



HAL
open science

Comparative survey on non linear filtering methods: the quantization and the particle filtering approaches

Afef Sellami

► **To cite this version:**

Afef Sellami. Comparative survey on non linear filtering methods: the quantization and the particle filtering approaches. 2005. hal-00012274v2

HAL Id: hal-00012274

<https://hal.science/hal-00012274v2>

Preprint submitted on 20 Oct 2005

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparative survey on non linear filtering methods : the quantization and the particle filtering approaches

Afef SELLAMI

Laboratoire de Probabilités et
Modèles Aléatoires
CNRS, UMR 7599
and Université Paris Dauphine
sellami@math.jussieu.fr

October 2005

Abstract

We rise a comparative study between two different approaches to construct non linear filter estimators : on one hand grid methods using zero order and first order quantization schemes, on the other hand particle filtering algorithms using sequential importance sampling or resampling. For each method, numerical implementation is explicited in addition to convergence arguments and algorithmic complexity. Numerical examples are then given over three state space models: the Kalman filter case, the canonical stochastic volatility model and the infinite dimension explicit filter introduced in [8]

Key words: Nonlinear filter, quantization schemes, particle filtering, importance sampling, Kalman filter, stochastic volatility, infinite dimension filter.

1 Introduction

We consider a fixed discrete horizon $n \in \mathbb{N}^*$ and some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. A signal process is an \mathbb{R}^d -valued discrete time hidden Markov chain $(X_k)_{0 \leq k \leq n}$ evolving according to some signal equation:

$$X_{k+1} = F_{k+1}(X_k, \varepsilon_{k+1}), \quad 0 \leq k \leq n-1, \quad (1.1)$$

where $F_k : \mathbb{R}^d \times \mathbb{R}^q \rightarrow \mathbb{R}^d$, is a Borel function and $(\varepsilon_k)_{1 < k \leq n}$ is a sequence of independent identically distributed (iid) \mathbb{R}^q -valued random variables, independent of X_0 . The distribution μ_0 of X_0 is supposed to be known. Furthermore, $\mathbf{P}_k(x, dx')$ will denote the

probability transition of X_k , and:

$$\mu_0 f = \int_{\mathbb{R}^d} f(x) \mu_0(dx) \quad \text{and} \quad \mathbf{P}_k f(x) = \int_{\mathbb{R}^d} f(x') \mathbf{P}_k(x, dx').$$

At each time step k , an $\mathbb{R}^{d'}$ -valued noisy observation Y_k is made. The $(Y_k)_{0 \leq k \leq n}$ dynamics are driven by a Borel function $G_k : \mathbb{R}^{d'} \times \mathbb{R}^d \times \mathbb{R}^{q'} \rightarrow \mathbb{R}^{d'}$ such that

$$Y_k = G_k(Y_{k-1}, X_k, \eta_k), \quad 1 \leq k \leq n, \quad (1.2)$$

where (η_k) is a sequence of iid $\mathbb{R}^{q'}$ -valued random variables, independent of $\sigma(X_0, \varepsilon_k, k \geq 1)$.

We assume for convenience, that $Y_0 = 0$ and that, for every $1 \leq k \leq n$, the distribution of Y_k given X_k and Y_{k-1} admits a continuous conditional pdf $y \mapsto g_k(Y_{k-1}, X_k, y)$. Furthermore, we assume that g_k satisfies the following Lipschitz assumption:

$$\forall x, x' \in \mathbb{R}^d, \quad \forall y, y' \in \mathbb{R}^{d'},$$

$$|g_k(y, x, y') - g_k(y, x', y')| \leq [g_k]_{Lip}^{y, y'} |x - x'| \quad \text{and} \quad L^{y, y'} = \max_{0 \leq k \leq n} \sup_{x \in \mathbb{R}^d} |g_k(y, x, y')| < +\infty.$$

Remark 1.1 As the observation process is fixed, we will drop the dependency of $[g_k]_{Lip}^{y, y'}$ and $L^{y, y'}$ in (y, y') for notational convenience.

The problem we aim to solve is to compute

$$\Pi_n f = \mathbb{E}[f(X_n) | Y_1 = y_1, \dots, Y_n = y_n],$$

for an appropriate Borel function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ and a given observation sequence $y_{1:n} = \{y_1, \dots, y_n\}$. Throughout the paper, we will use capital letters to denote random variables and small letters to designate their realizations. We will also denote by $x \mapsto p(x|y)$ the conditional pdf of a random variable X given that $Y = y$.

1.1 Sequential definition

In order to implement some numerical schemes solving the filtering problem, it is important to see that a recursive formulation of the problem is possible owing to the Markov property of both the signal process (X_k) and the pair signal-observation (X_k, Y_k) . To establish it we set $p(x_k | y_{1:k-1})$ and $p(x_k | y_{1:k})$ respectively the prediction and the filtering pdf, for $1 \leq k \leq n$. Then for a test function f with sufficient regularity properties, we have:

$$\begin{aligned} \mathbb{E}[f(X_k) | Y_1 = y_1, \dots, Y_{k-1} = y_{k-1}] &= \int f(x_k) p(x_k | y_{1:k-1}) dx_k, \\ \mathbb{E}[f(X_k) | Y_1 = y_1, \dots, Y_k = y_k] &= \int f(x_k) p(x_k | y_{1:k}) dx_k. \end{aligned}$$

Using the Chapman-Kolmogorov formula and the Bayes formula, we can establish the following recursion:

Prediction

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}. \quad (1.3)$$

Updating/Correction

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k, y_{1:k-1})p(x_k|y_{1:k-1})}{\int p(y_k|x_k, y_{1:k-1})p(x_k|y_{1:k-1})dx_k} = \frac{g_k(y_{k-1}, x_k, y_k)p(x_k|y_{1:k-1})}{\int g_k(y_{k-1}, x_k, y_k)p(x_k|y_{1:k-1})dx_k}. \quad (1.4)$$

Denoting by α a generic constant depending on the fixed observations, we can derive by induction from (1.4) that:

$$p(x_n|y_{1:n}) = \alpha \int \prod_{k=1}^{n-1} g_k(y_{k-1}, x_k, y_k)p(x_k|x_{k-1})dx_k \mu_0(dx_0). \quad (1.5)$$

Then, it easily derives from the Markov property that:

$$\int f(x_n)p(x_n|y_{1:n})dx_n = \alpha \mathbb{E}[f(X_n) \prod_{k=1}^{n-1} g_k(y_{k-1}, X_k, y_k)].$$

This suggests to consider the so called unnormalized filter π_n defined by:

$$\pi_n f = \mathbb{E}[f(X_n) \prod_{k=1}^n g_k(y_{k-1}, X_k, y_k)].$$

Setting $f \equiv 1$ shows that the normalization constant equals $\frac{1}{\pi_n \mathbf{1}}$. Finally one gets

$$\Pi_n f = \frac{\pi_n f}{\pi_n \mathbf{1}}.$$

Remark 1.2 For convenience, the dependency of Π_n and π_n in the observation process has been omitted, as it is fixed. For the same reason, we will denote $g_k(x) := g_k(y_{k-1}, x, y_k)$ for $1 \leq k \leq n$, and $g_0 := \mathbf{1}$.

Prediction equation (1.3) and update equation (1.4) can merge to write a one step transition equation, linking $p(x_k|y_{1:k})$ to $p(x_{k-1}|y_{1:k-1})$ (and consequently the intermediate filters π_k to π_{k-1}).

$$p(x_k|y_{1:k}) = \alpha p(x_{k-1}|y_{1:k-1})p(y_k|x_k, y_{k-1})p(x_k|x_{k-1}) \quad (1.6)$$

By introducing the operators $(H_k)_{0 \leq k \leq n}$ defined below, a sequential definition of the unnormalized filter π_n can be given.

Namely, if one defines, for any $x \in \mathbb{R}^d$:

$$\begin{cases} H_k f(x) = g_k(X_k) \mathbb{E}[f(X_{k+1})|X_k = x], & 0 \leq k \leq n-1, \\ H_n^n f(x) = g_n(x) f(x), \end{cases} \quad (1.7)$$

then we have

$$\pi_n f = \mu_0 \circ H_0 \cdots \circ H_n^n f. \quad (1.8)$$

Consequently, we can write sequentially, either in the forward way:

$$U_0 = \mu_0 \circ H_0, \quad U_k = U_{k-1} \circ H_k, \quad 1 \leq k \leq n-1, \quad (1.9)$$

or in the backward way:

$$R_n = H_n^n, \quad R_k = H_k \circ R_{k+1}, \quad 0 \leq k \leq n-1, \quad (1.10)$$

so that $\pi_n f = \mu_0 R_0 f = U_{n-1} \circ H_n^n f$.

As we can see, the operators involved in the sequential definition of the filter have to be estimated numerically. Both methods we will present and compare below follow two different known approaches to approximate a conditional expectation. The first one, the quantization filtering method, is a grid method to transform an expectation into a finite weighted sum. These grids and their *companion* weights can be pre-computed and stored off line. As concerns filtering, this method has been introduced in [14], where the authors use optimal quantizers to construct what we will call a zero order quantization filter. Further developments of this method can be found in [16], where the stationary property of optimal quantizers is used to develop some schemes based on first order approximations. In section 2, we recall the construction of these schemes, as well as the most important convergence results. Section 3 deals with the second filtering approach we will be interested in. It is a Monte Carlo particle method using importance sampling to simulate random grids. This method has been deeply developed by practitioners of filtering in different application fields (see [7]). Two algorithms will be focused on, the sequential importance sampling (**SIS**) and the sequential importance resampling (**SIR**). Numerical experiments have been made on three state models presented in section 4, details on tests and numerical results are then depicted in section 5.

