AN IMPROVED SIR-BASED SEQUENTIAL MONTE CARLO ALGORITHM

Roland Lamberti[1]*, Yohan Petetin*[1]*, François Septier*[2]*, François Desbouvries*[1]

(1) Samovar, Telecom Sudparis, CNRS, Université Paris-Saclay, 9 rue Charles Fourier, 91011 Evry, France
(2) Télécom Lille, Univ. Lille, CNRS, UMR 9189 - CRIStAL, F-59000 Lille, France

ABSTRACT

Sequential Monte Carlo (SMC) algorithms are based on importance sampling (IS) techniques. Resampling has been introduced as a tool for fighting the weight degeneracy problem. However, for a fixed sample size N, the resampled particles are dependent, are not drawn exactly from the target distribution, nor are weighted properly. In this paper, we revisit the resampling mechanism and propose a scheme where the resampled particles are (conditionally) independent and weighted properly. We validate our results via simulations.

***Index Terms*—** Importance sampling, resampling procedures, sequential Monte Carlo.

1. INTRODUCTION

Let us consider a hidden (resp. observed) process $\{\mathbf{X}_k\}_{k\geq 0}$ (resp. $\{\mathbf{Y}_k\}_{k\geq 0}$). We assume that $\{(\mathbf{X}_k, \mathbf{Y}_k)\}_{k\geq 0}$ is a hidden Markov chain, i.e. that

$$p(\mathbf{x}_{0:k}, \mathbf{y}_{0:k}) = p(\mathbf{x}_0) \prod_{i=1}^{k} f_{i|i-1}(\mathbf{x}_i|\mathbf{x}_{i-1}) \prod_{i=0}^{k} g_k(\mathbf{y}_i|\mathbf{x}_i). \quad (1)$$

In many applications we are interested in computing a moment of the pdf $p(\mathbf{x}_k|\mathbf{y}_{0:k})$ (or $p_{k|k}$ in shorthand notation) of $\mathbf{X}_k$ given the past observations $\mathbf{y}_{0:k} = \{\mathbf{y}_i\}_{i=0}^{k}$.

However, computing $p_{k|k}$ is often impossible in practice, and many approximate solutions have been proposed. Among them, SMC algorithms propagate over time a weighted discrete approximation $\hat{p}_{k|k} = \sum_{i=1}^{N} w_k^i \delta_{\mathbf{x}_k^i}$ of $p_{k|k}$. The popular Sampling Importance Resampling (SIR) algorithm [1] [2] relies on a conditional importance distribution $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{0:k})$ from which particles are sampled; these particles are weighted to correct the discrepancy between $p(\mathbf{x}_{0:k}|\mathbf{y}_{0:k})$ and $q(\mathbf{x}_{0:k}|\mathbf{y}_{0:k})$; finally, an optional resampling step aims at struggling against the degeneration of the weights by discarding particles with weak weights and replicating those with large weights. After this step, all particles are weighted uniformly with $1/N$.

In this paper, we do not discuss the choice of the importance distribution $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{0:k})$ [2] [3] nor that of the resampling scheme [4] [5], but we rather revisit the combination of the sampling, weighting and resampling steps as a unique step. Starting from a set $\{w_{k-1}^i, \mathbf{x}_{k-1}^i\}_{i=1}^N$ which approximates $p_{k-1|k-1}$ and from a given importance distribution $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{0:k})$, we first recall that the classical (multinomial) resampling step produces a conditionally dependent set $\{\frac{1}{N}, \mathbf{x}_k^i\}_{i=1}^N$. Alternately, we propose a scheme which produces a properly weighted (from an IS point of view) set of conditionally independent particles $\{\overline{w}_k^i, \overline{\mathbf{x}}_k^i\}_{i=1}^N$, and in which particles $\mathbf{x}_k^i$ and $\overline{\mathbf{x}}_k^i$ share the same marginal distribution. The paper is organized as follows. In section 2 we revisit the original (static) SIR mechanism of Rubin [6] and we display the distribution from which are drawn the final samples; we improve this mechanism from IS considerations, and apply it to SMC algorithms in section 3; section 4 is devoted to simulations, and we end this paper with a conclusion.

