FIXED-LAG SEQUENTIAL MONTE CARLO

Arnaud Doucet[1] *and Stéphane Sénécal*[2]

[1]Department of Engineering, University of Cambridge, ad2@eng.cam.ac.uk
[2]The Institute of Statistical Mathematics, Tokyo, steph@ism.ac.jp

ABSTRACT

Sequential Monte Carlo methods, aka particle methods, are an efficient class of simulation techniques to approximate sequences of complex probability distributions. These probability distributions are approximated by a large number of random samples called particles which are propagated over time using a combination of importance sampling and resampling steps. The efficiency of these algorithms is highly dependent on the importance distribution used. Even if the optimal importance distribution is chosen, the algorithm can be inefficient. Indeed, current standard sampling strategies extend the paths of particles over one time step and weight them consistently but do not modify the locations of the past of the paths. Consequently, if the discrepancy between two successive probability distributions is high, then this strategy can be highly inefficient. In this paper, we propose an extended importance sampling technique that allows us to modify the past of the paths and weight them consistently without having to perform any local Monte Carlo integration. This approach reduces the depletion of samples. An application to an optimal filtering problem for a toy nonlinear state space model illustrates this methodology.

1. INTRODUCTION

For the past decade, sequential Monte Carlo (SMC) methods have been considered for many applications in engineering and statistics [4]. In particular, they are now used extensively to solve optimal filtering problems for nonlinear non Gaussian state-space models arising in telecommunications [9, 10] and robotics for instance [4]. These methods approximate the sequence of probability distributions of interest using a large set of random samples, named particles, using simple sampling and resampling mechanisms. No linearity or gaussianity assumption is required. Asymptotically, i.e. as the number of particles goes to infinity, convergence of the particle approximations towards the sequence of probability distributions can be ensured [3]. However, for practical implementations, a finite and sometimes quite restricted number of particles has to be considered. In these cases, it is crucial to design an efficient importance sampling distribution. The optimal importance distribution for an adequate criterion has been established [5] and various approximations to this distribution have been proposed in the literature [5, 8]. Alternative look-ahead techniques have been proposed to improve the sampling schemes [8, 2, 11]: they attempt to boost the number of particles which will become significant in the next simulation steps. Nevertheless, this class of techniques does not introduce diversity in the set of particles and just re-weight them in a consistent way. We propose here a special importance sampling technique which allows us to re-impute the path of a particle on a fixed-lag without having to performed explicitly a Monte Carlo integration. This method is a natural extension of standard schemes and applies straightforwardly everywhere SMC are currently used.

The paper is organized as follows. In section §2, standard SMC methods are reviewed and we outline their limits. The fixed-lag SMC method is presented in section §3. Some numerical experiments to illustrate this approach are provided in section §4. Finally, we conclude in section §5.

This work was supported by The Japanese Ministry of Education and the Japan Society for the Promotion of Science. The first author was partially supported by EPSRC.

2. SEQUENTIAL MONTE CARLO

Let $\{\pi_n(x_{1:n})\}_{n\geq 1}$ denote a sequence of probability density functions indexed by the discrete-time index n, and $z_{i:j}=(z_i,\ldots,z_j)$ and $Z_{i:j}=(Z_i,\ldots,Z_j)$ for any deterministic z_n/random Z_n sequences. Without loss of generality, we assume that π_n is defined on E^n. We are interested in obtaining N ($N\gg 1$) weighted random samples $\left\{W_n^{(i)},X_{1:n}^{(i)}\right\}$ ($W_n^{(i)}\geq 0$, $\sum_{i=1}^N W_n^{(i)}=1$) such that for any test function $\varphi_n : E^n\rightarrow\mathbb{R}$

$$\sum_{i=1}^N W_n^{(i)}\varphi_n\left(X_{1:n}^{(i)}\right)\overset{N\rightarrow\infty}{\rightarrow}\int\varphi_n(x_{1:n})\,\pi_n(x_{1:n})dx_{1:n}.$$

Note that a particle $X_{1:n}^{(i)}$ represents a path from time 1 to n. A standard algorithm satisfying this requirement is the Sequential Importance Sampling Resampling (SISR) scheme described in [4]. Given $\left\{W_{n-1}^{(i)},X_{1:n-1}^{(i)}\right\}$ approximating π_{n-1} at time $n-1$, the SISR performs as follows at time n:

1. Sample $\left\{X_n^{(i)}\right\}$ using a proposal distribution $q_n(\cdot|\cdot)$

$$X_n^{(i)}\sim q_n(\cdot|X_{1:n-1}^{(i)}) \tag{1}$$

2. Update and normalize the weights

$$W_n^{(i)}\propto W_{n-1}^{(i)}\underbrace{\frac{\pi_n(X_{1:n}^{(i)})}{\pi_{n-1}(X_{1:n-1}^{(i)})q_n(X_n^{(i)}|X_{1:n-1}^{(i)})}}_{\text{incremental weight}}. \tag{2}$$

3. If the degeneracy of weights is high, resample $\left\{X_{1:n}^{(i)}\right\}$ according to $\left\{W_n^{(i)}\right\}$ to obtain N unweighted particles (i.e. weights of resampled particles $W_n^{(i)}\leftarrow N^{-1}$).

This last rejuvenation step is required to counteract the degeneracy problem of the set of samples: variance of weights naturally increase with time, so that after a small number of iterations, all the particles except one might be assigned a non-zero weight. On the contrary to steps 1 and 2, which can be performed in parallel, resampling step 3 makes the particles interacting, and is thus generally the most computational expensive part for the sampling scheme. The complexity of SISR algorithms is proportional to the number of particles N.

It is easy to check that the proposal distribution minimizing at time n the variance of the incremental weight conditional upon $\left\{X_{1:n-1}^{(i)}\right\}$ is given by $q_n^{\text{opt}}(x_n|x_{1:n-1})=\pi_n(x_n|x_{1:n-1})$. In this case the associated incremental weight is given by $\pi_n(x_{1:n-1})/\pi_{n-1}(x_{1:n-1})$. However, in many interesting cases, it is typically difficult to sample from $\pi_n(x_n|x_{1:n-1})$ and impossible to compute in closed-form $w_n(x_{1:n-1})=\pi_n(x_{1:n-1})/\pi_{n-1}(x_{1:n-1})$[1]. Several approaches have

[1]For most models, $w_n(x_{1:n-1})$ actually only depends on x_{n-1}.

been proposed to approximate $\pi_n(x_n|x_{1:n-1})$ and $w_n(x_{1:n-1})$ in the optimal filtering context [8, 5].
We would like to emphasize at this point that, even if the optimal importance distribution can be used, this does not guarantee that the algorithm is efficient. Indeed if the discrepancy between $\pi_n(x_{1:n-1})$ and $\pi_{n-1}(x_{1:n-1})$ is high, then the variance of $w_n(x_{1:n-1})$ will be high and the algorithm will suffer from a severe depletion of samples. An obvious way to improve the algorithm would consist at time n of not only imputing $\left\{X_n^{(i)}\right\}$ but also re-imputing the variables $\left\{X_{n-L+1:n-1}^{(i)}\right\}$ in light of π_n. This is the approach developed in this paper.
Remark. In the optimal filtering framework, an hidden Markov process $\{X_n\}_{n\geq 1}$ of initial pdf $X_1 \sim \mu$ and transition density $X_n|X_{n-1} \sim f(\bullet|X_{n-1})$ is considered and the observations $\{Y_n\}_{n\geq 1}$ are conditionally independent of marginal density $Y_n|X_n \sim g(\bullet|X_n)$. In this case, $\pi_n(x_{1:n})$ is given by the joint posterior density of the states $X_{1:n}$ given a realization of the observations $Y_{1:n} = y_{1:n}$

$$\pi_n(x_{1:n}) = p(x_{1:n}|y_{1:n}) \propto \mu(x_1)\, g(y_1|x_1) \prod_{k=2}^{n} f(x_k|x_{k-1})\, g(y_k|x_k).$$

For this case, it is trivial to establish that $\pi_n(x_n|x_{1:n-1}) = p(x_n|y_n, x_{n-1})$ and $w_n(x_{1:n-1}) = p(y_n|x_{n-1})$.