2 Quantization based filters

2.1 Zero order scheme

A process quantizer size $(N_k)_{0 \leq k \leq n}$ being fixed, and a quantizer Γ_k being precomputed, we set $(\hat{X}_k)_{0 \leq k \leq n}$ a marginal quantization of $(X_k)_{0 \leq k \leq n}$ defined by:

$$\hat{X}_k = Proj_{\Gamma_k}(X_k).$$

Quantization filters are constructed using recursive schemes. The zero order scheme introduced in [14] uses quantizers to approximate operators H_k by piecewise constant operators defined on the grids Γ_k . Namely, we define:

$$\begin{cases} \hat{H}_k f(\hat{X}_k) = g_k(\hat{X}_k) \mathbb{E}[f(\hat{X}_{k+1}) | \hat{X}_k], & 0 \leq k \leq n-1, \\ \hat{H}_n^n f(\hat{X}_n) = H_n^n f(\hat{X}_n) = g_n(\hat{X}_n) f(\hat{X}_n). \end{cases} \quad (2.1)$$

So, defining $\hat{\mu}_0$ as the (discrete) distribution of \hat{X}_0 , we have respectively the following forward and backward iterative approximation schemes :

$$\hat{U}_0 = \hat{\mu}_0 \hat{H}_0, \quad \hat{U}_k = \hat{U}_{k-1} \circ \hat{H}_k, \quad 1 \leq k \leq n-1, \quad (2.2)$$

and

$$\hat{R}_n = H_n^n, \quad \hat{R}_k = \hat{H}_k \circ \hat{R}_{k+1}, \quad 0 \leq k \leq n-1, \quad (2.3)$$

so that

$$\hat{\pi}_n f = \hat{\mu}_0 \hat{R}_0 f = \hat{U}_{n-1} \circ H_n^n f.$$

This formulation of the zero order scheme provides an implementable solution to the non linear filtering problem. This scheme consists of a recursive procedure based on weighted sums involving the optimal quantization grids and some weights, which only depend on the signal distribution. These quantities are usually computed off line, since they do not depend on the observation process.

To be more specific, we denote $\Gamma_k = \{x_k^1, \dots, x_k^{N_k}\}$ and $p_k^{ij} = \mathbb{E}[\mathbf{1}_{\{\hat{X}_{k+1}=x_{k+1}^j\}} | \hat{X}_k = x_k^i] \in \mathbb{R}$, then the implemented forward algorithm writes as Algorithm 0.

Algorithm 0 Zero order quantization based algorithm

$k = 0$	$1 \leq i \leq N_0$ and $1 \leq j \leq N_1$ $\hat{U}_0(j) = \sum_{i=1}^{N_0} \hat{\mu}_0^i p_0^{ij}$
$0 \leq k \leq n-2$	$1 \leq i \leq N_{k+1}$ and $1 \leq j \leq N_{k+2}$ $\hat{H}_{k+1}(i, j) = g_{k+1}(x_{k+1}^j) p_{k+1}^{ij}$ $\hat{U}_{k+1}(j) = \sum_{i=1}^{N_k} \hat{H}_{k+1}(i, j) \hat{U}_k(i)$
$k = n$	$1 \leq j \leq N_n$ $H_n^n f(x_n^j) = g_n(x_n^j) f(x_n^j)$ $\hat{\pi}_n f = \sum_{j=1}^{N_n} \hat{U}_{n-1}(j) H_n^n f(x_n^j)$

Note that $(\hat{\mu}_0^i)_{1 \leq i \leq N_0}$ designates the vector of discrete probabilities : $\hat{\mu}_0^i = \mathbb{P}(\hat{X}_0 \in \mathbf{C}_i(\Gamma_0))$. The use of the backward formulation (2.3) associated to the L^2 -optimal marginal (N_k) -quantizations of (X_k) , allows to establish a convergence rate to zero for the error estimate, for a Borel test function f satisfying $H_n^n f$ bounded Lipschitz continuous, when the quantizer size $N = \sum_{k=0}^n N_k$ goes to $+\infty$. Namely,

$$|\pi_n f - \hat{\pi}_n f| = O(N^{-1/d}),$$

(see [14]), with a constant depending on the observation vector and on the time horizon n .

2.2 First order schemes

In order to obtain better convergence rates, we introduce first order schemes (see [4, 15] and [16]). The main idea to develop them is to write a piecewise linear approximation

of $R_k f$, mimicking some first order Taylor expansion (see [4, 15]) and using differential terms to introduce first order correctors. The generic first order scheme could be written as follows:

$$\begin{cases} \widehat{R}_n f(\widehat{X}_n) &= H_n^n f(\widehat{X}_n), \\ \widehat{R}_k f(x_k^i) &= g_k(x_k^i) \mathbb{E}[\widehat{R}_{k+1}(\widehat{X}_{k+1}) + \langle \widehat{DR}_{k+1} f, \Delta_{k+1} \rangle | \widehat{X}_k = x_k^i], \\ &0 \leq k \leq n-1. \end{cases} \quad (2.4)$$

where $\widehat{DR}_{k+1} f$ is a numerical approximation of $DR_{k+1} f$ when it exists and $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product in \mathbb{R}^d .

A first order quantization based unnormalized filter estimator, is then defined by:

$$\widehat{\pi}_n f = \widehat{\mu}_0 \widehat{R}_0 f.$$

The way the approximation $\widehat{DR}_{k+1} f$ is defined leads to several variants. To establish error estimates, one makes an extensive use of the stationarity property of L^2 -optimal quantizers i.e.

$$\mathbb{E}[X_k | \widehat{X}_k] = \widehat{X}_k. \quad (2.5)$$

In the following, we will present the two schemes constructed in [16]: the one step recursive scheme based on a recursive definition of the differential term estimator \widehat{DR}_k and the two step recursive scheme based on an integration by part transformation of conditional expectation derivative $\mathbf{DP}_k R_{k+1}$.

To establish such first order scheme, we need F_k to be differentiable and the test function f to satisfy $H_n^n f$ differentiable with continuous Lipschitz bounded derivatives (in [16], we see that we can relax this assumption in case of regularizing semi-groups \mathbf{P}_k).

2.2.1 One step recursive first order scheme

The recursive definition of the differential term estimator is given in [16] by:

$$\begin{cases} \widehat{DR}_n(\widehat{X}_n) f &= DH_n^n f(\widehat{X}_n), \\ \widehat{DR}_k f(\widehat{X}_k) &= Dg_k(\widehat{X}_k) \mathbb{E}[\widehat{R}_{k+1} f(\widehat{X}_{k+1}) | \widehat{X}_k] + g_k(\widehat{X}_k) \mathbb{E}[\partial_x F_k(X_k, \varepsilon_{k+1})' \widehat{DR}_{k+1} f(\widehat{X}_{k+1}) | \widehat{X}_k] \\ &k = 0, \dots, n-1. \end{cases} \quad (2.6)$$

where $\partial_x F_k(X_k, \varepsilon_{k+1})'$ is the matrix transpose of $\partial_x F_k(X_k, \varepsilon_{k+1})$. Combined with (2.6), scheme (2.4) is completely computable, as it can be rewritten easily using finite weighted sums. The first order corrector terms introduce new companion parameters γ_k^{ij} , and δ_k^{ij} defined by:

$$\begin{aligned} \gamma_k^{ij} &= \mathbb{E}[\partial_x F_k(X_k, \varepsilon_{k+1})' \mathbf{1}_{\{\widehat{X}_{k+1} = x_{k+1}^j\}} | \widehat{X}_k = x_k^i] \in \mathcal{M}_d(\mathbb{R}), \\ \delta_k^{ij} &= \mathbb{E}[\Delta_{k+1} \mathbf{1}_{\{\widehat{X}_{k+1} = x_{k+1}^j\}} | \widehat{X}_k = x_k^i], \\ &= \mathbb{E}[(X_{k+1} - x_{k+1}^j) \mathbf{1}_{\{\widehat{X}_{k+1} = x_{k+1}^j\}} | \widehat{X}_k = x_k^i] \in \mathbb{R}^d, \end{aligned} \quad (2.7)$$

which can be computed off-line during the quantization operation. The on line computation cost is reduced to the computation of \widehat{R}_k , \widehat{R}_k and \widehat{DR}_k operators.

Moreover, the resulting scheme can be reformulated in a forward way, which allows to consider problems with intermediate filtering dates or multiple test functions without adding heavy computations. The implemented algorithm, Algorithm 1, uses a forward formulation based on matricial operators (see [16] for details).

FORWARD EXPRESSION

$$\widehat{U}_0 = \widehat{\mu}_0 \widehat{\mathcal{H}}_0 \quad \text{and} \quad \widehat{U}_k = \widehat{U}_{k-1} \widehat{\mathcal{H}}_k \quad k = 1, \dots, n-1,$$

where $\widehat{\mathcal{H}}_k$ is a lower triangular operator matrix defined by $\widehat{\mathcal{H}}_k = \begin{pmatrix} \widehat{H}_k^1 & 0 & 0 \\ \widehat{H}_k^2 & \widehat{H}_k^3 & 0 \\ 0 & \widehat{H}_k^4 & \widehat{H}_k^1 \end{pmatrix}$.

