelihood g^{y_n} as the test function f, the adaptive partition $\mathcal{D}(\epsilon, g^{y_n})$ satisfying equation (6.10) with $\widetilde{Q}^m_{s_j}$ in place of $Q^m_{s_j}$ is analytically obtained for the system of equations (7.1). Note that the likelihood g^{y_n} is the density function of $\mathcal{N}(y_n, R_n)$ and that the adaptive partition does not depend on y_n but on R_n. The number of iterations k as a function of ϵ and R is listed in Table 7.1. In this case, Figure 7.2 reveals the upper bound of equation (6.11) tends to decrease as ϵ becomes smaller. Therefore, by choosing θ to be the same order of ϵ, one can combine the adaptive partition and the adaptive recombination to achieve a desired degree of accuracy to some extent.

For the recombination of the PCF, equation (6.12) with $f=g^{y_n}$ for all $y_n \in \mathbb{R}^N$, i.e.

$$\sup_{y_n} \left| \left(\Phi^{m,j}_{\mathcal{D},(u,r)}(\mu^0) - \widehat{\Phi}^{m,j}_{\mathcal{D},(u,r)}(\mu^0), P_{T-t_j} g^{y_n} \right) \right| < \theta, \tag{7.5}$$

is met such that the recombination does not depend on y_n but on R_n. We choose the recombination degree $r=5$ and simulate the PCF for the cases of $\epsilon = 10^{-2}, 10^{-3}$ with $\theta = 0.3 \times \epsilon$.

For the APCF, the tolerance τ has to be chosen in addition to the parameters $\{\epsilon, \theta\}$. The value of τ varies in each case, but we choose it so that the SPL operator in equation (6.7) allows $1/4 \sim 1/3$ part of particles leap to the next observation time for all iterations except the first and last few steps. The remaining particles are reduced by the adaptive recombination, i.e., the recombination satisfies

$$\left| \left(\Phi^{m,j}_{\mathcal{D},(u,r),\tau}(\mu^0) - \widehat{\Phi}^{m,j}_{\mathcal{D},(u,r),\tau}(\mu^0), P_{T-t_j} g^{y_n} \right) \right| < \theta \tag{7.6}$$

where $\mu^0 = \widetilde{\pi}^{\mathrm{APCF}}_{n-1|n-1}$. We again choose the recombination degree $r=5$ and simulate the APCF for the cases of $\epsilon = 10^{-2}, 10^{-3}$ with $\theta = 0.3 \times \epsilon$.

With the value of D being fixed, we apply the PCF and APCF to obtain the values of equation (7.4) for the evolving posterior measure. Figure 7.3 shows that the performances of the two filtering algorithms are stable and that the numerical error estimates of high-order moments are insensitive to n (the remaining cases produce similar plots and are not shown).

In our numerical simulations, the number of patches needed to satisfy equation (7.5) in the PCF increases as the time partition approaches to the next observation time, eventually about $8^3 \sim 16^3$. On the contrary, equation (7.6) in the APCF is satisfied with 2^3 (< 10) patches in most cases. As a result, APCF saves computation time significantly compared with PCF.

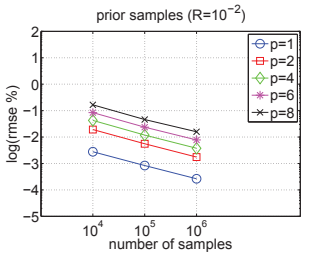
7.3.2. Dependence on the observation location. When $R = 10^{-2}$ is fixed and $D = 1,2,3$ varies, the relative L^2 errors of the pth moments of PCF and APCF are shown in figures 7.4(b), 7.4(i), 7.4(j), and 7.4(k). We have implemented two cases of $\epsilon = 10^{-2}$ and $\epsilon = 10^{-3}$. The recombination times are measured using Visual Studio with Intel 2.53 GHz processor (the autonomous ODEs are solved analytically). Figure 7.4 reveals the following.

