

A NEW PARTICLE FILTERING ALGORITHM WITH STRUCTURALLY OPTIMAL IMPORTANCE FUNCTION

Boujemaa Ait-El-Fquih and François Desbouvries

GET / INT / Dépt. CITI and CNRS UMR 5157, 9 rue Charles Fourier, 91011 Evry, France

{Boujemaa.Ait-El-fquih, Francois.Desbouvries}@int-evry.fr

ABSTRACT

Bayesian estimation in nonlinear stochastic dynamical systems has been addressed for a long time. Among other solutions, Particle Filtering (PF) algorithms propagate in time a Monte Carlo (MC) approximation of the a posteriori filtering measure. However, a drawback of the classical PF algorithms is that the optimal conditional importance distribution (CID) is often difficult (or even impossible) to compute and to sample from. As a consequence, suboptimal sampling strategies have been proposed in the literature. In this paper we bypass this difficulty by rather considering the prediction sequential importance sampling (SIS) problem; the filtering MC approximation is obtained as a byproduct. The advantage of this prediction-PF method is that it combines optimality and simplicity, since for the prediction problem, the optimal CID happens to be the prior transition of the underlying Markov Chain (MC), from which it is often simple to sample from.

Index Terms— Particle Filtering, Sequential Importance Sampling, Optimal importance function, Hidden Markov Chains.

1. INTRODUCTION

Let us consider the following classical stochastic dynamical system :

$$\begin{cases} \mathbf{x}_{n+1} &= g_n(\mathbf{x}_n, \mathbf{u}_n) \\ \mathbf{y}_n &= h_n(\mathbf{x}_n, \mathbf{v}_n) \end{cases}, \quad (1)$$

in which $g_n(\cdot, \cdot)$ is some (possibly nonlinear) function from $\mathbb{R}^m \times \mathbb{R}^p$ to \mathbb{R}^m , $h_n(\cdot, \cdot)$ is some (possibly nonlinear) function from $\mathbb{R}^m \times \mathbb{R}^q$ to \mathbb{R}^q , and $\mathbf{u} = \{\mathbf{u}_n\}_{n \in \mathbb{N}}$ and $\mathbf{v} = \{\mathbf{v}_n\}_{n \in \mathbb{N}}$ are zero-mean sequences which are independent, jointly independent and independent of \mathbf{x}_0 .

Let $p(\mathbf{x}_n | \mathbf{y}_{0:n})$, say, denote the probability density function (pdf) of \mathbf{x}_n given $\mathbf{y}_{0:n} = \{\mathbf{y}_i\}_{i=0}^n$ (other pdf or conditional pdfs of interest are defined similarly). One can check that the following properties hold :

$$p(\mathbf{x}_{n+1} | \mathbf{x}_{0:n}) = p(\mathbf{x}_{n+1} | \mathbf{x}_n); \quad (2)$$

$$p(\mathbf{y}_{0:n} | \mathbf{x}_{0:n}) = \prod_{i=0}^n p(\mathbf{y}_i | \mathbf{x}_{0:n}); \quad (3)$$

$$p(\mathbf{y}_i | \mathbf{x}_{0:n}) = p(\mathbf{y}_i | \mathbf{x}_i) \text{ for all } i, 0 \leq i \leq n, \quad (4)$$

so (1) is often referred to as a Hidden Markov Chain (HMC).

In this paper we deal with the Bayesian filtering problem, which consists in recursively computing $p(\mathbf{x}_n | \mathbf{y}_{0:n})$ as new observations become available. From (2)-(4) we get

$$p(\mathbf{x}_{0:n} | \mathbf{y}_{0:n}) = \frac{p(\mathbf{x}_n | \mathbf{x}_{n-1}) p(\mathbf{y}_n | \mathbf{x}_n)}{p(\mathbf{y}_n | \mathbf{y}_{0:n-1})} p(\mathbf{x}_{0:n-1} | \mathbf{y}_{0:n-1}) \quad (5)$$

