

OPTIMAL SIR ALGORITHM VS. FULLY ADAPTED AUXILIARY PARTICLE FILTER : A MATTER OF CONDITIONAL INDEPENDENCE

François Desbouvries, Yohan Petetin and Emmanuel Monfrini

Telecom Institute, Telecom SudParis, CITI Department & CNRS UMR 5157,
9 rue Charles Fourier, 91011 Evry, France

{francois.desbouvries,yohan.petetin,emmanuel.monfrini}@it-sudparis.eu

ABSTRACT

Particle filters (PF) and auxiliary particle filters (APF) are widely used sequential Monte Carlo (SMC) techniques. In this paper we comparatively analyse the Sampling Importance Resampling (SIR) PF with optimal conditional importance distribution (CID) and the fully adapted APF (FA-APF). Both algorithms share the same Sampling (S), Weighting (W) and Resampling (R) steps, and only differ in the order in which these steps are performed. The order of the operations is not insignificant : starting at time $n - 1$ from a common set of particles, we show that one single updated particle at time n will marginally be sampled in both algorithms from the same probability density function (pdf), but as a whole the full set of particles will be conditionally independent if created by the FA-APF algorithm, and dependent if created by the SIR algorithm, which results in support degeneracy.

Index Terms— Sequential Monte Carlo, Particle Filtering, Auxiliary Particle Filtering, Conditional Independence.

1. INTRODUCTION

Let x_n and y_n be respectively a hidden and an observed process. Let $p(x_n|\mathbf{y}_{0:n})$, say, denote the pdf (w.r.t. Lebesgue measure) of x_n given $\mathbf{y}_{0:n} = \{y_i\}_{i=0}^n$. We assume that $\{x_n, y_n\}_{n \in \mathbb{N}}$ is a Hidden Markov Chain : $p(\mathbf{x}_{0:n}, \mathbf{y}_{0:n}) = p(x_0) \prod_{i=1}^n p(x_i|x_{i-1}) \prod_{i=0}^n p(y_i|x_i)$. Bayesian filtering consists in computing $p(x_n|\mathbf{y}_{0:n})$, or at least some approximation of the measure $p(dx_n|\mathbf{y}_{0:n})$ with density $p(x_n|\mathbf{y}_{0:n})$.

Recursive solutions are of particular interest, and indeed $p(x_n|\mathbf{y}_{0:n})$ can be computed from $p(x_{n-1}|\mathbf{y}_{0:n-1})$ by the well known recursion (here \mathcal{N} stands for numerator) :

$$p(x_n|\mathbf{y}_{0:n}) = \frac{p(y_n|x_n) \int p(x_n|x_{n-1})p(x_{n-1}|\mathbf{y}_{0:n-1})dx_{n-1}}{p(y_n|\mathbf{y}_{0:n-1}) = \int \mathcal{N}dx_n} \quad (1)$$

Many efforts have been devoted to the computation of eq. (1). If exact computing is difficult or impossible, one needs to resort to approximations. Among them PF and APF methods (see e.g. [1]-[4]) are SMC methods which propagate a discrete approximation $\hat{p}(dx_n|\mathbf{y}_{0:n})$ of $p(dx_n|\mathbf{y}_{0:n})$.

The aim of this paper is to comparatively analyse two closely related algorithms, the SIR algorithm with optimal CID [1] and the FA-APF [4]. We show that starting from a common set of points $\mathbf{x}_{n-1}^{\{i\}} \stackrel{def}{=} \{x_{n-1}^i\}_{i=1}^N$ sampled from $p(x_{n-1}|\mathbf{y}_{0:n-1})$, a new particle produced by either algorithm is marginally sampled from the same pdf, but the whole set of particles is (conditionally on $\mathbf{x}_{n-1}^{\{i\}}$ and y_n) independent if produced by FA-APF, while it is dependent if produced by SIR. This paper is organized as follows. The SIR and FA-APF algorithms are recalled in §2. Our analysis is performed in §3, and in §4 we compare both algorithms via simulations.