2. IMPROVED SIR MECHANISM: THE STATIC CASE

2.1. Our method

Let $\pi(\mathbf{x}) \propto p(\mathbf{x})$ be a pdf known up to a constant. Our aim is to obtain M samples according to $\pi(\mathbf{x})$ and/or to compute moment

$$\Theta = \int f(\mathbf{x})\pi(\mathbf{x})\mathrm{d}\mathbf{x}. \quad (2)$$

If sampling directly from $\pi(\mathbf{x})$ is impossible, one can resort to IS and next to resampling. Let $q(\mathbf{x})$ be an importance distribution. Rubin's SIR mechanism consists of three steps [6]:

S. for all i, $1 \leq i \leq N$, $\tilde{\mathbf{x}}^i \sim q(\mathbf{x})$;

W. for all i, $1 \leq i \leq N$, $w^i \propto p(\tilde{\mathbf{x}}^i)/q(\tilde{\mathbf{x}}^i)$, $\sum_{i=1}^N w^i = 1$;

R. for all i, $1 \leq i \leq M$, $\mathbf{x}^i \sim \sum_{j=1}^N w^j \delta_{\tilde{\mathbf{x}}^j}$.

When $N \to \infty$, each sample $\mathbf{x}^i$ becomes drawn from $\pi(\mathbf{x})$ [7]. Of course, if the objective is to compute an estimate of Θ in (2), it is well known that one should use the weighed set $\{w^i, \tilde{\mathbf{x}}^i\}_{i=1}^N$ rather than the resampled one $\{\frac{1}{M}, \mathbf{x}^i\}_{i=1}^M$. However, in the particle filtering (PF) context the set $\{\frac{1}{M}, \mathbf{x}^i\}_{i=1}^M$ plays a key role since it struggles against the degeneration of the weights and aims at recreating diversity for the subsequent time iterations; consequently we here focus on the statistical

properties of the resampled set $\{\mathbf{x}^i\}_{i=1}^M$. For a given N, each sample $\mathbf{x}^i$ is indeed distributed according to

$$\tilde{q}(\mathbf{x}) = Nh(\mathbf{x})q(\mathbf{x}), \quad (3)$$

$$h(\mathbf{x}) = \int \frac{\frac{p(\mathbf{x})}{q(\mathbf{x})}}{\frac{p(\mathbf{x})}{q(\mathbf{x})} + \sum_{i=1}^{N-1} \frac{p(\mathbf{z}^i)}{q(\mathbf{z}^i)}} \prod_{i=1}^{N-1} q(\mathbf{z}^i) \mathrm{d}\mathbf{z}^i. \quad (4)$$

However, it is clear that the resampling step introduces dependency among the final samples. At this point, two remarks naturally arise:

1. if we had at our disposal a set $\{\overline{\mathbf{x}}^i\}_{i=1}^M$ of i.i.d. samples drawn from $\tilde{q}(\mathbf{x})$, then for any function $f(.)$ the MC estimate $\widetilde{\Theta} = \sum_{i=1}^M \frac{1}{M} f(\overline{\mathbf{x}}^i)$ of (2) would outperfom that built from the SIR algorithm $\widehat{\Theta} = \sum_{i=1}^M \frac{1}{M} f(\mathbf{x}^i)$. Indeed, both estimates have the same mean but [8]

$$\mathrm{var}(\widehat{\Theta}) = \mathrm{var}(\widetilde{\Theta}) + \frac{M-1}{M} \mathrm{var}\left(\sum_{k=1}^N w^k f(\tilde{\mathbf{x}}^k) \right); \quad (5)$$

2. although $\widetilde{\Theta}$ has a lower variance than $\widehat{\Theta}$, it may not be optimal to weight i.i.d. samples $\{\overline{\mathbf{x}}^i\}_{i=1}^M$ with $1/M$; from an IS perspective, $\overline{\mathbf{x}}^i$ should be weighted by $\overline{w}^i \propto p(\overline{\mathbf{x}}^i)/\tilde{q}(\overline{\mathbf{x}}^i)$, $\sum_{i=1}^M \overline{w}^i = 1$. However, a closed form expression of $\tilde{q}$ in (3) is not available.