Using some additionnal assumptions on \mathbf{P}_k , g_k and F_k (see [16]), we establish for L^2 -optimal N_k -quantizers of (X_k) , that:

$$|\pi_n f - \widehat{\pi}_n f| = O(N^{-\frac{2}{d}}).$$

2.2.2 Two step recursive first order scheme

This method is based on an integration by part formula (see [4, 5, 2]) which leads to write $\mathbf{DP}_k R_{k+1} f$ as a weighted conditional expectation, namely:

$$\mathbf{DP}_k R_{k+1} f(x) = \mathbb{E}[R_{k+1} f(X_{k+1}) \Psi_k(X_k, \varepsilon_{k+1}) | X_k = x],$$

where $\Psi_k(X_k, \varepsilon_{k+1}) = (\Psi^i(x, \varepsilon))_{0 \leq i \leq d}$ and

$$\begin{aligned} \Psi^i : \mathbb{R}^d \times \mathbb{R}^d &\rightarrow \mathbb{R} \\ (x, \varepsilon) &\mapsto \operatorname{div} (\partial_\varepsilon F(x, \varepsilon)^{-1} \partial_x F(x, \varepsilon))^i + \frac{1}{\mathbf{p}(\varepsilon)} \langle (\partial_\varepsilon F(x, \varepsilon)^{-1} \partial_x F(x, \varepsilon))^i, \mathbf{Dp}(\varepsilon) \rangle \end{aligned}$$

$(\partial_\varepsilon F(x, \varepsilon)^{-1} \partial_x F(x, \varepsilon))^i$ designates the i^{th} line of the matrix $(\partial_\varepsilon F(x, \varepsilon)^{-1} \partial_x F(x, \varepsilon))$.

Note that some assumptions on both F_k and the signal innovation pdf \mathbf{p} are needed to allow such a transformation (see [16] for details). The resulting formula for \widehat{R}_k , $0 \leq k \leq n-1$, is consequently:

$$\begin{aligned} \widehat{R}_k f(\widehat{X}_k) &= g_k(\widehat{X}_k) \widehat{\mathbf{P}}_k \widehat{R}_{k+1} f(\widehat{X}_k) + g_k(\widehat{X}_k) \times \\ &\left(\mathbb{E}[\langle \mathbf{D}g_{k+1}(\widehat{X}_{k+1}) \widehat{\mathbf{P}}_{k+1} \widehat{R}_{k+2} f(\widehat{X}_{k+1}), \Delta_{k+1} \rangle | \widehat{X}_k] - \mathbb{E}[\langle g_{k+1}(\widehat{X}_{k+1}) \times \right. \\ &\left. \mathbb{E}[\widehat{R}_{k+2} f(\widehat{X}_{k+2}) \Psi_{k+1}(X_{k+1}, \varepsilon_{k+2}) | \widehat{X}_{k+1}], \Delta_{k+1} | \widehat{X}_k] \right). \end{aligned} \quad (2.8)$$

This formula introduces a new weight λ_k^{ij} :

$$\lambda_k^{ij} = E[\Psi_k(X_k, \varepsilon_{k+1}) \mathbf{1}_{\{\widehat{X}_{k+1} = x_{k+1}^j\}} | \widehat{X}_k = x_k^i] \in \mathbb{R}^d$$

Algorithm 1 One step recursive 1st order quantization based filtering scheme

$$\begin{array}{ll}
 k = 0 & 1 \leq i \leq N_0 \text{ and } 1 \leq j \leq N_1 \\
 & \hat{U}_0^{11}(j) = \sum_{i=1}^{N_0} \mu_0^i p_0^{ij} \qquad \hat{U}_0^{21}(j) = 0 \in \mathbb{R}^d \\
 & \hat{U}_0^{22}(j) = \sum_{i=1}^{N_0} \mu_0^i \gamma_0^{ij} \qquad \hat{U}_0^{31}(j) = 0 \in \mathbb{R} \\
 & \hat{U}_0^{32}(j) = \sum_{i=1}^{N_0} \mu_0^i \delta_0^{ij} \qquad \hat{U}_0^{33}(j) = \hat{U}_0^{11}(j) \\
 \\
 0 \leq k \leq n-2 & 1 \leq i \leq N_{k+1} \text{ and } 1 \leq j \leq N_{k+2} \\
 & \hat{H}_{k+1}^1(i, j) = g_{k+1}(x_{k+1}^i) p_{k+1}^{ij} \qquad \hat{H}_{k+1}^2(i, j) = Dg_{k+1}(x_{k+1}^i) p_{k+1}^{ij} \\
 & \hat{H}_{k+1}^3(i, j) = g_{k+1}(x_{k+1}^i) \gamma_{k+1}^{ij} \qquad \hat{H}_{k+1}^4(i, j) = g_{k+1}(x_{k+1}^i) \delta_{k+1}^{ij} \\
 \\
 & \hat{U}_{k+1}^{11}(j) = \sum_{i=1}^{N_{k+1}} \hat{U}_k^{11}(i) \hat{H}_{k+1}^1(i, j) \\
 & \hat{U}_{k+1}^{21}(j) = \sum_{i=1}^{N_{k+1}} \hat{U}_k^{21}(i) \hat{H}_{k+1}^1(i, j) + \hat{U}_k^{22}(i) \hat{H}_{k+1}^2(i, j) \\
 & \hat{U}_{k+1}^{22}(j) = \sum_{i=1}^{N_{k+1}} \hat{U}_k^{22}(i) \hat{H}_{k+1}^3(i, j) \\
 & \hat{U}_{k+1}^{31}(j) = \sum_{i=1}^{N_{k+1}} \hat{U}_k^{31}(i) \hat{H}_{k+1}^1(i, j) + \langle \hat{U}_k^{32}(i), \hat{H}_{k+1}^2(i, j) \rangle \\
 & \hat{U}_{k+1}^{32}(j) = \sum_{i=1}^{N_{k+1}} \hat{H}_{k+1}^3(i, j) \hat{U}_k^{32}(i) + \hat{U}_k^{33}(i) \hat{H}_{k+1}^4(i, j) \\
 & \hat{U}_{k+1}^{33}(j) = \sum_{i=1}^{N_{k+1}} \hat{U}_k^{33}(i) \hat{H}_{k+1}^1(i, j) \\
 \\
 k = n & H_n^n f(x_n^j) = g_n(x_n^j) f(x_n^j) \qquad H_n^n f(x_n^j) = g_n(x_n^j) f(x_n^j) \\
 & \hat{\pi}_n f = \sum_{j=1}^{N_n} (\hat{U}_{n-1}^{33}(j) + \hat{U}_{n-1}^{31}(j)) H_n^n f(x_n^j) + \langle \hat{U}_{n-1}^{32}(j), D H_n^n f(x_n^j) \rangle
 \end{array}$$

Depending only on the signal parameters, it can be precomputed like p_k^{ij} and δ_k^{ij} and kept off line in accessible tables. As for the previous first order scheme, a forward transcription of the scheme is possible using finite weighted sums which gives the following implemented algorithm, Algorithm 2.

Under some additional assumptions on the weight function Ψ_k , we insure a convergence rate of order $N^{-\frac{2\rho}{d}}$ where $\rho \in]\frac{1}{2}, 1)$ (see [16]).

Algorithm 2 Two step recursive 1st order quantization based filtering scheme

$$0 \leq k \leq n-2 \quad 1 \leq i \leq N_k, 1 \leq j \leq N_{k+1} \text{ and } 1 \leq l \leq N_{k+2} \quad (g_0 := \mathbf{1})$$

$$\begin{aligned} \hat{H}_k^1(i, j) &= g_k(x_k^j) p_k^{ij} \\ \hat{H}_k^2(i, l) &= g_k(x_n^i) \sum_{j=1}^{N_{k+1}} \langle p_{k+1}^{jl} Dg_{k+1}(x_{k+1}^j) - g_{k+1}(x_{k+1}^j) \lambda_{k+1}^{jl}, \delta_k^{ij} \rangle \end{aligned}$$

Initial $1 \leq j \leq N_1$ and $1 \leq l \leq N_2$

$$\begin{aligned} \hat{U}_0^1(j) &= \sum_{i=0}^{N_0} \hat{\mu}_0^i \hat{H}_0^1(i, j) \\ \hat{U}_0^2(l) &= \sum_{i=1}^{N_k} \hat{\mu}_0^i \hat{H}_0^2(i, l) \end{aligned}$$

Transition $1 \leq k \leq n-2, 1 \leq j \leq N_{k+1}$ and $1 \leq l \leq N_{k+2}$

$$\begin{aligned} \hat{U}_{k+2}^2(l) &= \sum_{i=1}^{N_k} \hat{H}_k^2(i, l) \hat{U}_k^1(i) + \sum_{j=1}^{N_{k+1}} \hat{H}_{k+1}^1(j, l) \hat{U}_{k+1}^2(j) \\ \hat{U}_k^1(j) &= \sum_{i=1}^{N_k} \hat{H}_k^1(i, j) \hat{U}_{k-1}^1(i) \end{aligned}$$

Final

$$\begin{aligned} H_n^n f(x_n^j) &= g_n(x_n^j) f(x_n^j) \quad DH_n^n f(x_n^j) = f(x_n^j) Dg_n(x_n^j) + g_n(x_n^j) Df(x_n^j) \\ \hat{\pi}_n f &= \sum_{j=1}^{N_n} \hat{U}_n^2(j) H_n^n f(x_n^j) + \\ &\quad \sum_{i=1}^{N_{n-1}} \hat{U}_{n-2}^1(i) g_{n-1}(x_{n-1}^i) \sum_{j=1}^{N_n} \left(H_n^n f(x_n^j) p_{n-1}^{ij} + \langle DH_n^n f(x_n^j), \delta_{n-1}^{ij} \rangle \right) \end{aligned}$$

3 Particle filters

In this section, we provide some background on two classical Monte Carlo particle methods for non linear filters. Based on online simulations, this approach provides, like quantization based methods do, a reformulation of conditional expectations as finite weighted sums. This time the sum terms are random variables. The following subsections deal with two basic particle filtering methods using an importance sampling technique.