- The prior approximation of PCF with $\epsilon = 10^{-3}$ shows similar accuracy with 10^4 Monte-Carlo sampling (figures 7.4(a) and 7.4(b)).
- The accuracy of the APCF prior approximation is in general worse than PCF especially for higher order moments (Figure 7.4(b)).
- As the observation is located further far from the prior mean, i.e., as D increases, the posterior approximation obtained from Monte-Carlo bootstrap reweighting (SIR) becomes less accurate (figures 7.4(c), 7.4(d), and 7.4(e)). As the number of samples M increases, the error reduction asymptotically scales as $M^{-1/2}$ in all cases.
- Unlike the case of SIR, the accuracy of the importance samples (SISR) is not significantly influenced by the observation location as well as the number of samples (figures 7.4(f), 7.4(g), and 7.4(h)). This sample size insensitivity is presumably because SISR duplicates the samples in this parameter regime (compare with Figure 7.5(i)).
- The accuracy of the APCF posterior approximation is similar to PCF but APCF significantly reduces the recombination time which is insensitive to D (figures 7.4(i), 7.4(j), and 7.4(k)).
- The accuracy of the PCF and APCF posterior approximations with $\epsilon = 10^{-2}$ is similar to 10^4 Monte-Carlo reweighted samples (SIR) when $D = 1,2$ (figures 7.4(c), 7.4(i), 7.4(d), and 7.4(j)) and to 10^5 reweighted samples (SIR) when $D = 3$ (figures 7.4(e) and 7.4(k)).
- The accuracy of the PCF and APCF posterior approximations with $\epsilon = 10^{-3}$ is similar to 10^5 Monte-Carlo reweighted samples when $D = 1$ (figures 7.4(c) and 7.4(i)), to 10^6 reweighted samples when $D = 2$ (figures 7.4(d) and 7.4(j)) and to 10^7 reweighted samples when $D = 3$ (figures 7.4(e) and 7.4(k)).
- The accuracy of the PCF and APCF posterior approximations with $\epsilon = 10^{-3}$ is superior to 10^6 importance samples (SISR) when $D = 1$ (figures 7.4(f) and 7.4(i)), comparable to SISR when $D = 2$ (figures 7.4(g) and 7.4(j)), and inferior to SISR when $D = 3$ (figures 7.4(h) and 7.4(k)), in approximating higher-order moments.

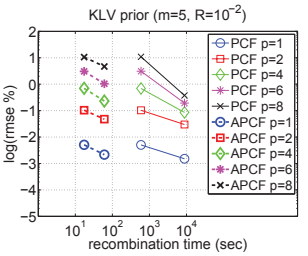
There is an important insight to be gained from this experimental analysis. Though PCF produces a more accurate description of the prior measure than APCF, the one from this naive approximation of the prior is not better at approximating the posterior. The point is that one needs an extremely accurate representation of the prior in certain localities. While APCF delivers this without undue cost, the PCF method would have to deliver this accuracy uniformly and well out into the tail of the prior. As a result, for the posterior approximation, APCF can achieve a similar accuracy to PCF while using significantly less computational cost.

In this example, the computational cost (recombination time) of PCF and APCF is uniform and irrespective of D for given $\epsilon = 10^{-2}, 10^{-3}$. However, when one uses SIR to achieve the accuracy due to APCF with $\epsilon = 10^{-3}$ in approximating higher-order moments, one needs more computational resources (large number of particles) as D becomes bigger. One also cannot expect an accuracy improvement from SISR except the rare event case ($D = 3$). Therefore, in the reliability aspect, APCF is clearly advantageous over sequential Monte-Carlo methods.

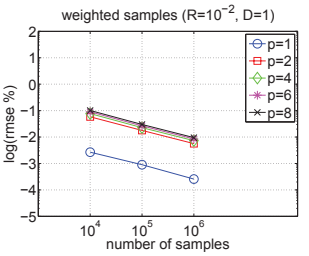
7.3.3. Dependence on the observation noise error. When $D = 1$ is fixed and $R = 10^{-1}, 10^{-2}, 10^{-3}$ varies, the values of equation (7.4) for PCF and APCF are