Consequently, the recursive propagation of $p(\mathbf{x}_n | \mathbf{y}_{0:n})$ is given by

$$p(\mathbf{x}_n | \mathbf{y}_{0:n}) = \frac{p(\mathbf{y}_n | \mathbf{x}_n) \int p(\mathbf{x}_n | \mathbf{x}_{n-1}) p(\mathbf{x}_{n-1} | \mathbf{y}_{0:n-1}) d\mathbf{x}_{n-1}}{p(\mathbf{y}_n | \mathbf{y}_{0:n-1})} \quad (6)$$

It remains to compute (6) in practice. In the linear Gaussian case the solution can be computed exactly by the Kalman filter. However, the exact recursive solution is difficult to compute in general, and consequently many approximate techniques have been developed. Among them, particle filters (see e.g. [1] [2] [3] [4] [5] [6] and references therein) are MC methods which aim at propagating an approximation of $p(d\mathbf{x}_n | \mathbf{y}_{0:n})$; such methods have found many applications (see e.g. [4] [5]) and have proven to be very efficient in practice.

However, a drawback of the classical PF algorithms is that the optimal conditional importance distribution (CID), from which one should sample particles from, is often difficult (or even impossible) to compute. In this paper we propose to compute the filtering measure indirectly, via the prediction SIS problem. As we shall see, the advantage of this prediction-PF method is that it combines both optimality and simplicity, since for the prediction problem, the optimal CID happens to be the prior transition of the MC $\mathbf{x} = \{\mathbf{x}_n\}$, from which it is often simple to sample from. Moreover computing the incremental importance weight also becomes straightforward.

This paper is organized as follows. In section 2 we first briefly recall the classical PF algorithms. Next in section 3 we introduce the prediction-PF algorithm. Simulations are performed in section 4, and we end the paper with some concluding remarks.

2. THE CLASSICAL PF ALGORITHMS

2.1. The generic PF algorithm

Let us recall the principle of particle filtering [4] [5] [6]. Assume that at time $n - 1$ we have a discrete random measure which approximates $p(d\mathbf{x}_{0:n-1}|\mathbf{y}_{0:n-1})$:

$$p(d\mathbf{x}_{0:n-1}|\mathbf{y}_{0:n-1}) \simeq \sum_{i=1}^N \kappa_{n-1}^i \delta_{\mathbf{x}_{0:n-1}^i}(d\mathbf{x}_{0:n-1}), \quad (7)$$

where δ_x is the Dirac measure at point x , the samples $\mathbf{x}_{0:n-1}^i$ are generated from an importance distribution $q(\mathbf{x}_{0:n-1}|\mathbf{y}_{0:n-1})$, and the importance weight κ_{n-1}^i associated to the i -th trajectory $\mathbf{x}_{0:n-1}^i$ is given by

$$\kappa_{n-1}^i \propto \frac{p(\mathbf{x}_{0:n-1}^i|\mathbf{y}_{0:n-1})}{q(\mathbf{x}_{0:n-1}^i|\mathbf{y}_{0:n-1})}, \quad \sum_{i=1}^N \kappa_{n-1}^i = 1. \quad (8)$$

Let us see how to compute (7) recursively. We first consider the updating of the trajectories. If we assume that the importance distribution factorizes as

$$q(\mathbf{x}_{0:n}|\mathbf{y}_{0:n}) = q(\mathbf{x}_n|\mathbf{x}_{0:n-1}, \mathbf{y}_{0:n})q(\mathbf{x}_{0:n-1}|\mathbf{y}_{0:n-1}),$$

i.e. that $q(\mathbf{x}_{0:n}|\mathbf{y}_{0:n})$ admits $q(\mathbf{x}_{0:n-1}|\mathbf{y}_{0:n-1})$ as marginal, then for all $1 \leq i \leq N$, $[\mathbf{x}_{0:n}^i] = [\mathbf{x}_{0:n-1}^i, \mathbf{x}_n^i]$, in which each particle \mathbf{x}_n^i is sampled from the CID $q(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n})$. As for the weights κ_n^i , we see from (5) that they can be computed recursively as

$$\kappa_n^i \propto \underbrace{\frac{p(\mathbf{x}_n^i|\mathbf{x}_{n-1}^i)p(\mathbf{y}_n|\mathbf{x}_n^i)}{q(\mathbf{x}_n^i|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n})}}_{\lambda_n^i} \times \underbrace{\frac{p(\mathbf{x}_{0:n-1}^i|\mathbf{y}_{0:n-1})}{q(\mathbf{x}_{0:n-1}^i|\mathbf{y}_{0:n-1})}}_{\propto \kappa_{n-1}^i}. \quad (9)$$