These two points are actually connected. Indeed, let us start with the second one: a natural way to approximate integral $h(\mathbf{x})$ (and finally $\tilde{q}(\mathbf{x})$) in (4) is to consider a local MC method which relies on M samples drawn from $\prod_{i=1}^{N-1} q(\mathbf{z}^i)$. So let $\{\tilde{\mathbf{x}}^{i,j}\}$, with $1 \leq i \leq N$ and $1 \leq j \leq M$, be $M \times N$ i.i.d. samples drawn from $q(.)$ (the reason why we take $M \times N$ and not $M \times (N-1)$ will be clearer below). An MC estimate of $\tilde{q}(\mathbf{x})$ reads

$$\hat{q}(\mathbf{x}) = q(\mathbf{x}) \times \frac{N}{M} \sum_{j=1}^M \frac{\frac{p(\mathbf{x})}{q(\mathbf{x})}}{\frac{p(\mathbf{x})}{q(\mathbf{x})} + \sum_{i=1}^{N-1} \frac{p(\tilde{\mathbf{x}}^{i,j})}{q(\tilde{\mathbf{x}}^{i,j})}}. \quad (6)$$

Next it is possible to produce M i.i.d. samples drawn from $\tilde{q}(\mathbf{x})$ by applying M SIR steps, but resampling one particle per step:

W. for all (i,j), $1 \leq i \leq N$, $1 \leq j \leq M$, set $w^{i,j} \propto p(\tilde{\mathbf{x}}^{i,j})/q(\tilde{\mathbf{x}}^{i,j})$, $\sum_{i=1}^N w^{i,j} = 1$;

R. for all j, $1 \leq j \leq M$, $\overline{\mathbf{x}}^j \sim \sum_{i=1}^N w^{i,j} \delta_{\tilde{\mathbf{x}}^{i,j}}$.

In summary, we get a set of M samples $\{\overline{\mathbf{x}}^j\}_{j=1}^M$ drawn independently from $\tilde{q}(.)$. The $M(N-1)$ extra samples that were drawn for producing that set are next recycled to approximately compute $\tilde{q}(\overline{\mathbf{x}}^j)$, which in turn enables us to weight particles $\{\overline{\mathbf{x}}^j\}_{j=1}^M$.

2.2. An illustrating example

In this section, we set $N = M$ and we illustrate the gain induced by using independent resampling, and moreover by post-weighting the resampled particles with $\frac{p}{\tilde{q}}$ weights rather than with uniform weights. First note that our independent resampling scheme requires an extra computational cost as compared to the classical SIR mechanism. It requires N^2 sampling steps and N resampling ones, while the classical SIR mechanism needs N sampling steps and N resampling ones. Consequently, we will also compare our method with a classical SIR procedure in which the final set of N points is resampled from N^2 intermediate samples. Note that in this case, the SIR algorithm produces N points identically (but not independently) distributed according to pdf $\tilde{q}$ in (3)-(4) but in which N is replaced by N^2. As a toy example let us consider a scalar linear Gaussian model $p(x) = \mathcal{N}(0, Q_G)$, $p(y|x) = \mathcal{N}(x, R_G)$, and $\pi(x) = p(x|y) \propto p(x)p(y|x)$. We average the RMSE values over 1000 MC runs.

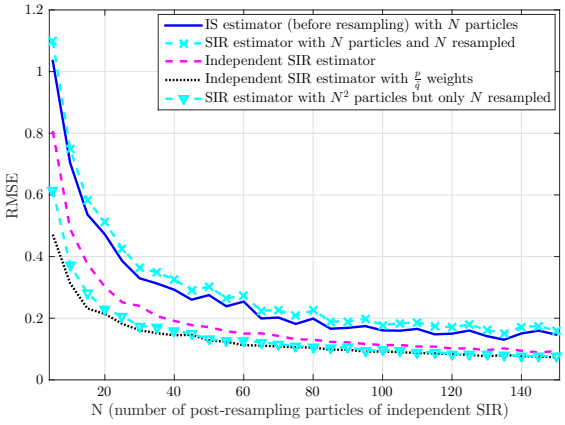

Fig. 1. Linear Gaussian model with $Q_G = 5$ and $R_G = 1$. Independent SIR algorithms use N^2 intermediate particles and independently resample N of them to build the estimators.