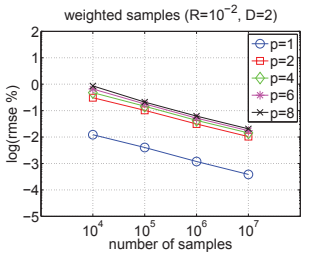
(a) unweighted prior samples



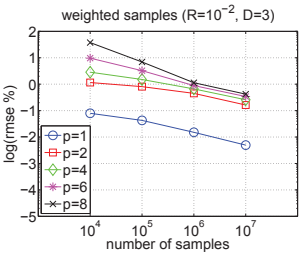
(b) cubature approximation of prior



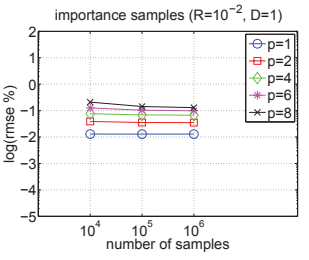
(c) bootstrap reweighted prior samples



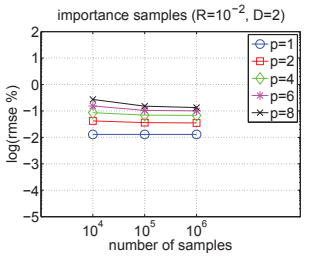
(d) bootstrap reweighted prior samples



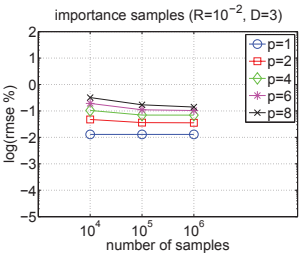
(e) bootstrap reweighted prior samples



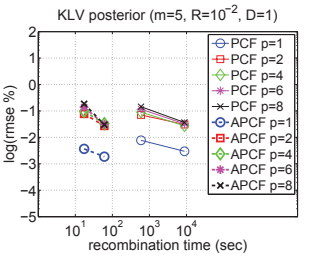
(f) weighted posterior samples



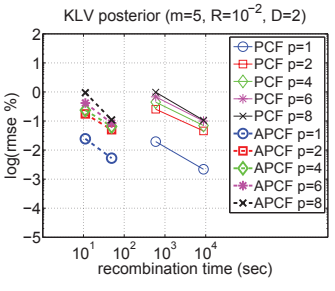
(g) weighted posterior samples



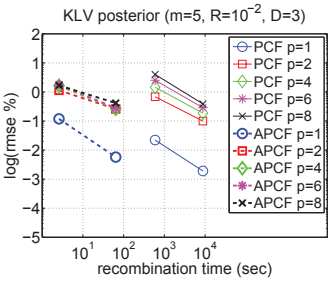
(h) weighted posterior samples



(i) cubature approximation of posterior

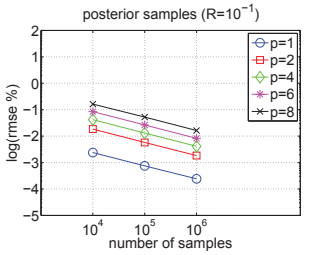


(j) cubature approximation of posterior

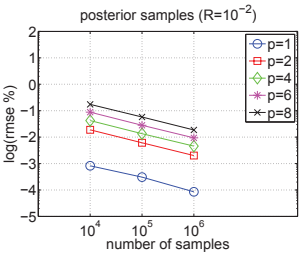


(k) cubature approximation of posterior

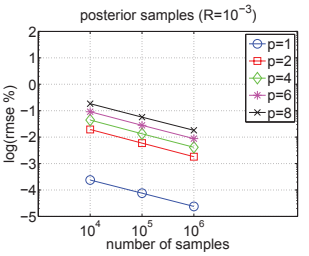
FIG. 7.4. *The prior and posterior approximations when* $R = 10^{-2}$ *is fixed and* $D = 1,2,3$ *varies. The top row is for the prior, and the other three rows are for the posterior. The second row (SIR) and the third row (SISR) are from Monte-Carlo samples. The last row is from cubature approximation when* $\epsilon = 10^{-2}, 10^{-3}$.