3.1 Sequential Importance Sampling (SIS)

Suppose we are able to simulate N independent random vectors $(X^i)_{1 \leq i \leq N}$, distributed according to $\mathcal{L}(X_n | Y_1 = y_1, \dots, Y_n = y_n)$. An estimator $\Pi_n^N f$ of the filter would be:

$$\Pi_n^N f = \frac{1}{N} \sum_{i=1}^N f(X^i).$$

In other words, from a quantization viewpoint, we approximate the *a posteriori* distribution by the set of random samples X^i with weights $\frac{1}{N}$. This estimate is unbiased and its rate of convergence is ruled by central limit theorem (see e.g. [7]).

Unfortunately, the initial assumption that the distribution $\mathcal{L}(X_n | Y_1 = y_1, \dots, Y_n = y_n)$ is simulatable is usually not satisfied. This distribution is unknown as it is the one we aim to approximate by the filtering procedure. In simulation, this problem is often over-passed by applying an importance sampling procedure. Namely, we choose a simulatable

distribution $q(x_n|y_{1:n})$, that we will call *importance distribution* and write:

$$\Pi_n f = \int f(x_n) \omega(x_n) q(x_n|y_{0:n}) dx_n,$$

where $\omega(x) = \frac{p(x|y_{0:n})}{q(x|y_{1:n})}$ will be called the *importance weight* function. Now, if we designate by $(X^i)_{1 \leq i \leq N}$ N iid samples of distribution q , a new definition of the filter estimator can be given by:

$$\Pi_n^N f = \sum_{i=1}^N f(X^i) \omega(X^i).$$

We approximate the *a posteriori* distribution by the set of the N random samples (or particles) X^i and the associated importance weights. In our case, we always know a distribution which can be chosen as an importance distribution: that of the hidden signal (X_k) , so we define:

$$q(x_{0:n}|y_{0:n}) = p(x_{1:n}) = \mu_0(x_0) \prod_{k=1}^n p(x_k|x_{k-1}). \quad (3.1)$$

This choice gives the advantage of making possible a sequential approximation of the particles and weights $\{x_k^i, \omega_k(x_k^i)\}$ via simulation, by the use of the prediction and update equations (1.3) and (1.4). At each time step $0 \leq k \leq n$, we simulate an N -sample $(X_{0:n}^i)$ having q as a pdf. The filter estimator at each date $1 \leq k \leq N$ is then:

$$\Pi_k^{q,N} f = \sum_{i=1}^N f(X_k^i) \omega_k(X_{0:k}^i),$$

with the sequential update equation for the importance weights:

$$\begin{aligned} \omega_{k+1}(x_{0:k+1}^i) &= \alpha \frac{p(x_{k+1}^i|y_{1:k+1})}{p(x_{0:k+1}^i)} \\ &= \alpha \frac{p(x_k^i|y_{1:k})p(y_{k+1}|x_{k+1}^i, y_k)p(x_{k+1}^i|x_k^i)}{p(x_{k+1}^i|x_k^i)p(x_{0:k}^i)} \\ &= \alpha \omega_k(x_{0:k}^i) g_{k+1}(y_k, x_{k+1}^i, y_{k+1}) \end{aligned} \quad (3.2)$$

$$\text{and} \quad \omega_{k+1}(x_{0:k+1}^i) = \frac{\omega_k(x_{0:k}^i) g_{k+1}(y_k, x_{k+1}^i, y_{k+1})}{\sum_{i=1}^N \omega_k(x_k^i) g_{k+1}(y_k, x_{k+1}^i, y_{k+1})} \quad (3.3)$$

Hence, we can implement sequentially Algorithm 3.

The convergence rate of the method is independent of the signal dimension, since it is of Monte Carlo type. For each bounded continuous test function f , we have:

$$\mathbb{E}[|\Pi_{y,n} f - \Pi_{y,n}^{q,N} f|] \leq \frac{m_n}{\sqrt{N}} \|f\|_\infty. \quad (3.4)$$

where m_n is a constant depending on n and on the observation.

Algorithm 3 Algorithm SIS

- Simulate from the signal initial distribution $\{(x_0^i)_{1 \leq i \leq N}, \frac{1}{N}\}$
 - * At a given date $0 \leq k \leq n - 1$, we have $\{(x_k^i)_{1 \leq i \leq N}, (\omega_k)_{1 \leq i \leq N}\}$
 - Simulate particles $x_{k+1}^i \sim p(x_{k+1}|x_k^i)$
 - Update weights using equation (3.2) and (3.3), with the observation y_{k+1}
 - Go to * for date $k + 1$
-

3.2 Sequential Importance Resampling or Bootstrap filter (SIR)

The method described previously suffers from the problem of weight degeneracy, as the sequential definition implies to multiply repeatedly the likelihood terms. This produces sometimes particles with very small weights, that we carry out throughout the estimation although their contribution to the distribution description is negligible. This problem occurs for example when the importance distribution is badly adjusted to the filter distribution.

One solution to the degeneracy problem is introduced in [10, 1]. It suggests to diffuse sequentially equally weighted particles by adding a resampling step to the previous algorithm. This new method called *Sequential Importance Resampling (SIR)* or Bootstrap filter uses interaction between particles to eliminate weakly weighted ones. As an interaction phase is introduced, samples are no longer independent and Monte Carlo arguments cannot be used to establish a convergence rate like in the previous case. Nevertheless, in [7, 13, 6], it is shown that by the means of convenient resampling procedures, for example the multinomial sampling procedure, a convergence rate of type (3.4) can be established.

Algorithm 4 Algorithm SIR

- Simulate from the signal initial distribution $(x_0^i)_{1 \leq i \leq N}$
 - * At a given date $0 \leq k \leq n - 1$, we have $(x_k^i)_{1 \leq i \leq N}$
 - Simulate particles $x_{k+1}^i \sim p(x_{k+1}|x_k^i)$
 - Update weights using equations (3.2) and (3.3), with the observation y_{k+1}
 - Resampling step:
Simulate N samples i_{k+1}^j from the discrete multinomial distribution
with parameters $(\omega_{k+1}^i)_{1 \leq i \leq N}$
 - Go to * with date $k + 1$ and modified particles $(x_{k+1}^{i_{k+1}^j})_{1 \leq j \leq N}$
-

The choice of resampling procedures, also called branching methods, can be conditioned by variance reduction criteria, numerical complexity or convergence rate preserving [6, 7]. The common point they share is that they aim to obtain an *unweighted* empirical distribution to sequentially approximate intermediate filtering pdf. Namely, they achieve that $\mathbb{E}[\mathbf{1}_{\{i_k^j=i\}}] = \omega_k^i$.

Algorithm 0	C_0N^2
Algorithm 1	$C_1N^2d^3$
Algorithm 2	C_2N^3d
Algorithm 3	C_3N
Algorithm 4	C_4N

Table 1: Comparison of complexity degrees for different numerical filtering algorithms

3.3 Elements for a comparison

As mentioned above, the underlying principles of numerical methods in sequential non linear filtering are the same (see [12]). Both particle and quantization based methods use the approximation of the objective distribution by a finite state one, so that the final expression of the filter estimator appears as a finite weighted sum. The difference lies in the construction of such an approximation. For quantization filters, we use off line precomputed marginal distribution quantizers. For particle filters, the grids are random samples of the same distributions, that need to be computed on line.

Following this remark, we see that both approaches are similar, and we expect that they behave the same way when treating comparable state models and observations. However some differences deserve to be pointed out.

As a Monte Carlo method, particle filters give a random solution to the filtering problem. This is a point we do not have to manage when treating with quantization methods. Conversely, being based on Monte Carlo convergence arguments, particle methods do not suffer from dimension dependency when considering their theoretical convergence rate, whereas quantization based methods do depend on the dimension of the state space. Considering the theoretical convergence results, quantization methods are still competitive till dimension 2 for zero order schemes and till dimension 4 for first order ones.

From an algorithmic viewpoint, some more differences deserve to be mentionned. It is about the complexity of each algorithm, summerized in Table 1.

Owing to the off line computations of the quantizer grids, constants C_0 , C_1 and C_3 for quantization filters represent elementary operations computation cost. For particle filters, C_3 and C_4 include simulation cost, and could be dependent of N which results in more complex algorithms. This occurs for **SIR** algorithm with some particular resampling algorithms.

Finally, we should remark that quantization methods need smaller grid sizes than Monte Carlo methods to attain convergence regions. This will be pointed out in numerical results below. This fact, in some cases, compensates the relatively high complexity range of quantization methods, particularly in low signal dimensions.

4 State Equations

We aim to compare numerical performances of the two approaches. In this paragraph, we briefly present three models chosen to make up the benchmark. As a first step, we consider the Kalman filter, a typical case where an exact explicit solution can be computed. Then, we will consider a canonical stochastic volatility model, issued from the discretization of a diffusion process representing an asset price dynamics on a financial market. We will finally examine the nonlinear filter case introduced in [8] by V. Genon Catalot, for which some semi-closed forms of the solution are available. It can also be adapted to treat asset prices.

