A NEW UNSCENTED PARTICLE FILTER

Qi Cheng and Pascal Bondon

CNRS UMR 8506, Université Paris-Sud, 3 rue Joliot-Curie, 91192 Gif-sur-Yvette, France.
{cheng, bondon}@ lss.supelec.fr

ABSTRACT

We present a new unscented particle filter for dynamic systems that outperforms the general particle filter and the unscented particle filter when the variance of the observation noise is small. Our algorithm uses a bank of unscented Kalman filters to refine the prediction in particle filter. The key difference with the traditional unscented particle filter is the introduction of an auxiliary model and a bank of unscented Kalman filter with this auxiliary model to generate the importance distribution in the particle filter. This structure makes efficient use of the latest observation information. Our new algorithm use fewer particles than the general particle filters and its performance outperforms them.

Index Terms— Monte Carlo methods, nonlinear filters, Kalman filtering

1. INTRODUCTION

Consider a dynamic nonlinear discrete time system described by a state-space model

$$x_t = f(x_{t-1}) + u_t, \tag{1}$$

$$y_t = h(x_t) + v_t, \tag{2}$$

where x_t is the hidden state, y_t is the observation, and u_t, v_t are the state and observation noises. Both noises are independent and identically distributed sequences and are mutually independent. When we write (1), we always assume implicitly that u_t is independent of $\{x_{t-k}, k \geq 1\}$. This condition is natural when the process (x_t) is generated from the model in the increasing time order. Then, x_t is an homogeneous Markov chain, i.e., the conditional probability density of x_t given the past states $x_{0:t-1} = (x_0, \ldots, x_{t-1})$ depends only on x_{t-1} through the transition density $p(x_t|x_{t-1})$, and the conditional probability density of y_t given the states $x_{0:t}$ and the past observations $y_{1:t-1}$ depends only on x_t through the conditional likelihood $p(y_t|x_t)$. We further assume that the initial state x_0 is distributed according to a density function $p(x_0)$.

In a Bayesian framework, the posterior density of the state given the past observations $p(x_t|y_{1:t})$, constitutes the solution to the inference problem and allows to calculate for instance the conditional mean $\mathrm{E}(x_t|y_{1:t})$ which is the best mean squared estimate of the state. A recursive update of the posterior density as new observations arrive is given by the recursive Bayesian filter defined by

$$p(x_{t+1}|y_{1:t}) = \int p(x_{t+1}|x_t)p(x_t|y_{1:t})dx_t,$$

$$p(x_{t+1}|y_{1:t+1}) = \frac{p(y_{t+1}|x_{t+1})p(x_{t+1}|y_{1:t})}{p(y_{t+1}|y_{1:t})},$$

where the conditional density $p(y_{t+1}|y_{1:t})$ can be calculated by $p(y_{t+1}|y_{1:t}) = \int p(y_{t+1}|x_{t+1})p(x_{t+1}|y_{1:t})dx_{t+1}$.

The difficulty to implement the recursive Bayesian filter is that the integrals are intractable, except for a linear Gaussian system in which case the closed-form solution of the integral equations is the well known Kalman filter. Particle filters (PF) use simulation-based methods to calculate the multidimensional integrals. The key idea of PF is to implement the recursive Bayesian filter by Monte Carlo methods. However, the performance of PF often varies largely and many efforts have been done to improve it, see [1], [2], [3]. In this paper, we present a new unscented particle filter (UPF) that outperforms the standard PF and the UPF when the variance of the observation noise is small.

The remainder of this paper is organized as follows. We briefly introduce the principle of the PF and the unscented Kalman filter (UKF) before the presentation of our algorithm. Then, the effectiveness of our method is illustrated on a typical model used in [3]. Finally, some conclusions are given.