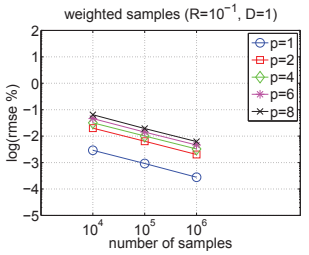
(a) unweighted posterior samples



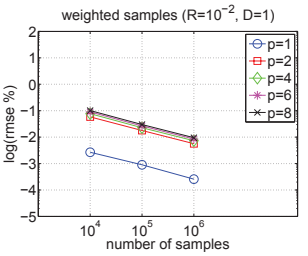
(b) unweighted posterior samples



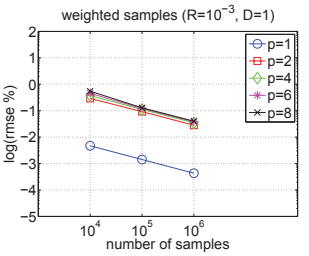
(c) unweighted posterior samples



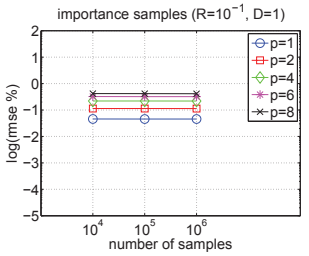
(d) bootstrap reweighted prior samples



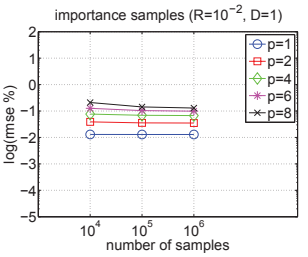
(e) bootstrap reweighted prior samples



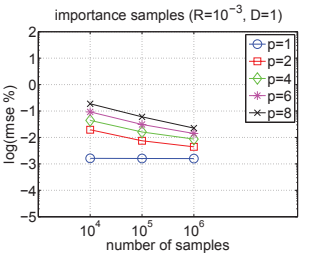
(f) bootstrap reweighted prior samples



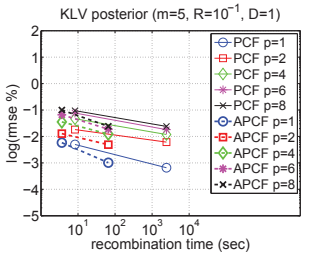
(g) weighted posterior samples



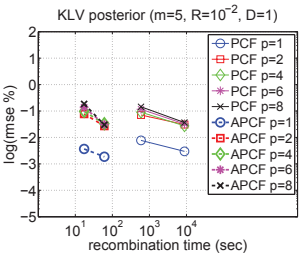
(h) weighted posterior samples



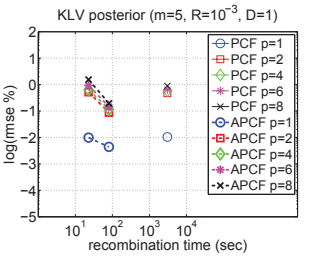
(i) weighted posterior samples



(j) cubature approximation of posterior



(k) cubature approximation of posterior



(l) cubature approximation of posterior

FIG. 7.5. *The posterior approximations when* $D=1$ *is fixed and* $R=10^{-1},10^{-2},10^{-3}$ *varies. The top three rows are from Monte-Carlo samples. The first row is from direct sampling of posterior, the second is from SIR, and the third is from SISR. The last row is from cubature approximation when* $\epsilon=10^{-2},10^{-3}$. *In Figure 7.5(l), the PCF with* $\epsilon=10^{-3}$ *is not shown.*

shown in figures 7.5(j), 7.5(k), and 7.5(l). We have implemented two cases of $\epsilon = 10^{-2}$ and $\epsilon = 10^{-3}$. Figure 7.5 reveals the following.

- The high-order moment approximations errors due to Monte-Carlo Gaussian samples (direct sampling of the posterior) are insensitive to its covariance (recall the diagonal elements of $C_{n|n}$ are of the same order with the value of R) (figures 7.5(a), 7.5(b), and 7.5(c)).
- As the likelihood becomes narrower, i.e. as R decreases, the posterior approximation obtained from Monte-Carlo bootstrap reweighting (SIR) becomes less accurate (figures 7.5(d), 7.5(e), and 7.5(f)).
- The accuracy of importance samples (SISR) tends to increase as R decreases (figures 7.5(g), 7.5(h), and 7.5(i)). In particular, when $R = 10^{-3}$, the moment approximations of SISR is comparable with those from direct sampling of the posterior except the mean (figures 7.5(c) and 7.5(i)).
- As R decreases, the recombination time needed to achieve a given degree of accuracy becomes bigger for PCF, but this is not the case for APCF, i.e. the recombination time for APCF is insensitive to R (figures 7.5(j), 7.5(k), and 7.5(l)).