2. TWO CLOSELY RELATED SMC FILTERS

Let us first recall the well known SIR algorithm with optimal CID (see e.g. [1]). It consists of the succession of a sampling step *S*, a weighting step *W* and an optional¹ resampling step *R*. The algorithm (which will simply be denoted as SIR in the sequel) is as follows :

SIR algorithm (with optimal CID and systematic Resampling).

Let $\hat{p}(dx_{n-1}|\mathbf{y}_{0:n-1}) = \sum_{i=1}^N \frac{1}{N} \delta_{x_{n-1}^i}(dx_{n-1})$.

S. For $1 \leq i \leq N$, sample \tilde{x}_n^i from $p(\tilde{x}_n|x_{n-1}^i, y_n)$;

W. For $1 \leq i \leq N$, compute $w_n^i \propto p(y_n|x_{n-1}^i), \sum_{i=1}^N w_n^i = 1$;

R. For $1 \leq i \leq N$, sample $x_n^i \sim \sum_{i=1}^N w_n^i \delta_{\tilde{x}_n^i}(dx_n)$.

Then $\hat{p}(dx_n|\mathbf{y}_{0:n}) = \sum_{i=1}^N \frac{1}{N} \delta_{x_n^i}(dx_n) \simeq p(dx_n|\mathbf{y}_{0:n})$.

Let us now briefly recall the principle of APF [4] [6] (see also [7] for recent developments). If exact computing of (1) is not possible, a natural idea consists in replacing $p(dx_{n-1}|\mathbf{y}_{0:n-1})$ by a discrete approximation $\sum_{i=1}^N w_{n-1}^i \delta_{x_{n-1}^i}(dx_{n-1})$. Injecting into (1) we get a continuous approximation of $p(x_n|\mathbf{y}_{0:n})$:

$$\pi(x_n|\mathbf{y}_{0:n}) = \frac{\sum_{i=1}^N w_{n-1}^i p(y_n|x_n) p(x_n|x_{n-1}^i)}{\int \mathcal{N}dx_n}$$

¹For the purposes of this paper we will assume systematic resampling, but variants are available, see e.g. [5].

which, since $p(x_n, y_n | x_{n-1}) = p(y_n | x_n) p(x_n | x_{n-1}) = p(x_n | x_{n-1}, y_n) p(y_n | x_{n-1})$, can be rewritten as

$$\pi(x_n | \mathbf{y}_{0:n}) = \sum_{i=1}^N \frac{w_{n-1}^i p(y_n | x_{n-1}^i)}{\sum_{i=1}^N w_{n-1}^i p(y_n | x_{n-1}^i)} p(x_n | x_{n-1}^i, y_n). \quad (2)$$

In the view of deriving an SMC algorithm one should get samples from (2) in order to further proceed at time $n + 1$. If direct sampling from π is difficult or impossible, one can approximate π by the mixture pdf

$$q(x_n) = \sum_{i=1}^N \frac{w_{n-1}^i \tau_{n-1}^i}{\sum_{i=1}^N w_{n-1}^i \tau_{n-1}^i} \tilde{q}(x_n | x_{n-1}^i), \quad (3)$$

sample from q (the so-called first-stage weights $\tilde{w}_{n-1}^i \propto w_{n-1}^i \tau_{n-1}^i$ and pdf \tilde{q} are degrees of freedom used for designing this importance density), and use importance sampling (IS). Since the target π and the importance density q are both mixture densities, the (so-called second-stage) weights computed by this IS method would be the ratio of a sum of N terms over another sum of N terms. This drawback is avoided by using data augmentation, see e.g. [4] for details.

Now, if direct sampling from π is indeed feasible then we are in the so-called *fully-adapted* (FA) case, in the terminology of [4]. Remember that to get a sample from a mixture $\sum_{i=1}^N \alpha_i p_i(x)$ one can first draw an index j from $\sum_{i=1}^N \alpha_i \delta_i$ ((*W,R*) steps) and then sample x from the $p_j(x)$ which has just been selected (*S* step). Applying to (2) we get the

FA-APF algorithm²

Let $\hat{p}(dx_{n-1} | \mathbf{y}_{0:n-1}) = \sum_{i=1}^N \frac{1}{N} \delta_{x_{n-1}^i}(dx_{n-1})$.