3. FIXED-LAG SEQUENTIAL MONTE CARLO

Assume at time $n-1$ that a set of weighted particles $\left\{W_{n-1}^{(i)}, X_{1:n-1}^{(i)}\right\}$ approximates π_{n-1}. In the standard approaches, the current paths $\left\{X_{1:n-1}^{(i)}\right\}$ are extended by sampling $\left\{X_n^{(i)}\right\}$ according to $q_n(\cdot|\cdot)$ and reweighted. In the fixed-lag framework, we propose to sample not only the variables at time n but also to modify the previous values from time $n-L+1$ to $n-1$ where $L>1$ is a fixed integer. Consequently, the sampling step 1 of the SISR algorithm is modified and consists now of sampling

$$X_{n-L+1:n}^{\prime(i)} \sim q_n(\cdot|X_{1:n-1}^{(i)}).$$

The problem with this approach is that the marginal density of the new paths $\left\{X_{1:n-L}^{(i)}, X_{n-L+1:n}^{\prime(i)}\right\}$ is given by

$$q_n\left(x_{1:n-L}, x'_{n-L+1:n}\right) = \int \pi_{n-1}(x_{1:n-1}) q_n(x'_{n-L+1:n}|x_{1:n-1}) dx_{n-L+1:n-1} \quad (3)$$

and typically does not admit a closed-form expression. An exception consists of the finite state-space case where the integral becomes a finite sum. However, in this case, the computational complexity typically increases exponentially with L; this method has been proposed by [7]. To avoid having to compute (3), we consider the density of the paths $\left\{X_{1:n-1}^{(i)}, X_{n-L+1:n}^{\prime(i)}\right\}$ given by

$$\pi_{n-1}(x_{1:n-1}) q_n(x'_{n-L+1:n}|x_{1:n-1}). \quad (4)$$

We now propose to perform importance sampling on this joint space, in order to avoid to integrate explicitly the set of variables $x_{n-L+1:n-1}$. For this task, it is necessary to define an appropriate target density on the same space. Clearly if one sets the target density as

$$\pi_n(x_{1:n-L}, x'_{n-L+1:n}) \lambda_n(x_{n-L+1:n-1}|x_{1:n-1}, x'_{n-L+1:n})$$

where $\lambda_n(\cdot|\cdot)$ is any arbitrary conditional density on E^{L-1} then the incremental weight defined by

$$\frac{\pi_n(x_{1:n-L}, x'_{n-L+1:n}) \lambda_n(x_{n-L+1:n-1}|x_{1:n-1}, x'_{n-L+1:n})}{\pi_{n-1}(x_{1:n-1}) q_n\left(x'_{n-L+1:n}|x_{1:n-1}\right)} \quad (5)$$

leads to a consistent Monte Carlo scheme. The *fixed-lag SISR* algorithm proceeds now as follows

1. Sample $\left\{X_{n-L+1:n}^{\prime(i)}\right\}$ using a proposal distribution $q_n(\cdot|\cdot)$

$$X_{n-L+1:n}^{\prime(i)} \sim q_n(\cdot|X_{1:n-1}^{(i)}).$$

2. Update and normalize the weights with incremental weight

$$\frac{\pi_n(X_{1:n-L}^{(i)}, X_{n-L+1:n}^{\prime(i)}) \lambda_n(X_{n-L+1:n-1}^{(i)}|X_{1:n-1}^{(i)}, X_{n-L+1:n}^{\prime(i)})}{\pi_{n-1}(X_{1:n-1}^{(i)}) q_n\left(X_{n-L+1:n}^{\prime(i)}|X_{1:n-1}^{(i)}\right)}.$$

3. If the degeneracy of weights is high, resample $\left\{X_{1:n-L}^{(i)}, X_{n-L+1:n}^{\prime(i)}\right\}$ according to $\left\{W_n^{(i)}\right\}$ to obtain N unweighted particles $\left\{X_{1:n}^{(i)}\right\}$ (i.e. weights of resampled particles $W_n^{(i)} \leftarrow N^{-1}$).

It is also possible to develop a fixed-lag version of the auxiliary particle method [6]. The choice of the densities $q_n(\cdot|\cdot)$ and $\lambda_n(\cdot|\cdot)$ is crucial for the method to be efficient. The details are omitted here, see [6], but it can be shown that the *optimal fixed-lag densities* $q_n(\cdot|\cdot)$ and $\lambda_n(\cdot|\cdot)$ minimizing the variance of incremental importance weights (5) conditional upon $\left\{X_{1:n-1}^{(i)}\right\}$ are given by $q_n^{\text{opt}}\left(x'_{n-L+1:n}\middle|x_{1:n-L}\right) = \pi_n\left(x'_{n-L+1:n}\middle|x_{1:n-L}\right)$ and

$$\lambda_n^{\text{opt}}(x_{n-L+1:n-1}|x_{1:n-1}, x'_{n-L+1:n}) = \pi_{n-1}(x_{n-L:+1:n-1}|x_{1:n-L})$$