The simulation shows that APCF again achieves a similar accuracy with PCF in all cases but, as the observation noise error decreases, APCF becomes more competitive than PCF for the solution of the intermittent data assimilation problem. It further shows that APCF is of higher order with respect to the recombination time and can achieve the given degree of accuracy with lower computational cost.

Although Y_n is *there and measurable*, it is sometimes the case that it is actually computationally very expensive to compute and that the thing one can compute is actually the evaluation of likelihood for a number of locations. For example, consider a tracking problem for an object of moderate intensity and diameter that does a random walk, is moving against a slightly noisy background, and is observed relatively infrequently. Its influence is entirely local. The likelihood function will be something like the Gaussian centred at the position of the object but completely uninformative elsewhere in the space. The smaller the object, the tighter or narrower the Gaussian, the harder the problem of finding the object becomes. One can compute the likelihood at any point in the space, but only evaluations at the location of the particle are informative. In that way, one sees the following:

1. The Y_n is *observable* but only partially observed – and with low noise is very expensive to observe accurately, as one has to find the particle.
2. The likelihood can be observed at points in the space.

In this sort of example, it would be quite wrong to assume that, if we know the prior distribution of X_n, then just because $Y_n = X_n + \eta_n$ we know the posterior distribution at zero cost. For sequential Monte-Carlo methods, bootstrap reweighting would seem to give a much better approach.

7.4. Prospective performance PCF and APCF with cubature on Wiener space of degree 7. A cubature formula on Wiener space of degree $m \geq 7$ is currently not available when $d = 3$. However, in our problem setting, we are able to emulate a prospective performance of higher-order cubature formula using Gauss–Hermite quadrature.

For the linear dynamics satisfying

$$X(\Delta) = F_\Delta X(0) + \nu_\Delta, \quad \nu_\Delta \sim \mathcal{N}(0, Q_\Delta),$$

where $F_\Delta \in \mathbb{R}^{3\times 3}$ is a matrix, we define the forward operator

$$\mathrm{GHC}^{(m)}\left(\sum_{i=1}^{n}\kappa_i\delta_{x^i},\Delta\right)\equiv\sum_{i=1}^{n}\sum_{j=1}^{n_m}\kappa_i\lambda_j\delta_{F_\Delta x^i+z^j}, \tag{7.7}$$

where $\{\lambda_j,z^j\}_{j=1}^{n_m}$ is a Gauss–Hermite cubature of degree m with respect to the law of ν_Δ. The authors have seen that the performance of GHC is similar to KLV on the flow level when $m=3,5$ and that equation (7.7) can be used as an alternative to equation (4.23) in the application of PCF and APCF to the test model.

	$\epsilon=10^{-2}$	$\epsilon=10^{-3}$	$\epsilon=10^{-4}$	$\epsilon=10^{-5}$
$R=10^{-1}$	2	4	6	10
$R=10^{-2}$	5	9	16	28
$R=10^{-3}$	9	17	30	54

TABLE 7.2. *The number of adaptive partition k for GHC with* $m=7$

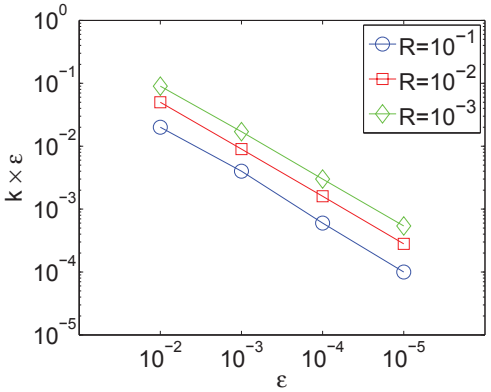
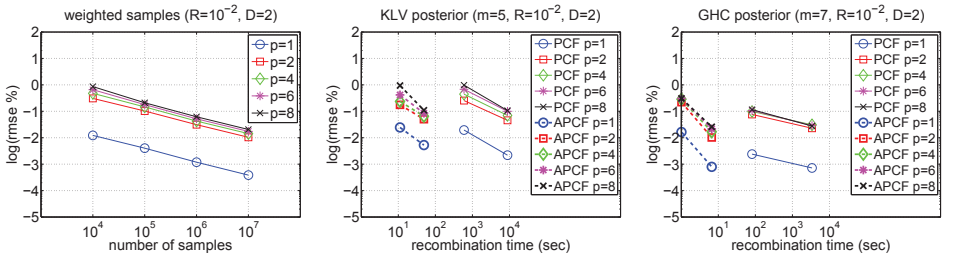


FIG. 7.6. *The upper bound of the total error along with the adaptive partition when* $m=7$.