4.1 Kalman filter (KF)

For this model, both signal and observation equations are linear with Gaussian independent noises. It is well known that the filter in this case is a Gaussian process which parameters (the two first moments) can be computed sequentially by a deterministic algorithm (KF), (see e.g. [9] for details). We set:

$$\begin{cases} X_k = \rho X_{k-1} + \theta \varepsilon_{k+1}, & X_0 \sim \mathcal{N}(m_0, \Sigma'_0 \Sigma_0) \\ Y_k = X_k + \alpha \eta_k, \\ \varepsilon_k \text{ and } \eta_k \text{ iid } \sim \mathcal{N}(0, I_d), \\ \rho, \theta, \alpha \in \mathcal{M}_d(\mathbb{R}). \end{cases} \quad (4.1)$$

4.2 Canonical stochastic volatility model (SVM)

This is the time discretization of a continuous diffusion model commonly used in finance. The stock price S_t and its volatility σ_t solve the following stochastic differential system:

$$\begin{cases} dS_t = \frac{1}{2} \sigma_t^2 S_t dt + \sigma_t S_t dW_t^1, \\ d(\log(\sigma_t^2)) = -\lambda \log(\sigma_t^2) dt + \tau dW_t^2. \end{cases} \quad (4.2)$$

So, the Euler scheme with time step Δ writes:

$$\begin{cases} \log\left(\frac{S_{k+1}}{S_k}\right) = \sigma_k \sqrt{\Delta} \eta_k, \\ \log(\sigma_{k+1}^2) - \log(\sigma_k^2) = -\lambda \log(\sigma_k^2) \Delta + \tau \sqrt{\Delta} \varepsilon_{k+1}. \end{cases} \quad (4.3)$$

where η_k and ε_k are iid $\mathcal{N}(0, 1)$.

Now, setting $Y_k = \log\left(\frac{S_{k+1}}{S_k}\right)$ and $X_k = \log(\sigma_k^2)$ leads to the following discrete time state equations.

$$\begin{cases} X_k = \rho X_{k-1} + \theta \varepsilon_{k+1}, \\ Y_k = \exp\left(\frac{X_k}{2}\right) \eta_k, \\ \varepsilon_k \text{ and } \eta_k \text{ iid } \sim \mathcal{N}(0, 1), \\ \rho = (1 - \lambda \Delta) \text{ and } \theta = \tau \sqrt{\Delta} \in \mathbb{R}_+^*. \end{cases} \quad (4.4)$$

When $\Delta \in (0, \frac{2}{\lambda})$ then $\rho = (1 - \lambda\Delta) \in (-1, 1)$ and (X_k) is a positively recurrent Markov chain which converges geometrically to its invariant distribution $\mathcal{N}(0, \frac{\theta^2}{1-\rho^2})$ (with respect to the total variation metric). The non linearity introduced in the observation equation makes impossible to determine exactly the filtering distribution. This is a case where numerical methods are necessary to solve the filtering problem.

4.3 Explicit non linear filter [8]

In this example, we treat a non linear non Gaussian state equation introduced by V. Genon Catalot in [8]. When both the noise distribution and the initial signal distribution are specified in an appropriate way, it is shown how to construct an infinite dimensional *explicit* non-linear filter, in the sense that all parameters of the a posteriori pdf can be determined by recursive explicit schemes.

For that we introduce the family of the so called *Serial Gaussian* distributions $SG(\sigma^2, (\alpha_i)_{i \geq 0})$ (see [8]) which probability density functions are defined by:

$$u \mapsto \left(\sum_{i \geq 0} \alpha_i \frac{u^{2i}}{\sigma^{2i} C_{2i}} \right) \frac{1}{\sigma \sqrt{2\pi}} \exp\left(-\frac{u^2}{2\sigma^2}\right),$$

where:

- $C_{2i} = \frac{(2i)!}{2^i i!}$ is the $2i^{th}$ moment of a $\mathcal{N}(0, 1)$ distribution.
- $\forall i \geq 0, \alpha_i \geq 0$ and $\sum_{i \geq 0} \alpha_i = 1$. We will denote $\alpha = (\alpha_i)_{i \geq 0}$.
- $\sigma > 0$.

$SG(\sigma^2, \alpha)$ can be seen as a mixture distribution with a scale parameter σ and a mixture parameter α . In a filtering context, considering the following state equations:

$$\begin{cases} X_k = \rho X_{k-1} + \theta \varepsilon_{k+1} \\ Y_k = X_k \eta_k \end{cases} \quad \text{where} \quad \begin{cases} \varepsilon_k & iid \sim \mathcal{N}(0, 1), \\ \eta_k & iid \sim \mathcal{B}(\pm 1, \frac{1}{2}) \times \sqrt{\mathcal{E}(\lambda)}, \\ X_0 & \sim SG(\sigma_0^2, (\alpha_i^0)_{i \geq 0}), \end{cases}$$

it is established that the filter distribution as well as the prediction one will be $SG(\sigma^2, (\alpha_i)_{i \geq 0})$, with parameters that can be evaluated sequentially. We will denote for $0 \leq k \leq n$:

$$\begin{aligned} \mathcal{L}(X_k | Y_{1:k}) &\sim SG(\hat{\sigma}_k^2, (\hat{\alpha}_i^k)_{i \geq 0}), \\ \mathcal{L}(X_{k+1} | Y_{1:k}) &\sim SG(\sigma_{k+1}^2, (\alpha_i^{k+1})_{i \geq 0}), \end{aligned}$$

where scale and mixture parameters are defined by Algorithm 5:

By construction, we see that starting from a special SG distribution, with finite number of mixture parameters leads to a finite dimension filter.

Algorithm 5 Explicit filter

Initial step : $\mathcal{L}(X_0) \sim SG(\sigma_0^2, (\alpha_i^0)_{i \geq 0})$ Transition from date k to $k + 1$:

Update: $\hat{\sigma}_k(Y_k) = \sigma_k \frac{|Y_k|}{\sqrt{Y_k^2 + 2\lambda\sigma_k^2}}$

$\hat{\alpha}_0^k(Y_k) = 0$

$\hat{\alpha}_i^k(Y_k) = \alpha_{i-1}^k \frac{f_{i-1}(\frac{Y_k}{\sqrt{\lambda}\sigma_k})}{\sum_{i \geq 0} \alpha_i^k f_i(\frac{Y_k}{\sqrt{\lambda}\sigma_k})}$ where $f_i(z) = \frac{(2i+1)z^{2i}}{(z^2+2)^{i+\frac{3}{2}}}$

Prediction: $\sigma_{k+1}^2 = \theta^2 + \rho^2 \hat{\sigma}_k^2$

$\alpha_i^{k+1} = \left(\frac{\rho \hat{\sigma}_k}{\sigma_{k+1}}\right)^{2i} \sum_{j \geq i} C_j^i \left(\frac{\theta}{\sigma_{k+1}}\right)^{2(i-j)} \hat{\alpha}_i^k$ where $C_j^i = \frac{j!}{i!(j-i)!}$

Finally, we note that the conditional pdf of the observation Y_k given the signal X_k , is independent of the past observations and writes as:

$$g_k(x, y) = \frac{\lambda x^2}{|y|^3} \exp\left(-\frac{\lambda x^2}{y^2}\right).$$

It satisfies the conditions to construct and establish the quantization filter convergence rates (see [16]).

5 Numerical experiments

5.1 Stationary suboptimal quantizers

Quantization based filters use precomputed quantizer grids and companion parameters. Although this preprocessing procedure is done off line, it is worth noting that in some cases, we can recycle optimal quantizers of standard distributions to construct stationary suboptimal quantizers that preserve the announced convergence rates. In fact, suppose there exists a sequence of affine invertible functions $T_k : \mathbb{R}^d \rightarrow \mathbb{R}^d$, such that $X_k = T_k(Z)$, for all $0 \leq k \leq n$, Z being a known distribution. Taking $\Gamma = \{z^1, \dots, z^N\}$ an L^2 -optimal N -quantizer of Z , we can define a stationary N -quantizer of X_k by setting:

$$\hat{X}_k = \sum_{i=1}^N T_k(z^i) \mathbf{1}_{\{T_k^{-1}(X_k) \in \mathbf{C}_i(\Gamma)\}} = T_k(\hat{Z}). \quad (5.1)$$

This quantizer is no more optimal (in fact it is not even a Voronoi quantizer), but, since the T_k are assumed to be affine, \hat{X}_k still satisfies the stationary property. That is the property needed to establish first order scheme theoretical convergence rate to zero. Namely, we have :

$$\mathbb{E}[X_k | \hat{X}_k] = T_k(\mathbb{E}[Z | \hat{Z}]) = T_k(\hat{Z}) = \hat{X}_k.$$

Furthermore, even if in this case \hat{X}_k is suboptimal, owing to the optimality of \hat{Z} , and to the Zador Theorem (see [3, 11]), we have $\|X_k - \hat{X}_k\|_2 = O(N^{-\frac{1}{d}})$. In fact we have:

$$\|X_k - \hat{X}_k\|_2 = \|T_k(Z) - T_k(\hat{Z})\|_2 \leq \|T_k\| \|Z - \hat{Z}\|_2.$$

This will be particularly interesting when we will consider for example linear signal dynamics with Gaussian innovations and Gaussian initial distribution. This is the case for Kalman filter or for the canonical stochastic volatility model (see (4.1) and (4.4)). It is also the case for particular explicit filter model, when the initial distribution is Gaussian (i.e. $SG(\sigma_0^2, (\alpha_i^0)_{i \geq 0})$ where $\alpha_0 = 1$ and $\alpha_i = 0$ for all $i > 0$). In these cases, X_k is Gaussian at all dates and we can drastically reduce the offline computation runtime needed to implement the quantization based filters since quantizer grids are obtained from the $\mathcal{N}(0, I_d)$ ones by an affine transformation.

Another important case of interesting preprocessing optimization is the case of stationary signal processes : the transition companion parameters do not depend on the time step considered, so only one transition is needed. Namey, $\mathbb{P}(\hat{X}_{k+1} = x^j | \hat{X}_k = x^i)$ is the same on the unique grid $\Gamma = \Gamma_0 = \{x^1, \dots, x^N\}$ quantizing the stationnary distribution.