Finally $\sum_{i=1}^N \kappa_n^i \delta_{\mathbf{x}_{0:n}^i}(d\mathbf{x}_{0:n})$ approximates $p(d\mathbf{x}_{0:n}|\mathbf{y}_{0:n})$, and thus $\sum_{i=1}^N \kappa_n^i \delta_{\mathbf{x}_n^i}(d\mathbf{x}_n)$ approximates $p(d\mathbf{x}_n|\mathbf{y}_{0:n})$.

2.2. Practical considerations

Now, SIS algorithms are well known to suffer from weights degeneracy. It has thus proved important in the above generic algorithm to resample from $\sum_{i=1}^N \kappa_n^i \delta_{\mathbf{x}_n^i}(d\mathbf{x}_n)$ (either systematically or according to some strategy), and also to choose the CID $q(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n})$ carefully.

To that respect, sampling from the *a priori* transition kernel of the Markov chain \mathbf{x} (i.e., choosing $q(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n}) = p(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i) = p(\mathbf{x}_n|\mathbf{x}_{n-1}^i)$, which was the original choice in the so-called bootstrap filter [1]) is popular because sampling from $p(\mathbf{x}_n|\mathbf{x}_{n-1}^i)$ is often straightforward. Moreover, computing the incremental weight λ_n^i in (9) reduces to evaluating the conditional likelihood of the new observation given the updated particle position.

However, this choice of the prior density can often lead to poor performances, and indeed the best choice is to sample

the particles from the optimal CID $q^{opt}(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n})$ [7], i.e. the distribution which minimizes the variance of the importance weights κ_n^i , conditionally on the observations $\mathbf{y}_{0:n}$ and past samples $\mathbf{x}_{0:n-1}^i$. One can see easily that q^{opt} is the *a posteriori* transition kernel of \mathbf{x} (which, conditionally on the observations \mathbf{y} , remains a Markov Chain):

$$\begin{aligned} q^{opt}(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n}) &= p(\mathbf{x}_n|\mathbf{x}_{0:n-1}^i, \mathbf{y}_{0:n}) \\ &= p(\mathbf{x}_n|\mathbf{x}_{n-1}^i, \mathbf{y}_{0:n}) \\ &= p(\mathbf{x}_n|\mathbf{x}_{n-1}^i, \mathbf{y}_n). \end{aligned} \quad (10)$$

For this choice of the CID, the incremental weight λ_n^i in (9) becomes

$$\lambda_n^i = p(\mathbf{y}_n|\mathbf{x}_{n-1}^i). \quad (11)$$

Now, sampling from (10) (or even computing this CID) is impossible in most cases, and computing (11) is not easy either. As a consequence many efforts have been expended in order to approximate this distribution (see e.g. [7] [8] [9] [5] [6]). As we shall see in the next section, it is indeed possible to bypass this difficulty and to re-design the PF algorithm in such a way that the optimal becomes the prior.

3. THE PREDICTION-PF ALGORITHM

Equation (6) describes the transition $p(\mathbf{x}_{n-1}|\mathbf{y}_{0:n-1}) \rightarrow p(\mathbf{x}_n|\mathbf{y}_{0:n})$ as the succession of two steps. We first predict the future value \mathbf{x}_n , based on the same amount of data (Markovian step $p(\mathbf{x}_{n-1}|\mathbf{y}_{0:n-1}) \rightarrow p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) = \int p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}|\mathbf{y}_{0:n-1})d\mathbf{x}_{n-1}$); and we next incorporate the new data \mathbf{y}_n (Bayesian step $p(\mathbf{x}_n|\mathbf{y}_{0:n-1}) \rightarrow p(\mathbf{x}_n|\mathbf{y}_{0:n}) \propto p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{y}_{0:n-1})$). However, it is well known in Kalman filtering that these steps are interchangeable. Let us thus see what happens in PF if we begin with the prediction problem, i.e. if we interchange the prediction and filtering steps.

