

# Adaptive Particle Filter Approach to Approximate Particle Degeneracy

J. Joseph Ignatious , A. Uma Mageswari, S. Abraham Lincon.

**Abstract** — The main problem of particle filter in nonlinear state estimation is the particle degeneracy. It can be overcome by Resampling operation. But Resampling operation leads to the problem of sample impoverishment. Therefore an algorithm named Variance reduction technique is proposed to solve sample impoverishment and degeneracy problem. It reduces the variance of the particle weights by selecting an exponential fading factor and this factor can be chosen adaptively and iteratively in terms of the effective particle number. Many improved particle filter algorithms were proposed to solve the degeneracy problem which are seemed to be complex. In this paper an algorithm is presented to show that the idea of Variance reduction technique is feasible to propose a new adaptive filtering algorithm.

**Keywords**— Particle filter, Adaptive Particle filter , particle degeneracy, sample Impoverishment, Adaptive Algorithm.

---

◆

## 1. INTRODUCTION

Particle filter (PF) is a Monte Carlo estimation algorithm suitable for the state estimation of nonlinear and/or non-Gaussian systems by constructing the posterior probability density function via a number of weighted particles. It is widely used in target tracking, signal processing, image processing and so on.[1-2]. However, PF may confront the particle degeneracy. The resampling operation solves the degeneration of the particle set to some extent, but it leads to the problem of sample impoverishment. Therefore, how to deal with the degeneration of the particle set effectively is a crucial problem in PF design. For many years, much attention has been paid to solve the issue of particle degeneration and sample impoverishment. Some novel techniques were adopted to solve this problem, and many improved algorithms of PF were proposed which included regularized particle filter[3], auxiliary particle filters, auxiliary extended and auxiliary unscented Kalman particle filters [4], genetic particle filter (GPF)[5], risk sensitive particle filters (RSPF) [6], and so on.

To overcome the sample impoverishment after resampling step, the fission bootstrap PF (FBPF)[7] algorithm was proposed , in which the preprocessing included weights sorting, particle reproducing by fission and weights normalizing was inserted before the original resampling step as soon as the particle set degenerates severely. However, the problems of weights allocation and the number of fission particles were not solved well. Annealed particle filter proposed restrained the degeneration by introducing an annealing factor but the choice of this factor is somewhat complex.

In order to solve the degeneracy of the particle set, this paper proposes an adaptive PF (APF) algorithm via variance reduction technique. After setting a threshold of effective particle number and the reduction rate, the fading factor was chosen by the variance of importance weights recursively and adaptively. A numerical example was given to show that the proposed APF in this paper has better estimation accuracy compared with generic particle filter sampling importance of resampling (PF-SIR), generic particle filter (GPF)[8]. In this paper , section 2 deals with the particle filter and its estimation, section 3 deals with the adaptive particle filter and its algorithm, section 4 deals with the numerical example based on univariate non stationary growth model, section 5 deals about the simulation results by simulating the UNGM[9] model equation and last chapter deals with the conclusion.

Through this paper we use the following notations:

**K**: time step  
 **$W_k$** : process noise  
 **$V_k$** : observation noise  
 **$X_k$** : State vector

- 
- J. Joseph Ignatious, Research Scholar, Faculty of Engineering, Annamalai University, Chidambaram, TamilNadu, India.  
E-mail: [jjj\\_02@yahoo.co.in](mailto:jjj_02@yahoo.co.in)
  - A. Uma Mageswari, Research Scholar, Faculty of Engineering, Annamalai University, Chidambaram, TamilNadu, India.  
E-mail: [uma.sathya123@yahoo.co.in](mailto:uma.sathya123@yahoo.co.in)
  - S. Abraham Lincon, Professor, Dept of E & I, Faculty of Engineering, Annamalai University, Chidambaram, TamilNadu, India.

$Y_k$ : Observation vector  
 $X(n)$ : Input signal  
 $D(n)$ : Desired signal  
 $E(n)$ : Error signal  
 $V(n)$ : Interfering noise

## 2. PARTICLE FILTER

The particle filters are also known as Sequential Monte Carlo methods (SMC), are sophisticated model estimation techniques based on simulation. The particle filter aims to estimate the sequence of hidden parameters,  $x_k$  for  $k = 0, 1, 2, 3, \dots$ , based only on the observed data  $y_k$  for  $k = 0, 1, 2, 3, \dots$ . All Bayesian estimates of  $x_k$  follow from the posterior distribution  $p(x_k | y_0, y_1, \dots, y_k)$ . In contrast, the MCMC or importance sampling approach would model the full posterior  $p(x_0, x_1, \dots, x_k | y_0, y_1, \dots, y_k)$ .