This is even more interesting, as it can be associated to the previous point. Assume for example that for any $0 \leq k \leq n$, we have $X_k \sim \mathcal{N}(0, \Sigma' \Sigma)$ where Σ satisfies $\Sigma' \Sigma = (\rho \Sigma)' \rho \Sigma + \theta' \theta$. In this particular case, we could first compute¹ Γ an L^2 -optimal quantizer of the centered reduced Gaussian distribution. The quantizers Γ_k are then deduced by the expansion $\Gamma_k = \Sigma \times \Gamma_0$ coupled with (5.1), the companion parameters stored off line are those of one single transition as they are time independant.

5.2 Convergence tests

We select the following three test functions:

$$f_1(x) = x, \quad f_2(x) = |x|^2, \quad f_3(x) = \exp(-|x|).$$

For quantization based methods, as the schemes are deterministic, we simply need to study the behavior of the filter estimators (or dierctly the errors when a reference value is available), as the total size N of the grids goes to infinity.

For particle filtering methods, as the estimators are random variables, the testing approach is slightly different. In fact, we need to represent the empirical mean as well as a variance estimate as functions of the particle number N . For that purpose, we simulate a large number M of realizations of the filter random estimator, and represent the empirical confidence interval containing 90% of the observed values. This is achieved by computing the 5% and the 95% centiles over the population of the M filter random estimator realizations. These values will be represented as functions of the particle number N and the scheme performance is measured with respect to the confidence interval length.

¹Optimal quantizers for the Gaussian distribution can be downloaded on <http://www.proba.jussieu.fr/pageperso/pages/>

5.3 Results and comments

5.3.1 Kalman filter: d=1

True values are computed by the Kalman-Bucy filter algorithm. For particle methods, we represent a realization of the filter estimator with 5000 particles. For quantization based filters, the two first order schemes are compared to the zero order one with $N_k = 100$ for all $0 \leq k \leq n$ (see Table 2).

$(\rho, \theta, \alpha, \Sigma_0)$	(0.65,1.0,0.1,0.05)			(0.996,0.0316,0.0632,0.7)		
	$\hat{\Pi}_{y,10}f_1$	$\hat{\Pi}_{y,10}f_2$	$\hat{\Pi}_{y,10}f_3$	$\hat{\Pi}_{y,25}f_1$	$\hat{\Pi}_{y,25}f_2$	$\hat{\Pi}_{y,25}f_3$
KF (Ref. Values)	0.0776126	0.9165	0.011546	1.01053	0.3643086	1.02273
SIS (5000 pts)	0.0605323	0.923421	0.0102159	1.08403	0.339045	1.17991
SIR (5000 pts)	0.0781505	0.915884	0.0115783	1.01208	0.363763	1.02594
QF0 (100 pts)	0.077596	0.916493	0.0115486	1.016	0.36235	1.03397
QF1 1-step (100 pts)	0.077601	0.916491	0.0115493	1.0121	0.363741	1.02593
QF1 2-step (100 pts)	0.0776013	0.916491	0.0115493	1.0121	0.363744	1.02591

Table 2: One dimensional Kalman filter case.

5.3.2 Kalman filter: d=3

We still consider equation (4.1) with parameters:

$$\rho = \begin{pmatrix} 0.996 & 0 & 0 \\ 0 & 0.996 & 0 \\ 0 & 0 & 0.996 \end{pmatrix}, \theta = \begin{pmatrix} 0.02 & 0.02 & 0.01 \\ 0.02 & 0.06 & -0.01 \\ 0.01 & -0.01 & 0.04 \end{pmatrix} \quad \text{and} \quad \alpha = I_3.$$

The initial signal distribution is centered and Gaussian with covariance matrix:

$$\Sigma_0' \Sigma_0 = \begin{pmatrix} 0.11 & 0.23 & 4e-4 \\ 0.23 & 0.53 & -8e-4 \\ 4e-4 & -8e-4 & 0.0018 \end{pmatrix}.$$

The chosen prior distribution is the stationary one. If we denote $\Gamma = \{z^1, \dots, z^N\}$ the L^2 -optimal N -quantizer of a centered reduced Gaussian distribution. At $0 \leq k \leq N$, $X_k \sim \mathcal{N}(0, \Sigma_0' \Sigma_0)$ and according to (5.1) we define the marginal stationary (N_k) -quantizer of (X_k) as follows:

$$\hat{X}_k = \sum_{i=1}^N \Sigma_0 z^i \mathbf{1}_{\{X_k \in \Sigma_0 \mathbf{c}_i(\Gamma)\}}.$$

Although the quantization filter convergence rate and numerical complexity depend on the signal dimension d , it remains interesting to apply them for medium dimension signals. The chosen quantizers are not optimal but we obtain satisfactory convergence results.

Convergence errors are depicted in Figure 1 for f_2 and in Figure 2 for f_3 . From log-log scale regression in Figure 1, we can evaluate the convergence rate improvement. The Table 3 summarizes the computed slopes of the log-log regressions.

Or0	Or1 1-step	Or1 2-step
-0.34	-0.52	-0.81

Table 3: Regression slopes on the log-log scale representation (d=3)

We observe nearly the expected theoretical results. A convergence rate of $\frac{1}{d} \approx 0.33$ for zero order schemes, for first order schemes, the slopes are different from the theoretical one $\frac{2}{d} \approx 0.66$, but still better than the zero order slopes.

Figure 3 depicts confidence intervals over 5000 particle filter realizations and the reference values computed exactly for f_2 and approximated via a Monte Carlo estimation for f_3 using the known simulatable distribution of the filter. While **SIS** degenerates, **SIR** method gives satisfying results. Compared with the quantization filters, it is important to see that the range of N is different from one approach to another when the same error range is considered. The error reached by quantization filters for 800 points is farly less than the confidence interval length given by particle filters with 7000 $\gg \sqrt{800}$ particles.

5.3.3 Stochastic volatility model

In Figure 4 a comparison is made between particle filter methods over $M = 4000$ realizations and quantization based ones. We note the degeneracy of **SIS** method. **SIR** and quantization based methods converge to the same values. Furthermore, even for small quantizer sizes, we see that quantization based estimations always lie in the confidence interval of **SIR** computed for large particle sizes ($N = 10000$).

5.3.4 Non linear explicit filter

For numerical application, we considered the case where the a priori signal distribution is Gaussian. This is a particular serial Gaussian distribution $SG(\sigma^2, (\alpha_i)_{i \geq 0})$, where $\alpha_0 = 1$ and $\alpha_i = 0$ for all $i > 0$. As it has been precised in paragraph 4.3, the model allows to construct a semi closed solution to the filtering problem: it is of *Serial Gaussian* distribution with recursively determined parameters (see Algorithm 5). Hence, reference values for the considered test functions can be computed via Monte Carlo simulations from the filter distribution.

Results in Figure 5 have been obtained for the set of parameters $(\rho, \theta, \lambda) = (0.5, 1, 0.1)$ and $n = 10$. We choose the stationary distribution $\mathcal{N}(0, \frac{\theta^2}{1-\rho^2})$ as the initial signal one. In Figure 6, are depicted resluts for $(\rho, \theta, \lambda) = (0.65, 1, 0.1)$ and $n = 10$. Here, the initial distribution has been fixed to $\mathcal{N}(0, \sigma_0^2)$ with $\sigma_0 = 0.05$.

On one hand the first line depicts the quantization based filter behaviour with respect

to the total quantizer size. On the other hand, the second and third line represent respectively the **SIS** and the **SIR** confidence intervals with the empirical mean value and centiles as functions of the particle number. Confidence intervals for particle filters are estimated over $M = 1000$ realizations.

As for the previous example, we observe the degeneracy of **SIS** filter, the convergence value is far from the reference value. Quantization based filter and **SIR** filter converge to the reference value. It is worth noting that the grid size order is quite different whether the method is a quantization based one or a particle filter one. Once again, in this example, particle methods need much more points than quantization ones. This compensates by far the higher complexity order of quantization approach.

References

- [1] S. Arulampalam, T. Clapp, N. Gordon, and S. Maskall. A tutorial on particle filters for On-line Non-linear/Non Gaussian Bayesian tracking. QinetiQ Ltd, DSTO, IEEE, 2001.
- [2] V. Bally, M.P. Bavouzet, and M. Messaoud. Integration by parts formula for locally smooth laws and applications to sensitivity computations. Rapport de recherche de l'INRIA : RR-5567, 2005.
- [3] V. Bally and G. Pagès. A quantization algorithm for solving discrete time multi-dimensional optimal stopping problems. *Bernoulli*, 9:1003–1049, 2003.
- [4] V. Bally, G. Pagès, and J. Printems. First order schemes in the numerical quantization method. *Mathematical finance*, 13(1):1–16, 2003.
- [5] V. Bally, G. Pagès, and J. Printems. A quantization tree method for pricing and hedging multidimensional american options. *Mathematical Finance*, 15(1):119–168, 2005.
- [6] D. Crisan and A. Doucet. A Survey of Convergence Results on Particle Filtering Methods for Practitioners. *IEEE Transactions on signal processing*, 50:736–746, 2002.
- [7] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo methods in Practice*. Springer, 1st edition, 2001.
- [8] V. Genon-Catalot. A non linear explicit filter. *Statist. and Prob. letters*, 61:145–154, 2003.
- [9] F. Le Gland. *Introduction au filtrage en temps discret. Filtre de Kalman, Modèles de Markov cachés*. IRISA/INRIA, 2002-2003.
- [10] N. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings*, 140(2):107–113, April 1993.