3.1. The generic Prediction-PF algorithm

Similarly to section 2.1, particle *prediction* can be set as follows. From model (1) we get

$$p(\mathbf{x}_{0:n+1}|\mathbf{y}_{0:n}) = \frac{p(\mathbf{x}_{n+1}|\mathbf{x}_n)p(\mathbf{y}_n|\mathbf{x}_n)}{p(\mathbf{y}_n|\mathbf{y}_{0:n-1})}p(\mathbf{x}_{0:n}|\mathbf{y}_{0:n-1}), \quad (12)$$

which should be compared to (5). Let us thus start from

$$p(d\mathbf{x}_{0:n}|\mathbf{y}_{0:n-1}) \simeq \sum_{i=1}^N w_n^i \delta_{\mathbf{x}_{0:n}^i}(d\mathbf{x}_{0:n}), \quad (13)$$

in which samples $\mathbf{x}_{0:n}^i$ are generated from $q(\mathbf{x}_{0:n}|\mathbf{y}_{0:n-1})$, and the prediction importance weights w_n^i now satisfy

$$w_n^i \propto p(\mathbf{x}_{0:n}^i|\mathbf{y}_{0:n-1})/q(\mathbf{x}_{0:n}^i|\mathbf{y}_{0:n-1}).$$

The importance function should be chosen such that

$$q(\mathbf{x}_{0:n+1}|\mathbf{y}_{0:n}) = q(\mathbf{x}_{n+1}|\mathbf{x}_{0:n}, \mathbf{y}_{0:n})q(\mathbf{x}_{0:n}|\mathbf{y}_{0:n-1}).$$

As for the updating of the importance weights, (9) becomes

$$w_{n+1}^i \propto \underbrace{\frac{p(\mathbf{x}_{n+1}^i|\mathbf{x}_n^i)p(\mathbf{y}_n|\mathbf{x}_n^i)}{q(\mathbf{x}_{n+1}^i|\mathbf{x}_{0:n}^i, \mathbf{y}_{0:n})}}_{\lambda_n^i} \times \underbrace{\frac{p(\mathbf{x}_{0:n}^i|\mathbf{y}_{0:n-1})}{q(\mathbf{x}_{0:n}^i|\mathbf{y}_{0:n-1})}}_{\propto w_n^i}. \quad (14)$$

Finally $\sum_{i=1}^N w_{n+1}^i \delta_{\mathbf{x}_{0:n+1}^i}(d\mathbf{x}_{0:n+1})$ approximates $p(d\mathbf{x}_{0:n+1}|\mathbf{y}_{0:n})$, and thus $p(d\mathbf{x}_{n+1}|\mathbf{y}_{0:n}) \simeq \sum_{i=1}^N w_{n+1}^i \delta_{\mathbf{x}_{n+1}^i}(d\mathbf{x}_{n+1})$, but also $p(d\mathbf{x}_n|\mathbf{y}_{0:n}) \simeq \sum_{i=1}^N w_{n+1}^i \delta_{\mathbf{x}_n^i}(d\mathbf{x}_n)$.

3.2. Optimal importance function

Similarly to section 2.2, one should also choose the CID $q(\mathbf{x}_{n+1}|\mathbf{x}_{0:n}, \mathbf{y}_{0:n})$ properly. It is easy to show the following result.