2. BACKGROUND

The principle of PF is to implement the recursive Bayesian filter by Monte Carlo simulations, [4]. The posterior density $p(x_t|y_{1:t})$ is represented by a set of N random samples x_t^i (particles) drawn from $p(x_t|y_{1:t})$ with associated normalized positive weights ω_t^i ($\sum_i \omega_t^i = 1$). The posterior density is approximated by the discrete distribution $\sum_{i=1}^N \omega_t^i \delta(x_t - x_t^i)$, and the expectation of any integrable function $g(\cdot)$ of hidden states is approximated by the sum,

$$E[g(x_t)] = \int g(x_t)p(x_t|y_{1:t})dx_t \simeq \sum_{i=1}^N \omega_t^i g(x_t^i).$$

The density $p(x_t|y_{1:t})$ is a marginal of the full posterior density $p(x_{0:t}|y_{1:t})$. In general, it is difficult to sample directly from the full posterior density. To overcome this difficulty, we sample from an easy sampling proposal importance distribution $q(x_{0:t}|y_{1:t})$. Defining the weights as

$$\omega_t^i = \frac{p(x_{0:t}^i|y_{1:t})}{q(x_{0:t}^i|y_{1:t})},$$

ω_t^i is updated by

$$\omega_t^i = \omega_{t-1}^i \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, y_t)}. \tag{3}$$

We can implement recursively a basic sequential importance sampling particle (SIS) filter in the following steps [4] :

1. Sample the particles $x_t^i \sim q(x_t|x_{t-1}^i, y_t)$;

2. Update the weights according to (3).

An important problem of PF is the degeneracy problem. After a few iterations, only few particles have non negligible weights and the estimation may become unreliable. Sampling importance resampling PF has been developed by [5] to overcome this drawback. The objective of resampling is to eliminate samples with low importance weights and multiply samples with high importance weights. Several methods of resampling have been developed, such as multinomial resampling, residual resampling and systematic resampling. In this paper, we use residual sampling, see [6] and [7].

2.1. The Importance Distribution

The choice of proposal importance distribution is one of the critical issues in PF. The performance of PF heavily depends on the proposal importance function. The optimal proposal importance distribution is $q(x_t|x_{0:t-1}, y_{1:t}) = p(x_t|x_{t-1}, y_k)$ and fully exploits the information in both x_{t-1} and y_t [7]. In practice, it is impossible to sample from this distribution due to its complication. The second choice of proposal function is the transmission prior function $q(x_t|x_{0:t-1}, y_{1:t}) = p(x_t|x_{t-1})$ for its easiness to sample. This is the most popular choice. But since this function does not use the latest information y_t, the performance depends heavily on the variance of observation noise. When the observation noise variance is small, the performance is poor, see [8], [9]. The third choice is to use the method of local linearization to generate the proposal importance distribution. Then, Rao Blackwellised PF was developed by [6] and uses extended Kalman filter (EKF) to generate the proposal distribution. UPF was introduced by [3] and uses UKF to generate the proposal distribution. In [10], EKF, UKF and Gaussian-Hermite filter were used to generate the proposal distribution. Since all these filters use the latest information y_t, the choice of the method of local linearization is better than the transmission prior function.

3. THE UNSCENTED KALMAN FILTER

The UKF was first developed by [11] in order to overcome the drawbacks of EKF. Since EKF uses the method of linearization to approach the true nonlinear systems, EKF performs badly when the system is severely nonlinear. However, UKF uses several so called sigma points to calculate the mean and covariance of random variables, see [12]. These sigma points propagate through the true nonlinear system. The posterior estimation is calculated by the average of the sigma points. UKF is essentially a kind of Quasi-Monte Carlo method, see [13]. UKF uses the sigma points to compute the covariances needed in Kalman filtering. The ability of processing nonlinearity is the advantage of UKF. The UKF is valid only when the posterior distribution can be closely approximated by a Gaussian distribution [8]. The drawbacks of UKF are

1. It preserves the linear update structure of the Kalman filter which is optimal only in linear Gaussian systems.

2. It uses only second order moments which is valid only for Gaussian distributions.

3. The number of sigma points is small and may not represent adequately complicated distributions.

UKF and UPF work well when the variance of the observation noise is small. This may be due to the fact that the distribution of a random variable with a small variance (10^{-4}) can be approximated reasonably by a Gaussian distribution.

4. A NEW UNSCENTED PARTICLE FILTER

In the following, we refer to (1) and (2) as the **main model**. We introduce now an auxiliary model and we use a bank of UKF with this model to generate the importance distribution.