Particle methods assume  $x_k$  and the observations  $y_k$  can be modeled in this form:  $x_0, x_1, \dots$  is a first order Markov process such that and with an initial distribution  $p(x_0)$ .

$$x_k | x_{k-1} \sim p_{x_k | x_{k-1}}(x | x_{k-1})$$

The observations  $y_0, y_1, \dots$  are conditionally independent provided that  $x_0, x_1, \dots$  are known. In other words, each  $y_k$  only depends on  $x_k$

$$y_k | x_k \sim p_{y|x}(y | x_k)$$

One example from this scenario is

$$x_k = g(x_{k-1}) + w_k \tag{1}$$

$$y_k = h(x_k) + v_k \tag{2}$$

where both  $w_k$  and  $v_k$  are mutually independent and identically distributed sequences with known probability density functions and  $g(\cdot)$  and  $h(\cdot)$  are known functions. These two equations can be viewed as state space equations and look similar to the state space equations for the Kalman filter. If the functions  $g(\cdot)$  and  $h(\cdot)$  are linear, and if both  $w_k$  and  $v_k$  are Gaussian, the Kalman filter finds the exact Bayesian filtering distribution. If not, Kalman filter based methods are a first-order approximation (EKF) or a second-order approximation (UKF in general, but if probability distribution is Gaussian a third-order approximation is possible). Particle filters are also an approximation, but with enough particles they can be much more accurate.

In the implementation of particle filters, there are three important operations:

1. Generation of particles (sample step),
2. Computation of the particle weights (importance step), and
3. Resampling.

The first two steps form the particle filter called Sequential Importance Sampling (SIS) filter. The filter that performs all three operations is called Sample Importance Resampling Filter (SIRF).

The particle filter is a useful tool to perform dynamic state estimation via Bayesian inference. It provides great efficiency and extreme flexibility to approximate any functional nonlinearity. The key idea is to use samples, also called particles, to represent the posterior distribution of the state given a sequence of sensor measurements. As new information arrives, these particles are constantly re-allocated to update the estimation of the state of the system. The efficiency and accuracy of the particle filter depends mainly on two key factors: the number of particles used to estimate the posterior distribution and the propagation function used to re-allocate these particles at each iteration. The standard implementation of the filter specifies both factors beforehand and keeps them fixed during the entire operation of the filter.

## 3. ADAPTIVE PARTICLE FILTER

An adaptive filter is a filter that self-adjusts its transfer function. The optimization algorithms is so complex that adaptive filters are used as digital filters. This is required for some applications where parameters of the desired processing operation are not known in advance. The adaptive filter uses feedback in the form of an error signal. The particle filter has emerged as a useful tool for problems requiring dynamic state estimation. The efficiency and accuracy of the filter depend mostly on the number of particles used in the estimation and on the propagation function used to re-allocate these particles at each iteration. Both features are specified beforehand and are kept fixed in the regular implementation of the filter. In practice this may be highly inappropriate since it ignores errors in the models and the varying dynamics of the processes.

This work presents a self adaptive version of the particle filter that uses statistical methods to adapt the number of particles and the propagation function at each iteration. Furthermore, our method presents similar computational load than the standard particle filter.

In this paper, we present a self adaptive particle filter that shown in Figure 1 uses statistical methods to select an appropriate number of particles and a suitable propagation function at each iteration. In this, after setting a threshold of

effective particle number and the reduction rate, the fading factor was chosen by the variance of importance weights recursively and adaptively.