which yield the associated optimal importance weight $w_n(x_{1:n-L}) = \pi_n(x_{1:n-L})/\pi_{n-1}(x_{1:n-L})$. In most cases, one cannot sample from $\pi_n(x_{n-L+1:n}|x_{1:n-L})$ and it is impossible to compute λ_n^{opt} and $w_n(x_{1:n-L})$ in closed-form. However, the framework we propose allows us to use approximations $\widehat{\pi}_n(x_{n-L+1:n}|x_{1:n-L})$ and $\widehat{\pi}_{n-1}(x_{n-L:+1:n-1}|x_{1:n-L})$ and consequently approximations $\widehat{\pi}_n(x_{1:n-L})$ and $\widehat{\pi}_{n-1}(x_{1:n-L})$ of $\pi_n(x_{1:n-L})$ and $\pi_{n-1}(x_{1:n-L})$.
Remark. The optimal filtering framework yields $\pi_n(x_{1:n}) = p(x_{1:n}|y_{1:n})$, $\pi_n(x_{n-L+1:n}|x_{1:n-L}) = p(x_{n-L+1:n}|y_{n-L+1:n}, x_{1:n-L})$ and $w_n(x_{1:n-L}) = p(y_n|y_{n-L+1:n}, x_{n-L})$. Clearly the incremental weight associated to the optimal or approximately optimal fixed-lag importance distribution is expected to have a lower variance than when $L=1$.
Remark. We do not claim that this method will outperform standard SMC. It entirely depends on the context and on the ability of the user to design good approximations of the optimal importance sampling distribution. In the next section, we consider a nonlinear state space model. Standard ($L=1$) and fixed-lag ($L>1$) sampling schemes are presented. To build such an approximation, we use the Extended Kalman filter and the forward filtering-backward sampling decomposition; see [1] for instance.

4. NUMERICAL EXPERIMENTS

To illustrate the fixed-lag approach we are proposing, the following state-space model is considered

$$X_n = \alpha(X_{n-1} + \beta X_{n-1}^3) + U_n \quad (6)$$
$$Y_n = X_n + V_n \quad (7)$$

where $U_n \sim \mathcal{N}(0, \sigma_u^2)$ and $V_n \sim \mathcal{N}(0, \sigma_v^2)$ stand respectively for the process and measurement noises. Parameters are set to α=0.9, β=0.2, σ_u=0.1 and σ_v=0.05. Sequences (6) and (7) of 200 samples are considered. We are interested in estimating the mean $E\{X_n|y_{1:n}\}$ of the marginal posterior distribution $p(x_n|y_{1:n})$. The estimates computed from Monte Carlo approximations are the empirical averages $\widehat{X}_n = \sum_{i=1}^N W_n^{(i)} X_n^{(i)}$ for different sampling schemes:

- the bootstrap filter [4]

- SISR with predictive distribution approximated by Kalman filters, denoted as SISR-KF and SISR-EKF
- SISR for sampling a block of variables with L-step ahead predictive distributions approximated by Kalman filters, denoted as BSISR-KF and BSISR-EKF

Simulations were run for 100 realizations. Firstly, we considered sample sets of $N = 100$ particles. To assess the approximation of the target distribution, the effective sample size (ESS), approximated as $(\sum_{i=1}^{N}[W_n^{(i)}]^2)^{-1}$ [4], is computed and used to perform resampling step 3 each time ESS$\leq \frac{N}{2}$. The averaged percentage of resampling steps (RS) for simulation runs is also recorded as well as the computing time (CPU) for processing the data. The block sampling schemes improve the estimation and the approximation of the target distribution as seen on table 1 for the model considered (6) (7).

algorithm	MSE	ESS	RS	CPU
Bootstrap	0.0021	36.8	70.3 %	0.68
SISR-KF	0.0019	64.7	19.3%	0.44
SISR-EKF	0.0019	65.8	19.2%	0.48
BSISR-KF	0.0018	72.3	0.9%	0.21
BSISR-EKF	0.0018	73.5	0.8%	0.24

Table 1: Simulation results for state space model (6) (7), $N = 100$ particles, 100 runs of particle filters for a single and for a block of $L = 2$ variables.

The Kalman and extended Kalman filter proposals give similar results for the sampling of single and for a block of variables. Also, these filters implemented to compute the posterior mean give a slightly higher mean squared error, equal to 0.0034 in this case. As the distribution of interest $p(x_n|y_{1:n})$ is not Gaussian, Monte Carlo approximations usually give better estimates than the means computed from Kalman filters.