W. For $1 \leq i \leq N$, compute $w_n^i \propto p(y_n | x_{n-1}^i)$, $\sum_{i=1}^N w_n^i = 1$;

R. For $1 \leq i \leq N$, sample $\tilde{x}_{n-1}^i \sim \sum_{i=1}^N w_n^i \delta_{x_{n-1}^i}(dx_{n-1})$;

S. For $1 \leq i \leq N$, sample x_n^i from $p(x_n | \tilde{x}_{n-1}^i, y_n)$.

Then $\hat{p}(dx_n | \mathbf{y}_{0:n}) = \sum_{i=1}^N \frac{1}{N} \delta_{x_n^i}(dx_n) \simeq p(dx_n | \mathbf{y}_{0:n})$.

As we see the FA-APF algorithm is a *reordering* of the SIR algorithm. In SIR the successive steps are $S \rightarrow W \rightarrow R$ (or, equivalently, $W \rightarrow S \rightarrow R$: steps S and W commute since the incremental weight $p(y_n | x_{n-1})$ does not depend on the new particle \tilde{x}_n^i), while the recursive loop of FA-APF is made of the successive steps $W \rightarrow R \rightarrow S$.

3. ANALYSING THE SIR AND FA-APF ALGORITHMS

In this section we analyse the SIR and FA-APF algorithms. Assume that at time $n - 1$ we have N points $\{x_{n-1}^i\}_{i=1}^N$ (at least approximately) sampled from $p(x_{n-1} | \mathbf{y}_{0:n-1})$. Starting

²In the view of §3 we set $w_{n-1}^i = \frac{1}{N}$ for all i , i.e. we start from an unweighted pdf.

from this common set of points, our aim is to compare the samples $\mathbf{x}_n^{\{i\}}$ produced either by the SIR algorithm, or by the FA-APF one. To that aim we are going to compute, for both algorithms, the conditional pdf $p(\mathbf{x}_n^{\{i\}} | \mathbf{x}_{n-1}^{\{i\}}, y_n)$.

3.1. Reformulating the SIR and FA-APF algorithms

Let us first rewrite the algorithms in terms of conditional pdf. Both algorithms compute $\mathbf{x}_n^{\{i\}}$ from $\mathbf{x}_{n-1}^{\{i\}}$ via an intermediate set of points $\tilde{\mathbf{x}}_n^{\{i\}} \stackrel{def}{=} \{\tilde{x}_n^i\}_{i=1}^N$ (computed by the *S* step in SIR, and by the (*W,R*) steps in FA-APF), and so each algorithm can be written as the succession of two steps.

- Let us begin with SIR. Let $\bar{p}(y_n | x_{n-1}^i) = \frac{p(y_n | x_{n-1}^i)}{\sum_{i=1}^N p(y_n | x_{n-1}^i)}$.

In the (*S*) step, conditionally on $\mathbf{x}_{n-1}^{\{i\}}$, we independently draw N particles $\tilde{\mathbf{x}}_n^{\{i\}}$. We next weight them, and conditionally on $\mathbf{x}_{n-1}^{\{i\}}$ and $\tilde{\mathbf{x}}_n^{\{i\}}$, we draw N independent samples $\mathbf{x}_n^{\{i\}}$ from this reweighted distribution (*R* step). This algorithm can be rewritten as follows :