Figure 1 first shows, as expected, that replacing the dependent samples produced by the classical SIR mechanism by independent samples drawn from $\tilde{q}$ improves the performance of the MC estimator of (2). Moreover, reweighting these independent samples with (approximated) $\frac{p}{\tilde{q}}$ weights further improves the performance. Other simulations have shown that with lower process noise variance Q_G or higher measurement noise variance R_G, the gain induced by using $\frac{p}{\tilde{q}}$ weights rather than uniform weights decreases. On the other hand, when reducing the measurement noise, the gain increases, up to a certain point where the $\frac{p}{\tilde{q}}$ weights become poorly estimated; in which case the performance of the independent SIR estimator with $\frac{p}{\tilde{q}}$ weights drops. Moreover, it can be seen that in this model a dependent SIR estimator using N^2 particles but resampling only N of them will have a better performance than

an independent SIR estimator with uniform weights; yet the independent SIR estimator using post-resampling weights $\frac{p}{\tilde{q}}$ remains the best solution.

3. APPLICATION TO SMC ALGORITHMS

3.1. The classical SIR algorithm

Let us start from $\{w_{k-1}^i, \mathbf{x}_{k-1}^i\}_{i=1}^N$ which approximates $p(\mathbf{x}_{k-1}|\mathbf{y}_{0:k-1})$. The classical SIR algorithm is an adaptation of Rubin's SIR mechanism recalled in section 2.1. The main difference is that the IS distribution is now a conditional IS one $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{0:k})$. We introduce the following notations: let

$$p_i(\mathbf{x}) = w_{k-1}^i f_{k|k-1}(\mathbf{x}|\mathbf{x}_{k-1}^i) g_k(\mathbf{y}_k|\mathbf{x}), \quad (7)$$

$$q_i(\mathbf{x}) = q(\mathbf{x}|\mathbf{x}_{k-1}^i, \mathbf{y}_{0:k}). \quad (8)$$

The SIR algorithm consists of the three following steps:

S. for all i, $1 \leq i \leq N$, sample $\tilde{\mathbf{x}}_k^i \sim q_i(\mathbf{x})$;

W. for all i, $1 \leq i \leq N$, set $w_k^i \propto \frac{p_i(\tilde{\mathbf{x}}_k^i)}{q_i(\tilde{\mathbf{x}}_k^i)}$, $\sum_{i=1}^N w_k^i = 1$;

R. for all i, $1 \leq i \leq N$, sample $\mathbf{x}_k^i \sim \sum_{j=1}^N w_k^j \delta_{\tilde{\mathbf{x}}_k^j}$.

The final set $\{\mathbf{x}_k^i\}_{i=1}^N$ is then approximately sampled from $p(\mathbf{x}_k|\mathbf{y}_{0:k})$ [9] [10].

Let us now adapt section 2.1 to the SMC context. Given $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$, each $\mathbf{x}_k^i$ is now drawn from $\tilde{q}_k(\mathbf{x}|\{\mathbf{x}_{k-1}^i\}_{i=1}^N)$ (or $\tilde{q}_k(\mathbf{x})$ in shorthand notation) where

$$\tilde{q}_k(\mathbf{x}) = \sum_{i=1}^N h_i(\mathbf{x}) q_i(\mathbf{x}), \quad (9)$$

$$h_i(\mathbf{x}) = \int \frac{p_i(\mathbf{x})/q_i(\mathbf{x})}{\frac{p_i(\mathbf{x})}{q_i(\mathbf{x})} + \sum_{j\neq i} \frac{p_j(\mathbf{x}^j)}{q_j(\mathbf{x}^j)}} \prod_{j\neq i} q_j(\mathbf{x}^j) \mathrm{d}\mathbf{x}^1 \cdots \mathrm{d}\mathbf{x}^j. \quad (10)$$