Proposition 1 *For the one-step ahead prediction problem, the posterior importance function :*

$$q(\mathbf{x}_{n+1}|\mathbf{x}_{0:n}^i, \mathbf{y}_{0:n}) = p(\mathbf{x}_{n+1}|\mathbf{x}_{0:n}^i, \mathbf{y}_{0:n}) \quad (15)$$

$$= p(\mathbf{x}_{n+1}|\mathbf{x}_n^i) \quad (16)$$

is the CID which minimizes the variance of weight w_{n+1}^i conditionally upon $\mathbf{x}_{0:n}^i$ and $\mathbf{y}_{0:n}$. Under this choice of the importance function, the incremental weight λ_n^i in (14) becomes

$$\lambda_n^i = p(\mathbf{y}_n|\mathbf{x}_n^i). \quad (17)$$

The main interest of the above algorithm is that *in the prediction problem, the optimal importance function coincides with the prior transition pdf of the Markov chain \mathbf{x}* , from which it is often easy to sample from. Computing the incremental weight via (17) is also much easier than via (11), since $p(\mathbf{y}_n|\mathbf{x}_n)$ is indeed nothing but the HMC elementary observational transition in (4). Stated otherwise, we cumulate the advantages of choosing both the prior (for practical issues) and the posterior (for optimality results) CIDs, since in the prediction problem both transitions coincide. Let us now summarize the above discussion in the following algorithm.

The Prediction-Particle Filter.

Let $p(d\mathbf{x}_n|\mathbf{y}_{0:n-1}) \simeq \sum_{i=1}^N w_n^i \delta_{\mathbf{x}_n^i}(d\mathbf{x}_n)$.

Prediction.

- For $i = 1, \dots, N$, sample $\mathbf{x}_{n+1}^i \sim p(\mathbf{x}_{n+1}|\mathbf{x}_n^i)$;
- For $i = 1, \dots, N$, compute $w_{n+1}^i \propto p(\mathbf{y}_n|\mathbf{x}_n^i) \times w_n^i$, $\sum_{i=1}^N w_{n+1}^i = 1$;
- Resample (if necessary) from $\sum_{i=1}^N w_{n+1}^i \delta_{\mathbf{x}_{n+1}^i}(d\mathbf{x}_{n+1})$;

- Then $p(d\mathbf{x}_{n+1}|\mathbf{y}_{0:n}) \simeq \sum_{i=1}^N w_{n+1}^i \delta_{\mathbf{x}_{n+1}^i}(d\mathbf{x}_{n+1})$.

Filtering.

- For $i = 1, \dots, N$, set $\kappa_n^i = w_{n+1}^i$;
- Then $p(d\mathbf{x}_n|\mathbf{y}_{0:n}) \simeq \sum_{i=1}^N \kappa_n^i \delta_{\mathbf{x}_n^i}(d\mathbf{x}_n)$.

4. SIMULATIONS

In this section we perform computer simulations in order to validate our Prediction-PF algorithm. We consider the scalar model

$$x_{n+1} = 0.5x_n + 25x_n/(1+x_n^2) \quad (18)$$

$$+ 8 \cos(1.2(n+1)) + u_n,$$

$$y_n = x_n^2/20 + v_n. \quad (19)$$

We set $x_0 \sim \mathcal{N}(0, 1)$, $u_n \sim \mathcal{N}(0, 10)$ and $v_n \sim \mathcal{N}(0, 1)$. In this experiment we take $N = 500$ particles, and we resample whenever $N_{eff} \stackrel{def}{=} \frac{1}{\sum_{i=1}^N (w_n^i)^2} < \sigma$ with $\sigma = 0.3N$. The plots are averaged over $P = 50$ realizations. Figs. 1 and 2 respectively display the tracking of the true state by the Prediction-PF algorithm and the empirical variance. As expected, in this model the Prediction-PF algorithm tracks the state quite well.

We finally compute the empirical standard deviation \mathcal{J}_N defined as

$$\mathcal{J}_N = \frac{1}{M} \sum_{n=1}^M \left(\frac{1}{P} \sum_{j=1}^P (\hat{x}_{n|n}(j) - x_n(j))^2 \right)^{1/2}, \quad (20)$$

in which j denotes the realization and n is the running time index with $0 \leq n \leq M$ and $M = 50$. The estimate $\hat{x}_{n|n}(j)$ is computed as $\sum_{i=1}^N \kappa_n^i x_n^i(j)$. Table 1 displays \mathcal{J}_N as a function of the number of particles N .

| N | \mathcal{J}_N |
|-----|-----------------|
| 100 | 5.17 |
| 250 | 4.57 |
| 500 | 4.55 |
| 700 | 4.16 |

Table 1. Empirical standard deviation as a function of the number of particles.