The number of iterations k in the adaptive partition, obtained from using GHC with Gauss–Hermite cubature of degree $m=7$ whose support size is $n_m=64$ in place of $Q_{s_j}^m$, is shown in Table 7.2. Here Figure 7.6 corresponds to Figure 7.2 and shows an enhanced accuracy. We apply GHC with degree $m=7$ to obtain a prior and posterior approximation, where the recombination degree $r=5$ and $\theta=0.2\times\epsilon$ is used. Our choice of τ is again such that $1/4\sim 1/3$ of the particles are allowed to leap to the next observation time. The RMSE errors (7.4) in the case of $R=10^{-2}$, $D=2$, and $\epsilon=10^{-2},10^{-3}$ are shown in Figure 7.7(c), and this can be viewed as a result from PCF and APCF with cubature on Wiener space of degree $m=7$. Its performance is in fact one higher-order improvement for both accuracy and recombination time in view of figures 7.7(a) and 7.7(b). From the simulation, we expect APCF with higher order cubature formula can outperform Monte-Carlo approximations in any parameter regimes including the ones for which it used to be not so successful when it uses a low order cubature formula. This further highlights the strong necessity to find out cubature formula on Wiener space



(a) bootstrap reweighted prior samples (b) cubature approximation of posterior (c) cubature approximation of posterior

FIG. 7.7. *The posterior approximations when* $R = 10^{-2}$ *and* $D = 2$. *The left is from Monte-Carlo samples and the middle and right is from cubature approximation when* $\epsilon = 10^{-2}, 10^{-3}$.

of degree $m \geq 7$ in order to solve the PDE or filtering problem with high accuracy in a moderate dimension.

8. Discussion

In this paper, we introduced a hybrid methodology for the numerical resolution of the filtering problem which we name the adaptive patched cubature filter (APCF). We explore some of its properties and report on a first attempt at a practical implementation. The APCF combines many different *methods*, each of which addresses a different part of the problem and has independent interest. At a fundamental level, all of the methods use high-order approaches to quantify uncertainty (cubature) and also to reduce the complexity of calculations (recombination based on heavy numerical linear algebra), while retaining explicit thresholds for accuracy in the individual computation. The thresholds for accuracy in a stage are normally achievable in a number of ways (e.g. small time step with low order or large time step with high order) and the determination of these choices depends on computational cost. Aside from this use of the error threshold and choices based on computational efficiency, there are several other points to observe in our development of this filter.

1. One feature is the surprising ease with which one can adapt the computations to the observational data and so avoid performing unnecessary computations. In even moderate dimensions (we work in $3+1$), this has a huge impact for the computation time while preserving the accuracy we achieve for the posterior distribution (figures 7.4(i), 7.4(j), 7.4(k), 7.5(j), 7.5(k), and 7.5(l)). It is an automated form of deterministic high-order importance sampling which has wider application than the one explored in this paper. For instance, it is used to deliver accurate answers to PDE problems with a piecewise smooth test function in the example developed in [29].
2. Another innovation allowing a huge reduction in computation is the ability to efficiently, *patch the particles* in the multiple-dimensional scenario. Although the problem might at first glance seem elementary, it is in fact the problem of data classification. To resolve this problem, we introduce an efficient algorithm for data classification based on extending the Morton order to floating point context. This method has now also been used effectively for efficient function extrapolation [34].
3. The KLV algorithm is at the heart of a number of successful methods for solving

PDEs in moderate dimension [33]. In each case, something has to be done about the explosion of scenarios after each time step, which in turn has to rely on an understanding of the errors. In this paper, we take a somewhat different approach to the literature [31] in the way we use higher-order Lipschitz norms systematically to understand how well functions have been smoothed and to measure the scales on which they can be well-approximated by polynomials. This has the consequence that one can be quite precise about the errors one incurs at each stage in the calculation. In the end, this is actually quite crucial to the logic of our approach since an efficient method requires optimisation over several parameters – something that is only meaningful if there are (at least in principle) uniform estimates on errors. As a result of this perspective, we do not follow the time steps and analytic estimates introduced by Kusuoka in [26], although we remain deeply influenced by balancing the smoothing properties of the semigroup with the use of non-equidistant time steps.