3.2. Our improved SIR algorithm

Our objective is to obtain samples from $\tilde{q}_k$ which are i) conditionally independent and ii) properly weighted. Let us begin with the first point. So let $\{\overline{w}_{k-1}^i, \overline{\mathbf{x}}_{k-1}^i\}_{i=1}^M$ be an approximation of $p_{k-1|k-1}$ (we use variable M to distinguish from the number of particles N of the classical SIR algorithm). The following procedure produces samples which are independent given $\{\overline{w}_{k-1}^i, \overline{\mathbf{x}}_{k-1}^i\}_{i=1}^M$:

S. for all (i,j), $1 \leq i,j \leq M$, sample $\tilde{\mathbf{x}}_k^{i,j} \sim q_i(\mathbf{x}_k)$

W. for all (i,j), $1 \leq i,j \leq M$, set $w_k^{i,j} \propto \frac{p_i(\tilde{\mathbf{x}}_k^{i,j})}{q_i(\tilde{\mathbf{x}}_k^{i,j})}$, $\sum_{i=1}^M w_k^{i,j} = 1$;

R. for all j, $1 \leq j \leq M$, sample $\overline{\mathbf{x}}_k^j \sim \sum_{l=1}^M w_k^{l,j} \delta_{\tilde{\mathbf{x}}_k^{l,j}}$.

Remark 1 Let us note that our approach is clearly distinct of the "local resampling" of [11] in which one particle from each $\{q_i\}_{i=1}^M$ is always selected at the *R* step.

Remark 2 One can easily adapt (5) to the sequential case; so starting from a common set $\{\overline{w}_{k-1}^i, \overline{\mathbf{x}}_{k-1}^i\}_{i=1}^M$, the estimate deduced from the unweighted and conditionally independent set $\{\overline{\mathbf{x}}_k^j\}_{j=1}^M$ has the same conditional mean but a lower conditional variance than that deduced from the set produced by the classical SIR algorithm $\{\mathbf{x}_k^j\}_{j=1}^M$.

Let us now discuss point ii) above, i.e. how to properly weight these independent samples. Computing $\tilde{q}_k(.)$ directly would require a large computational cost due to the sum in (9). However one can use the "auxiliary trick" (see e.g. [12] [13]). Observe that our final importance distribution $\tilde{q}_k(\mathbf{x})$ can be rewritten as

$$\tilde{q}_k(\mathbf{x}) = \sum_{i=1}^M \int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x} \times \frac{h_i(\mathbf{x}) q_i(\mathbf{x})}{\int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x}}. \quad (11)$$

So $\tilde{q}_k(\mathbf{x})$ coincides with a mixture with weights $\int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x}$ (the i^{th} weight is also the expectation of the i-th normalized importance weight given $\{\overline{\mathbf{x}}_{k-1}^i\}$) and pdf components $\frac{h_i(\mathbf{x}) q_i(\mathbf{x})}{\int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x}}$. Consequently, the final particles $\{\overline{\mathbf{x}}_k^j\}_{j=1}^M$ can be seen as the result of a sampling mechanism, in augmented dimension, from

$$\int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x} \times \frac{h_i(\mathbf{x}) q_i(\mathbf{x})}{\int h_i(\mathbf{x}) q_i(\mathbf{x}) \mathrm{d}\mathbf{x}}.$$

So let us consider an auxiliary variable i^j associated to our sample $\overline{\mathbf{x}}_k^j$ such that $\Pr(i^j = l|\{\tilde{\mathbf{x}}_k^{n,j}\}_{n=1}^M) = w_k^{l,j}$. The final weight $\overline{w}_k^j$ associated to $\overline{\mathbf{x}}_k^j$ reads [12]

$$\overline{w}_k^j = \frac{\overline{w}_{k-1}^{i^j} f_{k|k-1}(\overline{\mathbf{x}}_k^j|\overline{\mathbf{x}}_{k-1}^{i^j}) g_k(\mathbf{y}_k|\overline{\mathbf{x}}_k^j)}{h_{i^j}(\overline{\mathbf{x}}_k^j) q_{i^j}(\overline{\mathbf{x}}_k^j)} = \frac{p_{i^j}(\overline{\mathbf{x}}_k^j)}{h_{i^j}(\overline{\mathbf{x}}_k^j) q_{i^j}(\overline{\mathbf{x}}_k^j)}.$$