For the model considered here, the choice of the sampling scheme is crucial to propagate efficiently the particles in the sampling space. This is demonstrated by different values for averaged ESS in table 1. To focus more on the degeneracy involved by the sampling schemes, instantaneous ESS indexed by time is depicted in figure 1. Higher values and low decreasing rates for the ESS show that the degeneracy of the sampling scheme can be efficiently dealt within a block sampling approach for this model. This results in less frequent resampling steps, as shown by the RS column in table 1. Consequently, the computational time required by this algorithm is reduced. This is illustrated with the CPU measured quantities for different sampling schemes, given in table 1. As recalled in section §2, the main contribution of the computing time for SISR algorithm comes from the resampling step 3, and is globally proportional to the number of particles N. This quantity is displayed as a function of N and for the different sampling schemes on figure 2. For model (6) (7), the computational time is significantly reduced when considering a block approach in comparison with sampling schemes with one variable, for a given number of particles.

Performance of block sampling schemes is now considered for different lags L. The mean squared error is stable for different configurations and averaged estimates of effective sample size are depicted in table 2 for different number of particles N. Proposals of Kalman and extended Kalman filters lead to very similar results, and thus are not distinguished in table 2, but considering a block size L from 3 to 5 samples gives the best propagation of particles in the sampling space. Sampling blocks of larger sizes is not necessarily efficient; the computational complexity increases and the approximation of the optimal importance distribution deteriorates.

The proportion of resampling steps (RS), also recorded for the various configurations, does not vary significantly with parameter N. It is more sensitive to the block size L and thus has an incidence on the computational time. Figure 3 depicts the computing time as a function of the number of particles for block sampling schemes with different lags. As expected, considering blocks of larger lag

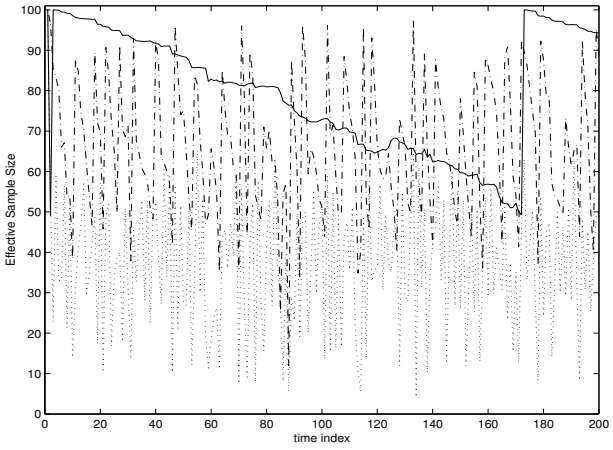

Figure 1: Approximated Effective Sample Size vs. time index for a realization of the Bootstrap filter (dotted), the SISR with Kalman filter proposal for a single variable (dashdotted) and for a block of L=2 variables (straight), $N = 100$ particles.

block size L	N=100	N=500	N=1000	RS
2	74	370	715	0.9%
3	96	493	985	0.9%
4	99	496	989	1%
5	98	494	988	1%
10	97	486	972	2.5%

Table 2: Approximated Effective Sample Size for state space model (6) (7), averaged over 100 runs of particle filters for blocks of L variables, considering N particles.

increases the computational time. A trade-off has thus to be made by the user with respect to estimation task and the quality of the Monte Carlo approximation, tackled by measuring mean squared error and effective sample size, and the computational power available: the number of particles, the choice of the size for blocks. It should finally be recalled that processing variables by block for the model (6) (7) considered here makes it possible to propagate the particles more efficiently than by considering a single variable, and thus it minimizes computational time for the simulation.

5. CONCLUSION

We have proposed an original methodology to impute blocks of variables within a Sequential Monte Carlo framework. Previous techniques proposed in the literature were relying on look-ahead approaches requiring expensive calculations. The method presented here is a cheaper and natural extension of standard SMC algorithms and can be used wherever these algorithms apply.