4. The focus on Lipschitz norms makes it natural to apply an adaptive approach to the recombination patches as well as to the prediction process. In both cases we can be lead by the local smoothness of the likelihood function as sampled on our high order high accuracy set of scenarios.
5. We have focussed our attention on the quality of the tail distribution of the approximate posterior we construct. This is important in the filtering problem because a failure to describe the tail behaviour of the tracked object implies that one will lose the trajectory altogether at some point. These issues are particularly relevant in high dimensions as the cost of increasing the frequency of observation can be prohibitive. If one wishes to ensure reliability of the filter in the setting where there is a significant discrepancy between the prior estimate and the realised outcome over a time step, then our APCF with cubature on Wiener space of degree 5 already shows in the three-dimensional example that it can completely outperform sequential importance resampling Monte-Carlo approach. The absence, at the current time, of higher-order cubature formulae is in this sense very frustrating as the evidence we give suggests that higher-degree methods will lead to substantial further benefits for both computation and accuracy.

In putting this paper together, we have realised that there are many branches in this algorithm that can be improved, in particular some parts of the adaptive process and also the recombination (a theoretical improvement in the order of recombination has recently been discovered [38]). There are also large parts that can clearly be parallelised. We believe that there is ongoing scope for increasing the performance of the APCF.

Acknowledgment. The authors would like to thank the following institutions for their financial support of this research. Wonjung Lee: NCEO project NERC and King Abdullah University of Science and Technology (KAUST) Award No. KUK-C1-013-04. Terry Lyons: NCEO project NERC, ERC grant number 291244 and EPSRC grant number EP/H000100/1. The authors also thank the Oxford-Man Institute of Quantitative Finance for its support. The authors thank the anonymous referees for their helpful comments and suggestions, which indeed contributed to improving the quality of the publication.