- [11] S. Graf and H. Luschgy. *Foundations of quantization for probability distributions*. Lecture Notes in Mathematics. Springer, 2000.
- [12] T.J. Lyons and T.S. Salisbury, editors. *Numerical Methods and Stochastics, USA, 1999*. American Mathematical Society.
- [13] P. Del Moral, J. Jacod, and P. Protter. The Monte Carlo method for filtering with discrete-time observations. *Probab. Theory and Relat. Fields*, 120(2):346–368, June 2001.
- [14] G. Pagès and H. Pham. Optimal quantization methods for nonlinear filtering with discrete time observations. To appear in *Bernoulli*, 2003.
- [15] G. Pagès, H. Pham, and J. Printems. Optimal quantization methods and applications to numerical problems in finance. In *Handbook of Computational and Numerical Methods in Finance*. S.T. Rachev, Birkhauser, Boston, 2004.
- [16] A. Sellami. Quantization based filtering method using first order approximation. Preprint of Laboratoire de Probabilités et Modèles Aléatoires. PMA-1009, 2005.

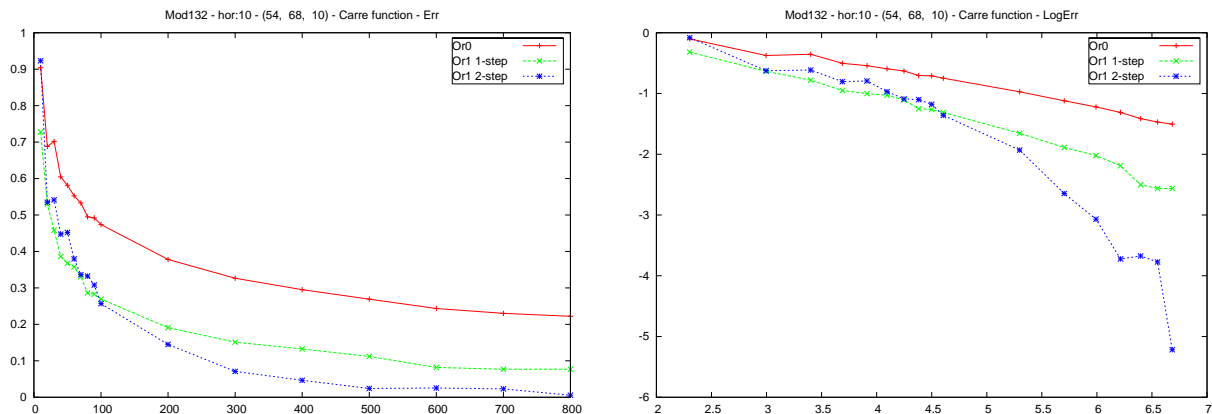


Figure 1: Quantization filter estimator errors for 3-dimensional Kalman case as a function of the quantizer size N_k (top: $\|\Pi_{10} f_2 - \hat{\Pi}_{10} f_2\|_2$, bottom: log-log scale representation).

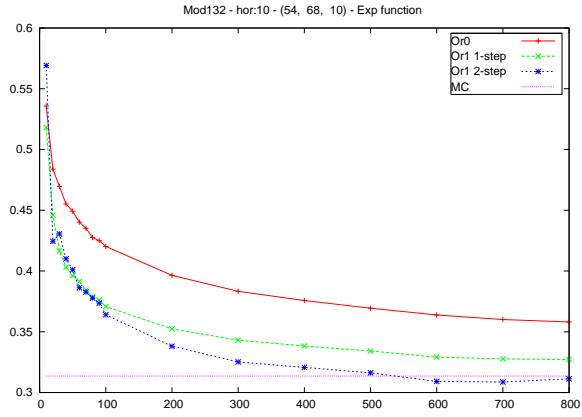


Figure 2: Quantization filter estimator errors for 3-dimensional Kalman case as a function of the quantizer size $N_k : \|\Pi_{10}f_3 - \hat{\Pi}_{10}f_3\|_2$.

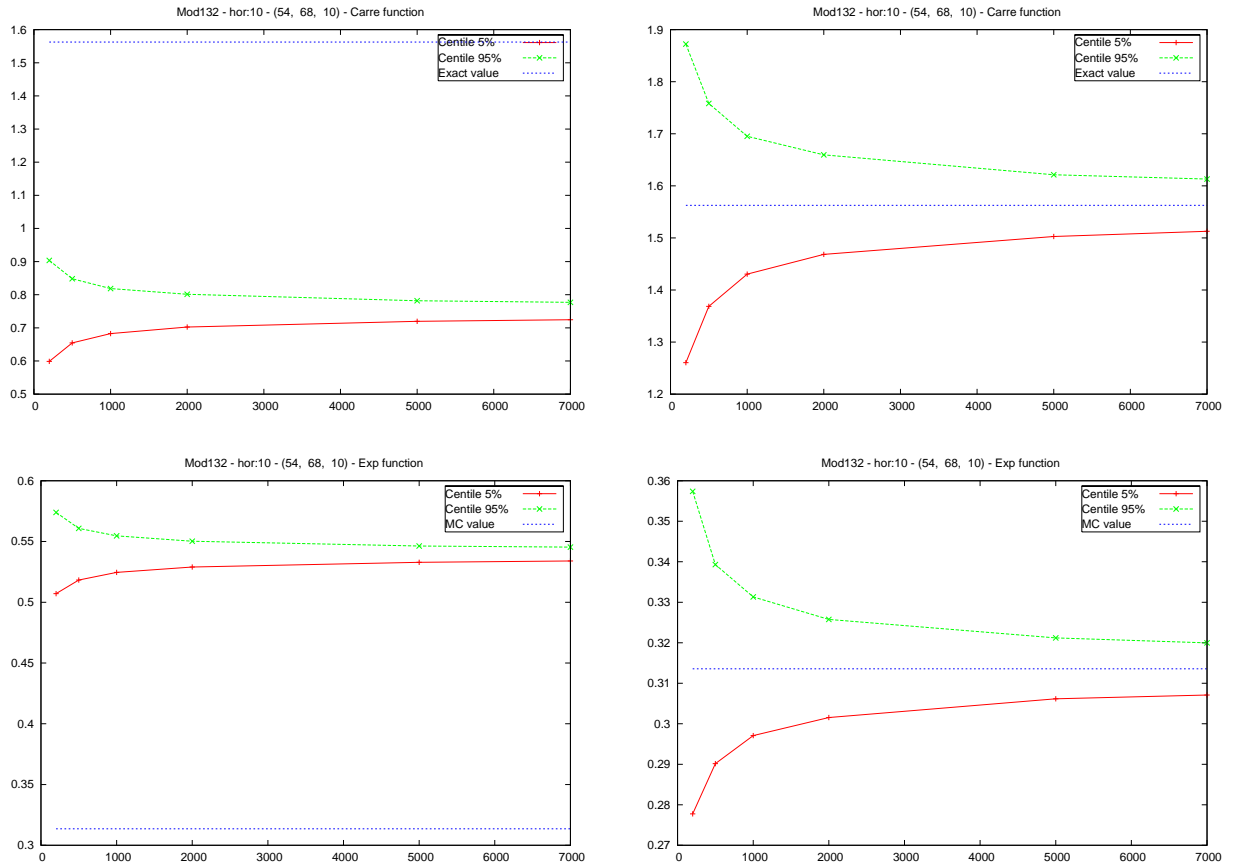


Figure 3: Particle filter confidence intervals for 3-dimensional Kalman case as a function of the particle number N .

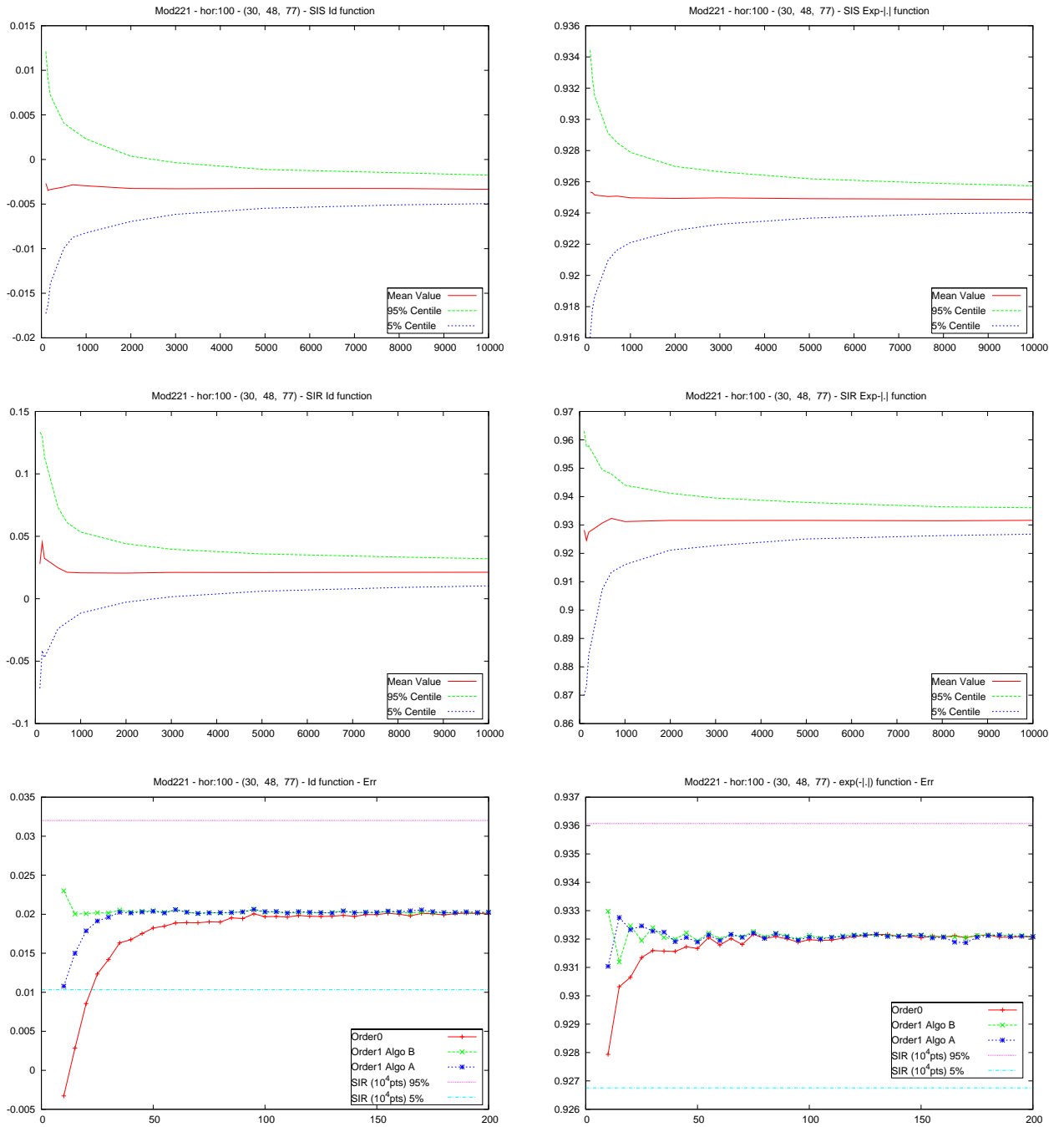


Figure 4: Particle filter and quantization filter approximations for SVM (left: $\Pi_{y,100}f_1$, right: $\Pi_{y,100}f_3$) - $(\beta, \sigma) = (0.995, 0.01)$.