SIR algorithm. Let $\mathbf{x}_{n-1}^{\{i\}} \stackrel{i.i.d.}{\sim} p(x_{n-1} | \mathbf{y}_{0:n-1})$. Then

$$\begin{aligned} \tilde{\mathbf{x}}_n^{\{i\}} &\stackrel{S}{\sim} p(\tilde{\mathbf{x}}_n | \mathbf{x}_{n-1}^{\{i\}}, y_n) = \prod_{i=1}^N p(\tilde{x}_n^i | x_{n-1}^i, y_n); \quad (4) \\ \mathbf{x}_n^{\{i\}} &\stackrel{(W,R)}{\sim} p(\mathbf{x}_n | \tilde{\mathbf{x}}_n^{\{i\}}, \mathbf{x}_{n-1}^{\{i\}}, y_n) \\ &= \prod_{i=1}^N \left(\sum_{j=1}^N \bar{p}(y_n | x_{n-1}^j) \delta_{\tilde{x}_n^j}(x_n^i) \right). \quad (5) \end{aligned}$$

- Now in FA-APF, conditionally on $\mathbf{x}_{n-1}^{\{i\}}$ and y_n , we draw N independent points $\mathbf{x}_n^{\{i\}}$ from $\pi(x_n | \mathbf{y}_{0:n})$ defined in (2). The algorithm can be rewritten on the whole as follows :

FA-APF algorithm. Let $\mathbf{x}_{n-1}^{\{i\}} \stackrel{i.i.d.}{\sim} p(x_{n-1} | \mathbf{y}_{0:n-1})$. Then

$$\begin{aligned} \tilde{\mathbf{x}}_{n-1}^{\{i\}} &\stackrel{(W,R)}{\sim} p(\tilde{\mathbf{x}}_{n-1} | \mathbf{x}_{n-1}^{\{i\}}, y_n) \quad (6) \\ &= \prod_{i=1}^N \left(\sum_{j=1}^N \bar{p}(y_n | x_{n-1}^j) \delta_{x_{n-1}^j}(\tilde{x}_{n-1}^i) \right); \\ \mathbf{x}_n^{\{i\}} &\stackrel{S}{\sim} p(\mathbf{x}_n | \mathbf{x}_{n-1}^{\{i\}}, \tilde{\mathbf{x}}_{n-1}^{\{i\}}, y_n) \\ &= \prod_{i=1}^N p(x_n^i | \tilde{x}_{n-1}^i, y_n). \quad (7) \end{aligned}$$

Note that in FA-APF we sample from the discrete pdf (6) and next from the continuous one (7), while this is the contrary in SIR ((4) is continuous and (5) discrete). As we will see, this will induce consequences on the degeneracy of the support of $\{x_n^i\}_{i=1}^N$ produced by the SIR algorithm (see Remark 2 below).

3.2. Marginal conditional pdf

We are now ready to compute $p(x_n | \mathbf{x}_{n-1}^{\{i\}}, y_n)$ for both algorithms :

Proposition 1 *Let $\{x_{n-1}^i\}_{i=1}^N$ be i.i.d. samples from $p(x_{n-1} | \mathbf{y}_{0:n-1})$. Let x_n^i (resp. \bar{x}_n^i) be a point computed by the SIR (resp. FA-APF) algorithm. Then both x_n^i and \bar{x}_n^i are sampled from $p(x_n | \mathbf{x}_{n-1}^{\{i\}}, y_n) = \pi(x_n | \mathbf{y}_{0:n})$ defined in (2).*

Proof 1 *By construction each sample \bar{x}_n^i computed by FA-APF is drawn from π . Let us address SIR. For any Borel set $A \in \mathcal{B}(\mathbb{R})$ let $\mu(A) = P(x_n \in A | \mathbf{x}_{n-1}^{\{i\}}, y_n)$. Then*

$$\begin{aligned} \mu(A) &= \int_{\mathbb{R}^N} \left[\sum_{i=1}^N \bar{p}(y_n | x_{n-1}^i) \delta_{\bar{x}_n^i}(A) \right] \prod_{j=1}^N p(\bar{x}_n^j | x_{n-1}^j, y_n) d\bar{\mathbf{x}}_n \\ &= \sum_{i=1}^N \bar{p}(y_n | x_{n-1}^i) \int_{\mathbb{R}^N} \delta_{\bar{x}_n^i}(A) \prod_{j=1}^N p(\bar{x}_n^j | x_{n-1}^j, y_n) d\bar{\mathbf{x}}_n \\ &= \sum_{i=1}^N \bar{p}(y_n | x_{n-1}^i) \int_{\mathbb{R}} \mathbb{1}_A(\bar{x}_n^i) p(\bar{x}_n^i | x_{n-1}^i, y_n) d\bar{x}_n^i \\ &= \int_A \underbrace{\sum_{i=1}^N \frac{p(y_n | x_{n-1}^i)}{\sum_{k=1}^N p(y_n | x_{n-1}^k)}}_{\pi(x_n | \mathbf{y}_{0:n})} p(x_n | x_{n-1}^i, y_n) dx_n, \end{aligned}$$