REFERENCES

[1] B. Anderson and J. Moore, *Optimal Filtering*, Prentice-hall Englewood Cliffs, NJ, 11, 1979.
[2] V. Beneš, *Exact finite-dimensional filters for certain diffusions with nonlinear drift*, Stochastics: An International Journal of Probability and Stochastic Processes, 5, 65–92, 1981.
[3] R.R. Bitmead and M. Gevers, *Riccati difference and differential equations: Convergence, monotonicity and stability*, The Riccati Equation, 263–291, 1991.
[4] J. Carpenter, P. Clifford, and P. Fearnhead, *Improved particle filter for nonlinear problems*, in Radar, Sonar and Navigation, IEE Proceedings, IET, 146, 2–7, 1999.
[5] D. Crisan and A. Doucet, *A survey of convergence results on particle filtering methods for practitioners*, Signal Processing, IEEE Transactions on, 50, 736–746, 2002.
[6] D. Crisan and S. Ghazali, *On the convergence rates of a general class of weak approximations of sdes*, Stochastic Differential Equations: Theory and Applications, 2, 221–248, 2006.
[7] D. Crisan and T. Lyons, *Minimal entropy approximations and optimal algorithms for the filtering problem*, Monte Carlo Methods and Applications, 8, 343–356, 2002.
[8] D. Crisan and S. Ortiz-Latorre, *A kusuoka–lyons–victoir particle filter*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science, 469, 20130076, 2013.
[9] D. Crisan and B. Rozovskii, *The Oxford Handbook of Nonlinear Filtering*, Oxford University Press, 2011.
[10] P. Del Moral, *Mean field simulation for Monte Carlo integration*, Chapman & Hall: London, 2013.
[11] P. Del Moral and F.K. Formulae, *Genealogical and Interacting Particle Systems with Applications, Probability and Its Applications*, Springer-Verlag, 2004.
[12] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, 2001.
[13] A. Doucet, S. Godsill, and C. Andrieu, *On sequential monte carlo sampling methods for Bayesian filtering*, Statistics and Computing, 10, 197–208, 2000.
[14] A. Doucet and A.M. Johansen, *A tutorial on particle filtering and smoothing: Fifteen years later*, Handbook of Nonlinear Filtering, 12, 656–704, 2009.
[15] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, Springer-Verlag, 2009.
[16] A. Gelb, *Applied Optimal Estimation*, MIT press, 1974.
[17] N. Gordon, D. Salmond, and A. Smith, *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*, in Radar and Signal Processing, IEE Proceedings F, IET, 140, 107–113, 1993.
[18] L. Gyurkó and T. Lyons, *Efficient and practical implementations of cubature on Wiener space*, in Stochastic Analysis 2010, Springer-Verlag, 73–111, 2011.
[19] A. Jazwinski, *Stochastic Processes and Filtering Theory*, San Diego, California: Mathematics in Science and Engineering, 64, 1970.
[20] R. Kalman et al., *A new approach to linear filtering and prediction problems*, Journal of Basic Engineering, 82, 35–45, 1960.
[21] R.E. Kalman and R.S. Bucy, *New results in linear filtering and prediction theory*, Journal of Basic Engineering, 83, 95–108, 1961.
[22] P. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, 23, 2011.
[23] H. Kushner, *Approximations to optimal nonlinear filters*, Automatic Control, IEEE Transactions on, 12, 546–556, 1967.
[24] S. Kusuoka, *Approximation of expectation of diffusion process and mathematical finance*, Advanced Studies in Pure Mathematics, 31, 147–165, 2001.
[25] S. Kusuoka, *Malliavin calculus revisited*, J. Math. Sci. Univ. Tokyo, 10, 261–277, 2003.
[26] S. Kusuoka, *Approximation of expectation of diffusion processes based on lie algebra and Malliavin calculus*, Advances in Mathematical Economics, Springer-Verlag, 69–83, 2004.
[27] S. Kusuoka and D. Stroock, *Applications of the malliavin calculus. iii*, Journal of the Faculty of Science, Univ. of Tokyo, 32(1), 1987.
[28] C. Litterer and T. Lyons, *Introducing cubature to filtering*, In Crisan, Dan and Rozovskii, Boris [9], 768-796, 2011.
[29] C. Litterer and T. Lyons, *High order recombination and an application to cubature on Wiener space*, The Annals of Applied Probability, 22, 1301–1327, 2012.
[30] J.S. Liu and R. Chen, *Sequential Monte Carlo methods for dynamic systems*, Journal of the American Statistical Association, 93, 1032–1044, 1998.
[31] T. Lyons and N. Victoir, *Cubature on Wiener space*, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 460, 169–198, 2004.
[32] G.M. Morton, *A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing*, International Business Machines Company, 1966.
[33] S. Ninomiya and N. Victoir, *Weak approximation of stochastic differential equations and appli-*

cation to derivative pricing, Applied Mathematical Finance, 15, 107–121, 2008.
[34] W. Pan, *Cubature and tarn option pricing*, unpublished.
[35] M.K. Pitt and N. Shephard, *Filtering via simulation: Auxiliary particle filters*, Journal of the American Statistical Association, 94, 590–599, 1999.
[36] M. Putinar, *A note on Tchakaloff's Theorem*, Proceedings of the American Mathematical Society, 125, 2409–2414, 1997.
[37] C. Reutenauer, *Free Lie algebras*, Handbook of algebra, 3, 887–903, 2003.
[38] M. Tchernychova, private communication.
[39] G. Uhlenbeck and L. Ornstein, *On the theory of the Brownian motion*, Physical Review, 36, 823, 1930.
[40] P. van Leeuwen, *Nonlinear data assimilation in geosciences: an extremely efficient particle filter*, Quarterly Journal of the Royal Meteorological Society, 136, 1991–1999, 2010.
[41] N. Watanabe and N. Ikeda, *Stochastic Differential Equations and Diffusion Processes*, North Holland Math. Library/Kodansha, Amsterdam/Tokyo, 1981.