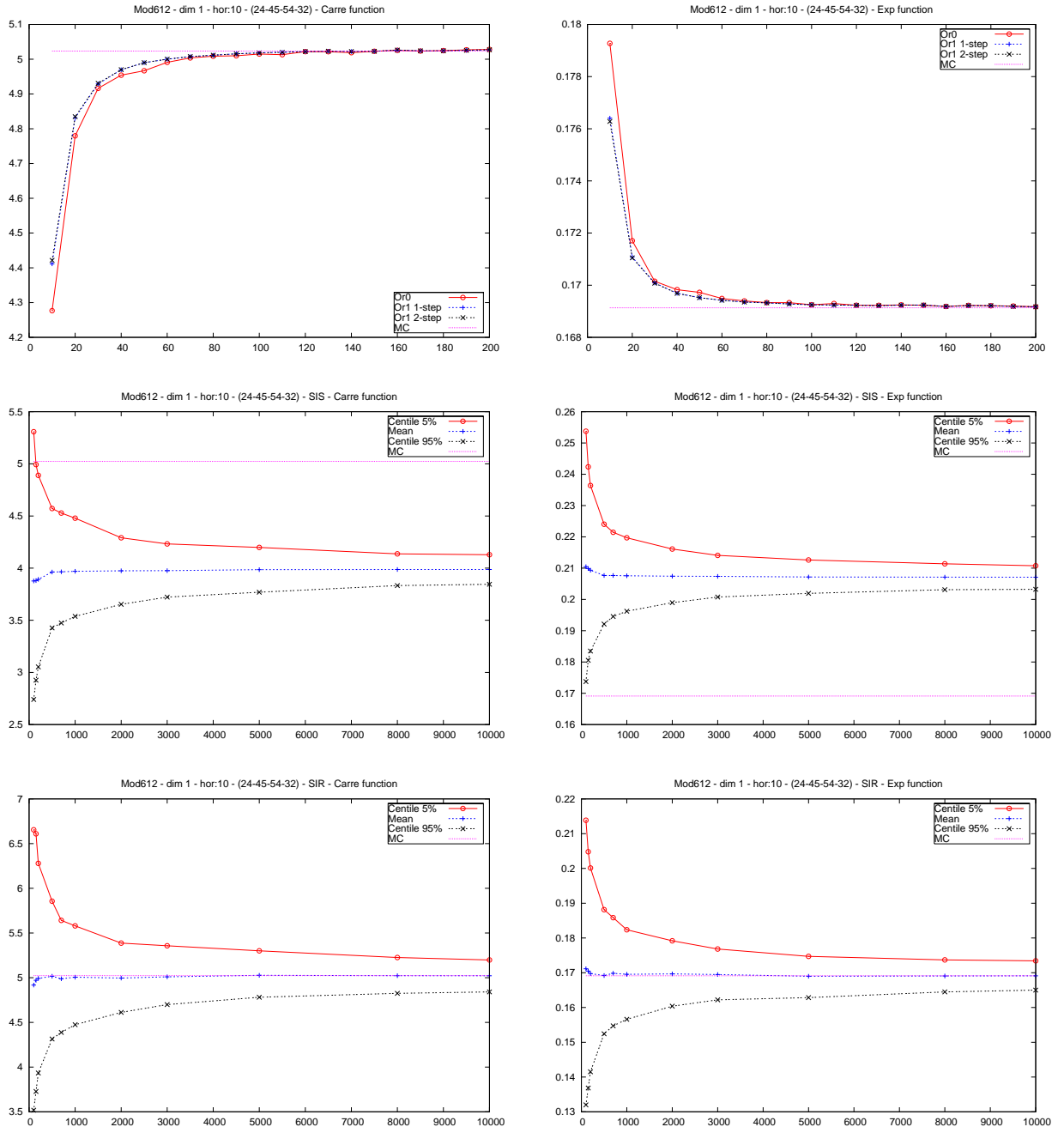


Figure 5: Explicit filter estimators as function of grid sizes (left: $\hat{\Pi}_{10}f_2$, right: $\hat{\Pi}_{10}f_3$) - $(\rho, \theta, \lambda, n) = (0.5, 1, 0.1, 10)$.

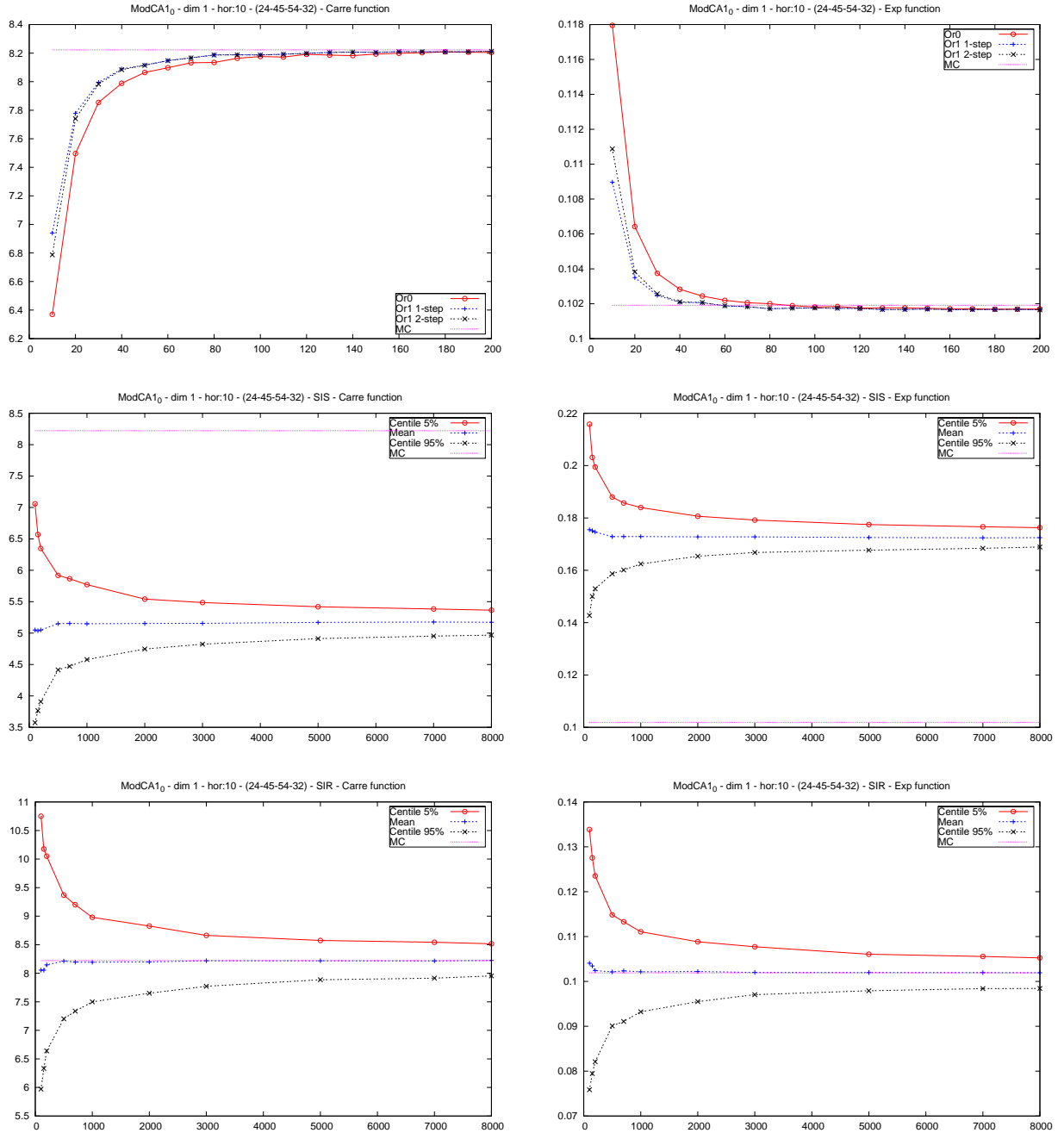


Figure 6: Explicit filter estimators as function of grid sizes (left: $\hat{\Pi}_{10}f_2$, right: $\hat{\Pi}_{10}f_3$) - $(\rho, \theta, \lambda, \sigma_0, n) = (0.65, 1, 0.1, 0.05, 10)$.