As in the static case, $h_i(\mathbf{x})$ can be estimated by recycling the M^2 intermediate samples $\tilde{\mathbf{x}}^{i,j}$,

$$h_i(\mathbf{x}) \approx \frac{1}{M} \sum_{l=1}^M \frac{p_i(\mathbf{x})/q_i(\mathbf{x})}{\frac{p_i(\mathbf{x})}{q_i(\mathbf{x})} + \sum_{j\neq i} \frac{p_j(\tilde{\mathbf{x}}_k^{j,l})}{q_j(\tilde{\mathbf{x}}_k^{j,l})}}. \quad (12)$$

Remark 3 If the conditional importance distribution is the so called optimal one, i.e. if $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{0:k}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k)$, then our algorithm reduces to the fully adapted auxiliary PF (see [12]). In that case $h_i(\mathbf{x}) = p(\mathbf{y}_k|\mathbf{x}_{k-1}^i)$ and no longer depends on $\mathbf{x}$. Remember however that in HMC models (1), $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k)$ is generally not available.

4. SIMULATIONS

We now validate our algorithm via simulations, and in particular highlight under which conditions using post-resampling

weights is beneficial in the dynamic case. As far as computational cost is concerned, the classical (resp. independent) SIR algorithm uses N (resp. M^2) sampling steps and N (resp. M) resampling ones in order to obtain an approximation of size N (resp. M). So we compare the algorithms with M and N satisfying $M^2 + M = 2N$, in order for them to have the same overall sampling/resampling budget. Figures display the RMSE between our estimators and a reference estimator calculated using a classical sequential SIR algorithm with 10^6 particles. We consider $T = 10$ time steps and average our RMSE values over these time steps as well as over 1000 MC runs.

Let us consider a target tracking model with polar coordinates

$$\begin{aligned} \mathbf{x}_t &= \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{u}_t \\ \mathbf{y}_t &= \begin{pmatrix} \sqrt{x_{x,t}^2 + x_{y,t}^2} \\ \arctan \frac{x_{y,t}}{x_{x,t}} \end{pmatrix} + \mathbf{v}_t \end{aligned}$$

where $\mathbf{u}_k \sim \mathcal{N}(\mathbf{0}_4, \mathbf{Q})$ and $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}_2, \mathbf{R})$ with

$$\mathbf{Q} = \sigma_Q^2 \begin{pmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{pmatrix}, \mathbf{R} = \begin{pmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{pmatrix}.$$

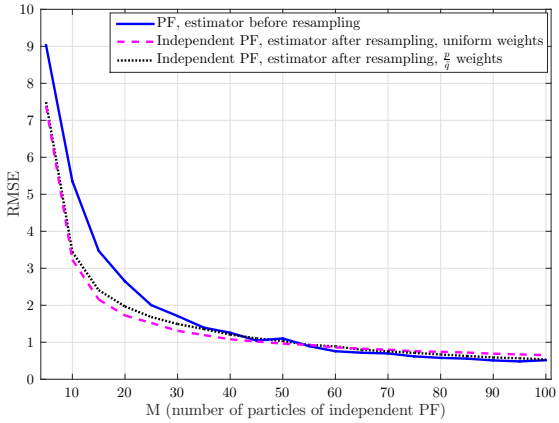

Fig. 2. Polar target tracking model with $\sigma_Q = 2$, $\sigma_\rho = 0.1$ and $\sigma_\theta = \frac{\pi}{1800}$. The RMSE is displayed with respect to the number of particles used for independent PF; the dependent PF uses $N = \frac{M^2+M}{2}$ particles so that computational costs are comparable.