4.1. Auxiliary model

The **auxiliary model** is designed to represent a constant signal r_t observed through z_t and writes

$$r_t = r_{t-1} + m_t, \tag{4}$$

$$z_t = h(r_t) + n_t. \tag{5}$$

The noise m_t is Gaussian with a small variance, typically 10^{-5}. The variances of the observation noises n_t and v_t in (2) are supposed to be equal. If $h(\cdot)$ is linear and n_t is a Gaussian noise, the Kalman filter can be used to track the state r_t. In most applications, $h(\cdot)$ is nonlinear. Then, UKF can be used to track r_t and will work well if the variance of n_t is small.

4.2. The algorithm

We develop a modified UPF (MUPF) which works well when the variance of the observation noise is small. In this case, the posterior distribution of the hidden state is highly peaked and can be well approximated by a Gaussian distribution with a small variance. In MUPF, at step t, we use the UKF in the main model to generate the importance distribution from x_{t-1}^1 to sample the first particle x_t^1 in the same way as in the UPF. Then, we use x_t^1 and the auxiliary model using UKF to generate the importance distribution from which the second particle x_t^2 is sampled. Iterating this process, we obtain recursively $x_t^3, \ldots, x_t^N$, where N is the total number of particles used in the PF, see Figure 1. Let $N(\bar{x}_t^1, P_t^1)$ be the Gaussian importance distribution of x_t^1 obtained by using UKF with the main model. Because of the drawbacks of UKF, $N(\bar{x}_t^1, P_t^1)$ may be a poor approximation of the true posterior distribution of x_t. MUPF can provide a better approximation. Indeed, when the variance of observation noise is small, the observation y_t is highly informative and in a sense contains more information than the prior transmission function. For this reason, only the first particle x_t^1 is sampled directly from the prior $p(x_t|x_{t-1})$, and the others particles are sampled from importance distributions obtained from x_t^1 and the auxiliary model which is designed to fully exploit the information of y_t. In the auxiliary model, r_t is an almost constant signal. We set $\bar{r}_0 = x_t^1$, $P_{r_0} = P_t^1$ and $z_1 = y_t$. Then we calculate the posterior distribution $N(\bar{r}_1, P_{r_1})$ of r_1 using UKF with the auxiliary model, we set $N(\bar{x}_t^2, P_t^2) = N(\bar{r}_1, P_{r_1})$ and we sample a particle x_t^2 from the distribution $N(\bar{x}_t^2, P_t^2)$. We iterate this generating process by setting $z_2 = \cdots = z_{N-1} = y_t$. Since r_t is an almost constant signal and the variance of n_t is small, the importance distributions $N(\bar{x}_t^2, P_t^2), \ldots, N(\bar{x}_t^N, P_t^N)$ are

good approximations of the posterior distribution $p(x_t|y_{1:t})$.

MUPF

1. Initialization, $t = 0$:
   - For $i = 1, \ldots, N$,draw particle $x_0^i \sim p(x_0)$ and set $t = 1$.
2. Importance sampling step:
   - For $i = 1$, use UKF with the **main model** to generate the importance distribution $N(\bar{x}_t^1, P_t^1)$ from particle x_{t-1}^1. Sample $x_t^1 \sim N(\bar{x}_t^1, P_t^1)$ (Same procedure as UPF).
   - For $i = 2, 3, \ldots N$, use UKF with the **auxiliary model** to generate the importance distribution $N(\bar{x}_t^i, P_t^i)$ from particle x_t^{i-1}. Sample $x_t^i \sim N(\bar{x}_t^i, P_t^i)$.
3. Importance weight step:
   - For $i = 1, \ldots, N$, evaluate the importance weights $\tilde{\omega}_t^i = \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{N(\bar{x}_t^i, P_t^i)}$.
   - For $i = 1, \ldots, N$, normalise the importance weight $\omega_t^i = \frac{\tilde{\omega}_t^i}{\sum_{j=1}^N \tilde{\omega}_t^i}$.
4. Resampling step:
   - Resample N particles $\tilde{x}_{1:t}^i$ from the $x_{1:t}^i$ according to the normalized importance weights.
   - set $\omega_t^i = \frac{1}{N}$

Figure 2 illustrates the advantage of MUPF with the nonlinear non Gaussian model considered in Section 5. We take $t = 21$, the true value of x_t is 5.0461. We generate 20 particles by PF, UPF and MUPF. It is clear that MUPF performs better than PF and UPF, since except x_t^1 and x_t^2, all the others particles x_t^i provide a good prediction of the true state x_t. In practice, only 5 particles are needed in MUPF to obtain a very good performance.