so μ has pdf π w.r.t. the Lebesgue measure. \blacksquare

3.3. Joint conditional pdf

As we see from Proposition 1, conditionally on the old particles $\mathbf{x}_{n-1}^{\{i\}}$ and the new observation y_n , each single point produced by both algorithms is *marginally* drawn from the same conditional pdf π . The difference appears when we consider the *joint* conditional pdf $p(\mathbf{x} | \mathbf{x}_{n-1}^{\{i\}}, y_n)$. We begin with the following result (the proof is omitted due to lack of space) :

Proposition 2 *Let $\{x_{n-1}^i\}_{i=1}^N$ be i.i.d. samples from $p(x_{n-1} | \mathbf{y}_{0:n-1})$. Let x_n^1 and x_n^2 be the first two points computed by SIR, and let $\bar{p}(x_n^1 | x_{n-1}^i, y_n) = \frac{p(x_n^1 | x_{n-1}^i, y_n)}{\sum_{i=1}^2 p(x_n^1 | x_{n-1}^i, y_n)}$. For any Borel set $A \in \mathcal{B}(\mathbb{R})$ let $\mu(A) = P(x_n^2 \in A | \mathbf{x}_{n-1}^{\{i\}}, y_n, x_n^1)$. Then μ is the mixed (i.e., discrete and continuous) probability measure*

$$\mu(dx) = \alpha \delta_{x_n^1}(dx) + \beta p(x) dx, \quad (8)$$

in which

$$\begin{aligned} \alpha &= P(x_n^2 = x_n^1 | \mathbf{x}_{n-1}^{\{i\}}, y_n, x_n^1) \\ &= \frac{\sum_{i=1}^2 (\bar{p}(y_n | x_{n-1}^i))^2 p(x_n^1 | x_{n-1}^i, y_n)}{\sum_{i=1}^2 \bar{p}(y_n | x_{n-1}^i) p(x_n^1 | x_{n-1}^i, y_n)}, \quad (9) \end{aligned}$$

$$\beta = 1 - \alpha, \quad (10)$$

$$p(x) = \sum_{i=1}^2 \bar{p}(x_n^1 | x_{n-1}^{3-i}, y_n) p(x | x_{n-1}^i, y_n). \quad (11)$$

Remark 1 (conditional independence.) Let us comment this result. The SIR algorithm draws particles $\tilde{\mathbf{x}}_n^{\{i\}}$, reweights them, and finally resamples, i.e. draws N samples $\mathbf{x}_n^{\{i\}}$ which are independent conditionally on \mathbf{y}_n , $\mathbf{x}_{n-1}^{\{i\}}$ and on $\tilde{\mathbf{x}}_n^{\{i\}}$ (see (5)). But these samples are no longer independent conditionally on \mathbf{y}_n and $\mathbf{x}_{n-1}^{\{i\}}$ only : since we resample from a discrete distribution $\sum_{i=1}^N \bar{p}(y_n | x_{n-1}^i) \delta_{\tilde{x}_n^i}(x_n)$, once the first particle x_n^1 has been drawn, the second one x_n^2 is either equal to x_n^1 with probability α (given by (9)), or can take any real value distributed according to the mixture pdf $p(x)$ defined in (11).