Figure 2 compares a dependent PF using $\frac{N^2+N}{2}$ particles to an independent PF with N particles. In this scenario measurements are very informative; as a result, for a low number of particles the independent PF keeps a good particle diversity, and thus outperforms the dependent resampling method

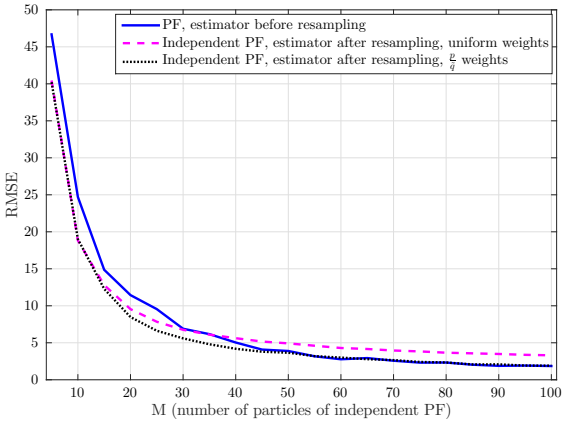

Fig. 3. Polar target tracking model with $\sigma_Q = 10$, $\sigma_\rho = 1$ and $\sigma_\theta = \frac{\pi}{180}$. The RMSE is displayed with respect to the number of particles used for independent PF; the dependent PF uses $N = \frac{M^2+M}{2}$ particles so that computational costs are comparable.

which suffers from particle degeneracy; but on the other hand, it is difficult to approximate the integral in (10) and so the weights $\frac{p}{\tilde{q}}$ suffer from a poor quality of $\tilde{q}$ MC estimation.

On the other hand Figure 3 considers a scenario where measurements are not so informative, and in this case unweighted independent resampling yields less performance gain as compared to dependent resampling. However, here the $\frac{p}{\tilde{q}}$ weights are well estimated and as a result the $\frac{p}{\tilde{q}}$ estimator outperforms the classical PF estimator.

5. CONCLUSION

In this paper we proposed a resampling mechanism which enhances particle diversity in SMC algorithms. We first observed that the resampling step of an SIR algorithm produces unweighted and conditionally dependent particles drawn from some mixture distribution. Starting from this observation, our resampling mechanism produces samples which are drawn from that same mixture distribution, but are properly weighted (from an IS point of view) and conditionally independent. Even though these weights cannot be computed exactly, they can be approximated by a local MC method which recycles the extra samples used for producing the independent particles. Simulations showed that in models where the approximation based on this local MC method is valid, the properly weighted estimator outperforms the uniformly weighted one. In addition, when the number of samples is small in the classical SIR approach (100-1500 particles) our approach performs better for the same computational cost.

6. REFERENCES

[1] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Statistics for Engineering and Information Science. Springer, New York, 2001.

[2] 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.

[3] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," *Advances in Neural Information Processing Systems*, vol. 19, no. 2, pp. 567–76, June 2000.

[4] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[5] R. Douc, O. Cappé, and É. 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] D. B. Rubin, "Using the SIR algorithm to simulate posterior distributions," in *Bayesian Statistics III*, M. H. Bernardo, K. M. Degroot, D. V. Lindley, and A. F. M. Smith, Eds. Oxford University Press, Oxford, 1988.

[7] A. F. M. Smith and A. E. Gelfand, "Bayesian statistics without tears : a sampling-resampling perspective," *The American Statistician*, vol. 46, no. 2, pp. 84–87, 1992.

[8] Y. Petetin and F. Desbouvries, "Optimal SIR algorithm vs. fully adapted auxiliary particle filter: a non asymptotical analysis," *Statistics and computing*, vol. 23, no. 6, pp. 759–775, 2013.

[9] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/ non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, no. 2, pp. 107–113, April 1993.

[10] O. Cappé, É. Moulines, and T. Rydén, *Inference in Hidden Markov Models*, Springer-Verlag, 2005.

[11] V. Elvira, L. Martino, D. Luengo, and M. Bugallo, "Improving Population Monte Carlo: Alternative weighting and resampling schemes," *Vixra.org*, 2015, http://vixra.org/pdf/1601.0174v1.pdf.

[12] M. K. Pitt and N. Shephard, "Filtering via simulation : Auxiliary particle filter," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–99, June 1999.

[13] O. Cappé, S. J. Godsill, and É. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," *Proc. of the IEEE*, vol. 95, no. 5, pp. 899–924, May 2007.