5. NUMERICAL EXAMPLE

To illustrate the performance of MUPF, we consider a nonlinear non Gaussian model which was used in [3] and given by

$$x_t = 1 + \sin(w\pi(t-1)) + \frac{1}{2}x_{t-1} + u_t,$$
$$y_t = \begin{cases} \frac{1}{5}x_t^2 + v_t, & t \leq 30, \\ \frac{1}{2}x_t - 2 + v_t, & t > 30, \end{cases}$$

where $w = 4e - 2$, u_t follows a $\Gamma(3, 2)$ distribution, and v_t follows a $N(0, 10^{-5})$ distribution. We compare PF, UPF and MUPF to estimate the hidden states x_t for $t = 1, \ldots, 60$.

In Table 1, we present the corresponding root mean-squared error (RMSE), using 200, 50, 20, and 5 particles, respectively, in each algorithm. The experiment is repeated 100 times independently, and Mean and Variance denote the estimated mean

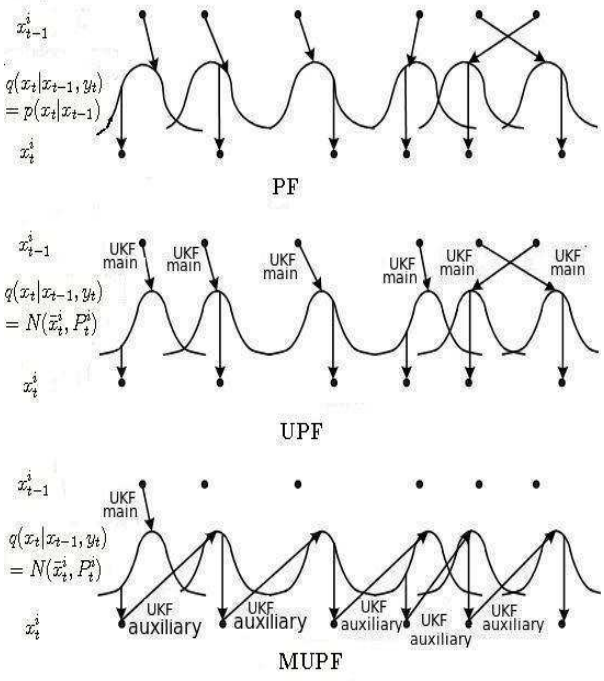

Fig. 1. Schema of the different sampling methods.

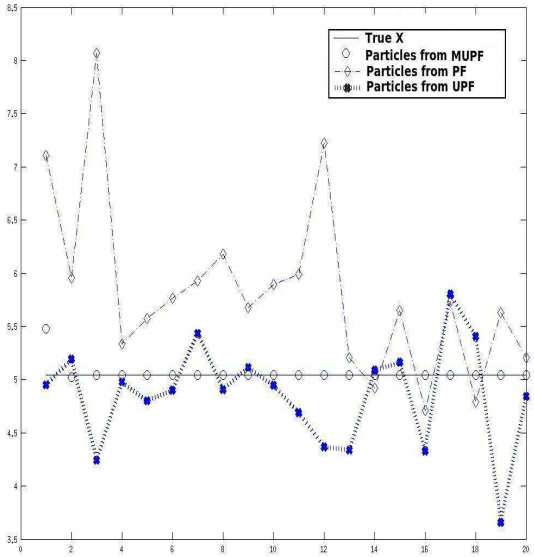

Fig. 2. Sampling results from PF, UPF and MUPF at a fixed time using 20 particles.