Proposition 3 (conditional independence.) *Let $\{x_{n-1}^i\}_{i=1}^N$ be i.i.d. samples from $p(x_{n-1} | \mathbf{y}_{0:n-1})$. Let $\mathbf{x}_n^{\{i\}} = \{x_n^i\}_{i=1}^N$ (resp. $\bar{\mathbf{x}}_n^{\{i\}} = \{\bar{x}_n^i\}_{i=1}^N$) be the N points computed by SIR (resp. FA-APF). Then conditionally on $\mathbf{x}_{n-1}^{\{i\}}$ and y_n , $\bar{\mathbf{x}}_n^{\{i\}}$ are i.i.d. samples from $\pi(x_n | \mathbf{y}_{0:n})$, while $\mathbf{x}_n^{\{i\}}$ are identically distributed from $\pi(x_n | \mathbf{y}_{0:n})$ but are not independent.*

Proof 2 Conditionally on $\mathbf{x}_{n-1}^{\{i\}}$ and y_n , the FA-APF algorithm draws independent samples by construction. That the N particles computed by the SIR algorithm are not independent is a consequence of proposition 2. \blacksquare

Remark 2 (support degeneracy.) As is well known, in a sequential IS algorithm weights degenerate with time, so a lot of computational effort is devoted to sampling particles which will be associated to vanishing weights. Resampling was introduced as a rescue to this problem, and indeed this technique eliminates particles associated with weights that are too low. But since in the SIR algorithm a point x_n^i can be sampled several times, the set $\{x_n^i\}_{i=1}^N$ is a subset of $\{\tilde{x}_n^i\}_{i=1}^N$. So the SIR algorithm is not computationally efficient, because the particles \tilde{x}_n^i which finally will not be selected during the R step have indeed been drawn unnecessarily. More importantly, due to resampling the support of the distribution shrinks, which is another form of degeneracy. Qualitatively this support degeneracy can be measured by the (conditional) average size of the support after resampling $E(N')$, defined as the expectation of the number of points which have been (re-)selected at least once. Since the total number of outcomes of each particle after the R step follows a multinomial distribution with parameters $\bar{p}(y_n | x_{n-1}^i)$, one can show easily that

$$E(N') = N - \sum_{i=1}^N (1 - \bar{p}(y_n | x_{n-1}^i))^N. \quad (12)$$

Now by contrast in FA-APF the support cannot degenerate. Of course, the R step still induces support shrinkage, so $\{\tilde{x}_{n-1}^i\}_{i=1}^N$ is a subset of $\{x_{n-1}^i\}_{i=1}^N$; but next each x_n^i is drawn from the continuous pdf $p(x_n^i | \tilde{x}_{n-1}^i, y_n)$, so particles x_n^i and x_n^j will be different even if $\tilde{x}_{n-1}^i = \tilde{x}_{n-1}^j$.

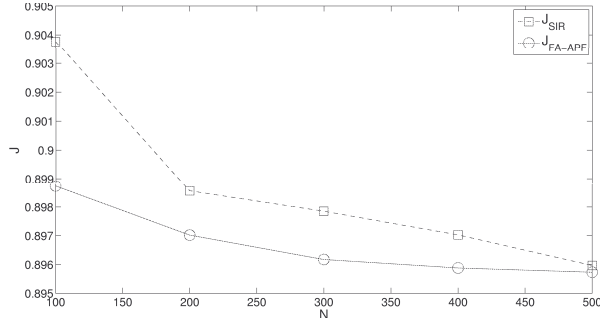


Fig. 1. Empirical Standard Deviation ($\beta_0 = 1$, $\beta_1 = 0.1$, $R = 3$)

4. SIMULATIONS

We now perform simulations. We consider the ARCH state space model :

$$x_{n+1} = \sqrt{\beta_0 + \beta_1 x_n^2} \times u_n \quad (13)$$

$$y_n = x_n + v_n \quad (14)$$

in which $u_n \sim \mathcal{N}(0, 1)$, $v_n \sim \mathcal{N}(0, R)$, and x_0 , $\{u_n\}$ and $\{v_n\}$ are independent. For this model $p(x_n|x_{n-1}, y_n) \sim \mathcal{N}\left(\frac{\beta_0 + \beta_1 x_{n-1}^2}{R + \beta_0 + \beta_1 x_{n-1}^2}, \frac{R(\beta_0 + \beta_1 x_{n-1}^2)}{R + \beta_0 + \beta_1 x_{n-1}^2}\right)$ and $p(y_n|x_{n-1}) \sim \mathcal{N}(0, R + \beta_0 + \beta_1 x_{n-1}^2)$. We sample P trajectories, and define \mathcal{J} as