5. CONCLUSION

In this paper we proposed the prediction-PF algorithm as a new SIS solution to the non linear Bayesian filtering problem. Instead of directly propagating in time a discrete random approximation of the filtering measure $p(d\mathbf{x}_n|\mathbf{y}_{0:n})$,

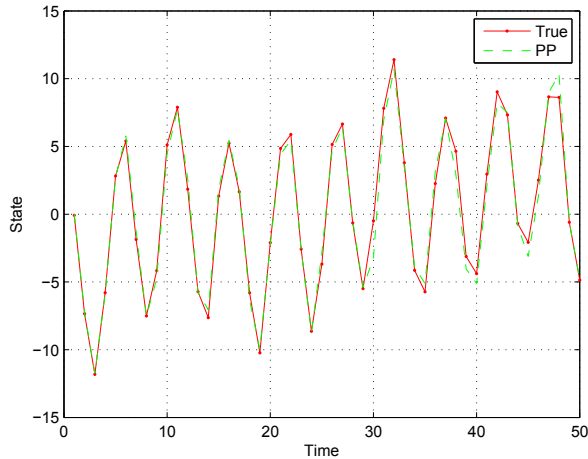


Fig. 1. true and estimated states

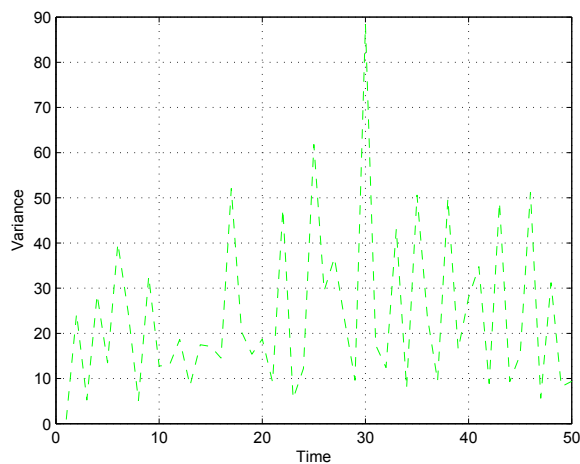


Fig. 2. empirical variance

we propagate MC approximations of the prediction measure $p(dx_n | y_{0:n-1})$; an MC approximation of $p(dx_n | y_{0:n})$ is obtained as a byproduct of this recursion. The main advantage of this (apparently indirect) solution is that for the prediction SIS problem, the optimal CID coincides with the elementary a priori transition kernel of the Markov chain x from which it is often simple to sample from. Also, computing the associated incremental importance weight becomes straightforward. The Prediction-PF algorithm is thus a very simple alternative to PF algorithms with suboptimal CID. Finally simulations confirmed the good tracking results expected by the method.

6. REFERENCES

- [1] 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.
- [2] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Statistics for Engineering and Information Science. 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 Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002.
- [4] P.M. Djuric, J.H. Kotecha, Jianqui Zhang, Yufei Huang, T. Ghirmai, M.F. Bugallo, and J. Miguez, "Particle filtering," *IEEE Signal Processing Magazine*, 2003.
- [5] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*, Springer-Verlag, 2005.
- [6] O. Cappé, S. J. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 899–924, May 2007.
- [7] 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.
- [8] Y. Huang and P. M. Djurić, "A new importance function for particle filtering and its application to blind detection in flat fading channels," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 02)*, Orlando, FL, USA, May 2002, vol. 2, pp. 1617–1620.
- [9] E. Arnaud and E. Memin, "Optimal importance sampling for tracking in image sequences : application to point tracking," in *Proceedings of the European Conference on Computer Vision (ECCV'04)*, Prague, May 2004.