Algorithm	Mean	Variance	Time	N
PF	0.4390	0.0598	1.89	200
UPF	0.0749	0.0089	11.25	
MUPF	0.0048	3.43e-7	11.33	
PF	0.6836	0.0320	1.94	50
UPF	0.1794	0.0098	2.91	
MUPF	0.0049	3.77e-7	2.99	
PF	0.7852	0.0261	0.86	20
UPF	0.3664	0.0096	1.17	
MUPF	0.0050	3.48e-7	1.19	
PF	1.0622	0.0187	0.83	5
UPF	0.5831	0.0086	0.86	
MUPF	0.0109	0.0027	0.86	

Table 1. RMSE of PF, UPF and MUPF.

and variance of RMSE, respectively, Time is the computing time, and N is the number of particles used in each method.

In Figure 3, we show the result of the estimation of the state x_t obtained after resampling for $t = 1, \ldots, 60$, using 20 particles in each algorithm. When $t = 21$,$x_t = 5.0461$, and the values of the 20 particles before resampling were shown in Figure 2.

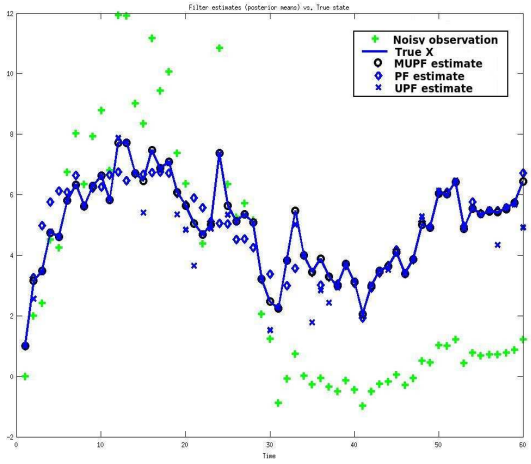

Fig. 3. The results of estimation using 20 particles.

The results show that MUPF needs a smaller number of particles than PF and UPF to get accurate estimations of the state. With 5 particles, MUPF outperforms PF and UPF using 200 particles. Finally, UPF and MUPF have the same complexity of implementation.

6. CONCLUSION

We have proposed a new UPF. The idea is to add a bank of UKF in the sampling step in the framework of general PF. The main problem in PF is the choice of the importance distribution that greatly influence its performance. In general, the particles generated from the transmission prior function do not match an highly peaked likelihood function. UPF can performs better by moving the particles to areas of high likelihood through one step prediction of UKF. However, due to the drawbacks of UKF, the one step prediction of UKF may be rough. By using a bank of UKF with an adequate auxiliary model, we have proved that the precision of the prediction can be improved. When the observation noise variance is small, the performance of our new algorithm outperforms the PF and UPF greatly with the same complexity as UPF.

7. REFERENCES

[1] M. K. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters," *J. Amer. Statist. Assoc.*, vol. 94, no. 446, pp. 590–599, 1999.

[2] R. Chen and J. Liu, "Mixture Kalman filters," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 62, no. 3, pp. 493–508, 2000.

[3] R. van der Merwe, N. de Freitas, A. Doucet, and E. Wan, "The unscented partical filter," in *Advances in Neural Information Processing Systems*, Nov 2000.

[4] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filter for on-line nonlinear/non-gaussian bayesian tracking.," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[5] 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, 1993.

[6] J. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Statist. Assoc.*, vol. 93, no. 443, pp. 1032–1044, 1998.

[7] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[8] O. Cappe, S. J. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential monte carlo," *IEEE Proceedings*, vol. 95, no. 5, pp. 899–924, 2007.

[9] C. Musso, N. Oudjane, and F. Le Gland, "Improving regularised particle filters," in *Sequential Monte Carlo methods in practice*, Stat. Eng. Inf. Sci., pp. 247–271. Springer, New York, 2001.

[10] D. Guo, Wang X., and Chen R., "New sequential monte carlo methods for nonlinear dynamic systems," *Statistics and Computing*, vol. 15, no. 2, pp. 135–147, 2005.

[11] S. Julier and J. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997.

[12] R. van der Merwe and E. Wan, "The square-root unscented kalman filter for state and parameter estimation.," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Salt Lake City, Utah, 2001, IEEE.

[13] D. Guo and X. Wang, "Quasi-monte carlo filtering in nonlinear dynamic systems.," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2087–2098, 2006.