However, as in the standard case, one can only expect the algorithm developed to be efficient if it is possible to design some sensible approximation of the (fixed-lag) optimal importance distribution. Given the higher computational cost of fixed-lag importance sampling schemes compared to standard ones, it is difficult to assess beforehand whether it is beneficial for a specific application. Nevertheless, the example presented in the previous section shows that it can significantly reduce the number of resampling steps and overall being more computationally efficient. Generally speaking, our guidelines are that we will observe significant gains when the discrepancy between two successive target distributions is high. In the optimal filtering framework, this situation occurs when one receives for example a very informative observation; this is demonstrated by the numerical simulation in the previous section. This suggests that the fixed-lag sampling approach could also only be

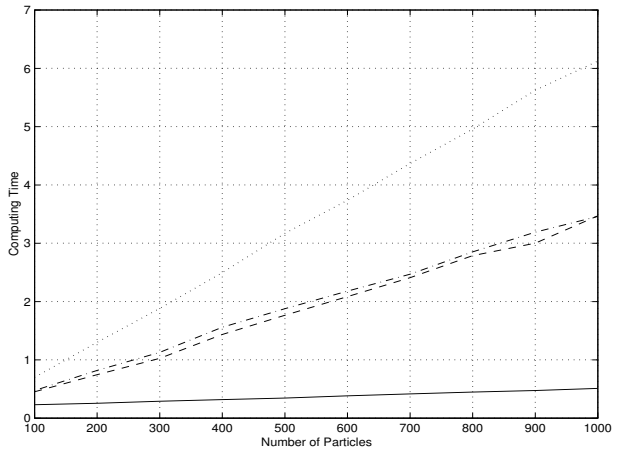

Figure 2: Computational time vs. number of particles N for the Bootstrap filter (dotted), the SISR with Kalman filter proposal for a single variable (KF: dashed, EKF: dashdotted) and for a block of L=2 variables (straight), 100 realizations.

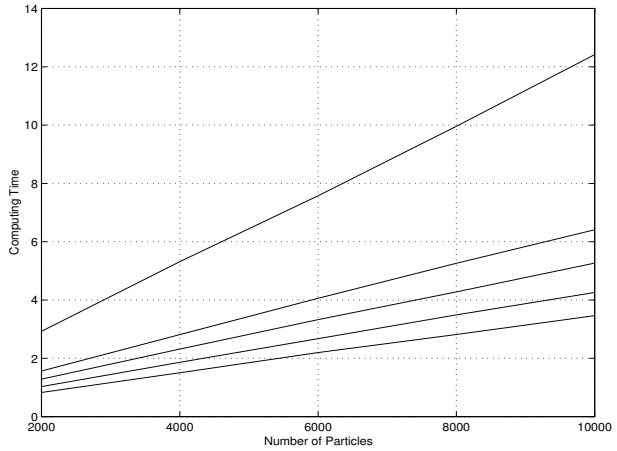

Figure 3: Computational time vs. number of particles N for the block sampling scheme with different lags from L=2 (bottom), 3, 4, 5, 10 (top), 100 realizations.

used in cases where one observes a significant drop of the effective sample size using standard techniques. Applications of this methodology for various navigation and tracking applications are currently under study.

REFERENCES

[1] C. K. Carter and R. Kohn. On the Gibbs sampling for state space models. *Biometrika*, 81:541–553, 1994.

[2] R. Chen, X. Wang, and J. S. Liu. Adaptive joint detection and decoding in flat-fading channels via mixture Kalman filtering. *IEEE Trans. Info. Theory*, 46:2079–2094, 2000.

[3] P. Del Moral. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Probability and Applications. Springer, 2004.

[4] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo methods in practice*. Statistics for engineering and information science. Springer, 2001.

[5] A. Doucet, S. Godsill, and C. Andrieu. On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. *Statistics and Computing*, 10:197–208, 2000.

[6] A. Doucet and S. Sénécal. Sampling strategies for sequential Monte Carlo methods. Technical report, Signal Processing Group, Department of Engineering, University of Cambridge, 2003.

[7] H. Meirovitch. Scanning method as an unbiased simulation technique and its application to the study of self-attracting random walks. *Phys. Rev. A*, 32:3699–3708, 1985.

[8] M. K. Pitt and N. Shephard. Filtering via simulation: auxiliary particle filter. *Journal of the American Statistical Association*, 94:590–599, 1999.

[9] E. Punskaya, A. Doucet, and W. J. Fitzgerald. On the use and misuse of particle filtering in digital communications. In *Proc. EUSIPCO*, 2002.

[10] S. Sénécal, P.-O. Amblard, and L. Cavazzana. Sequential monte carlo method for blind equalization of a nonlinear satellite communication channel. In *Proc. ICASSP'03*, volume 6, pages 697–700, 2003.

[11] X. Wang, R. Chen, and D. Guo. Delayed-pilot sampling for mixture Kalman filter with application in fading channels. *IEEE Trans. Sig. Proc.*, 50:241–253, 2002.