the empirical standard deviation $\frac{1}{T} \sum_{n=1}^T \left[\frac{1}{P} \sum_{j=1}^P (\hat{x}_{n|n}^j - x_n^j)^2 \right]^{\frac{1}{2}}$

(x_n^j and $\hat{x}_{n|n}^j$ are respectively the true and estimated state at time n for trajectory j). We set $\beta_0 = 1$, $\beta_1 = 0.1$, $R = 3$, $P = 400$ and $T = 50$. Fig. 1 displays \mathcal{J} against the number of particles N . FA-APF gives better results and the distance between the two methods decreases as N increases. Note that $\mathcal{J}_{SIR}(400) = \mathcal{J}_{FA-APF}(200) \simeq 0.8970$, so that similar performances are obtained by FA-APF as compared to SIR, but with half as many particles.

In the second experiment we set $\beta_0 = 9$, $\beta_1 = 5$ and $R = 1$. Fig. 2 displays the empirical MSE $MSE(n) = \frac{1}{P} \sum_{j=1}^P (\hat{x}_{n|n}^j - x_n^j)^2$ as a function of time. As we see FA-APF outperforms SIR for almost all values of k . This is because we take few particles ($N = 50$), and the support of particles for SIR is smaller than that of FA-APF (we checked that for SIR the empirical mean is located around 31, which is consistent with the theoretical value (12)).

5. CONCLUSION

In this paper we analyzed two closely related, well known SMC techniques : the SIR algorithm with optimal CID and the FA-APF algorithm. Both algorithms share the same operations and only differ by the order in which they are per-

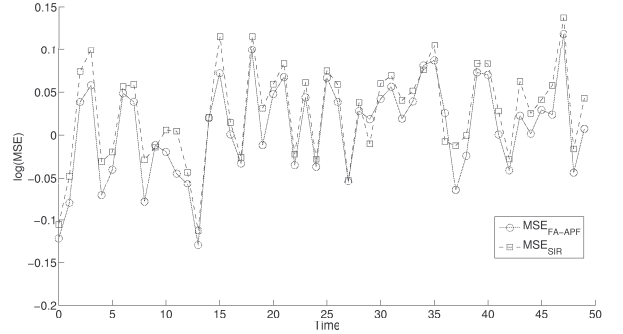


Fig. 2. Empirical MSE ($\beta_0 = 9$, $\beta_1 = 5$, $R = 1$)

formed : in SIR we sample, reweight and resample; in FA-APF we weigh, resample and finally sample. Starting from a common set of particles, we first showed that a particle produced by both algorithms is marginally drawn from the same mixture pdf. However in FA-APF the particles are also sampled independently, whereas they are dependent in SIR, which induces support degeneracy.

6. REFERENCES

- [1] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, 2000.
- [2] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer Verlag, New York, 2001.
- [3] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear / non-Gaussian Bayesian tracking," *IEEE Tr: Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [4] M. K. Pitt and N. Shephard, "Filtering via simulation : Auxiliary particle filter," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 550–99, june 1999.
- [5] R. Douc, O. Cappé, and E. Moulines, "Comparison of resampling schemes for particle filtering," in *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*, Zagreb, Croatia, September 2005.
- [6] P. Fearnhead, "Computational methods for complex stochastic systems: A review of some alternatives to MCMC," *Statistics and Computing*, vol. 18, no. 2, pp. 151–71, 2008.
- [7] N. Whiteley and A. M. Johansen, "Recent developments in auxiliary particle filtering," in *Inference and Learning in Dynamic Models*, Barber, Cemgil, and Chiappa, Eds. 2010, Cambridge University Press.