Variance reduction is always realized by introducing the correlation into the sample set. Here we adopt an idea of variance reduction to achieve the aim of reducing the weights variance of particles

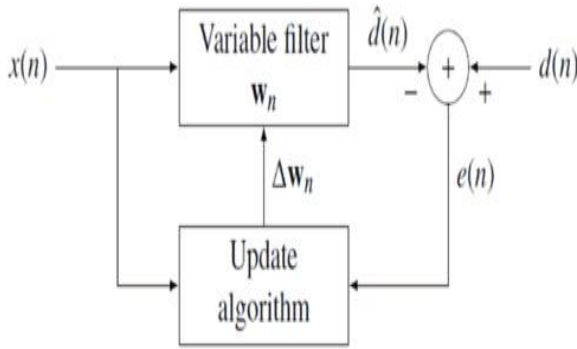


Figure 1. Adaptive filter structure

The input signal  $x(n)$  is the sum of a desired signal  $d(n)$  and interfering noise  $v(n)$

$$x(n) = d(n) + v(n) \tag{3}$$

The variable filter has a Finite Impulse Response (FIR) structure. For such structures the impulse response is equal to the filter coefficients. The coefficients for a filter of order  $P$  are defined as

$$\mathbf{w}_n = [w_n(0), w_n(1), \dots, w_n(p)]^T \tag{4}$$

The error signal or cost function is the difference between the desired and the estimated signal

$$e(n) = d(n) - \hat{d}(n) \tag{5}$$

The variable filter estimates the desired signal by convolving the input signal with the impulse response. In vector notation this is expressed as

$$\hat{d}(n) = \mathbf{w}_n * \mathbf{x}(n) \tag{6}$$

Where

$$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-p)]^T \tag{7}$$

is an input signal vector. Moreover, the variable filter updates the filter coefficients at every time instant

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \Delta \mathbf{w}_n \tag{8}$$

Where the parameter  $\Delta \mathbf{w}_n$  is a correction factor for the filter coefficients. The adaptive algorithm generates this correction factor based on the input and error signal  $x(n)$

### 3.1 Adaptive Algorithm

To alleviate the degeneration on particle set, it seems necessary to reduce the variance of weights of prior particles.

#### Variance reduction of weights

Variance reduction is always realized by introducing the correlation into sample set. For PF, the principles of reducing the weights variance should be as follows

- 1) For the particle having relatively high weights, we decrease their weights.
- 2) For the particle having relatively low weights, we increase their weights.
- 3) Keep the original order of all particle weights remaining unchanged, i.e. keep a higher weight to be higher and a lower weight to be lower.

The above proposed theorem subsequently provides powerful evidence that a small (larger than 0 and over smaller than 1) exponent factor to all weights can decrease the weights variance of particles, i.e., increase the effective particle number. And, the smaller the exponent factor is the larger the effective particle number.

#### Step 1:

Select the parameter that determines the rate of variance reduction at each time  $k$  and set a threshold  $N_{thr}$  for effective particle number, let  $t=n-1, w_k^0 = w_k$

#### Step 2:

Compute the effective particle number  $N_{eff}$

#### Step 3:

While  $N_{eff} < N_{thr} \alpha = t/n, w_k = (w_k^0)^\alpha, t = t-1$

#### Step 4:

Normalize the weight for each particle

$$w_k^i = w_k^i / \sum w_k^i$$

#### Step 5:

Estimate the state

$$X_k = \sum w_k^i x_k^i$$

**Step 6:**

Estimate the covariance  $P_k = \Sigma(x_k^i - x_k) \sim w_k^i(x_k^i - x_k)^T$

**Step 7:**

Draw new particles  $x_k^i \sim N(x_k^i, p_k)$ , and set  $\{w_k^i = 1/N\}^N$

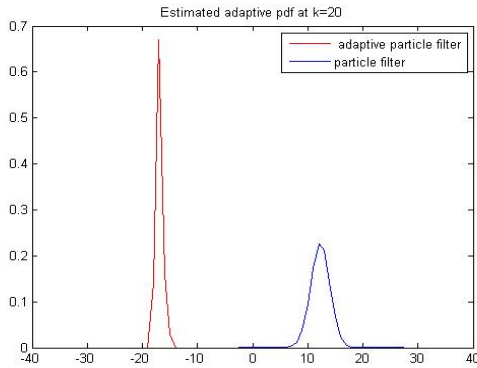
**4.UNIVARIATE NON-STATIONARY GROWTH MODEL AND RESULTS**

To illustrate some of the advantages of APF over PF, let us now consider an example [10-11], in which we estimate a model called Univariate Nonstationary Growth Model (UNGM), which is previously used as benchmark. What makes this model particularly interesting in this case is that it is highly nonlinear and bimodal, so it is really challenging for traditional filtering techniques. The dynamic state space model for UNGM can be written as

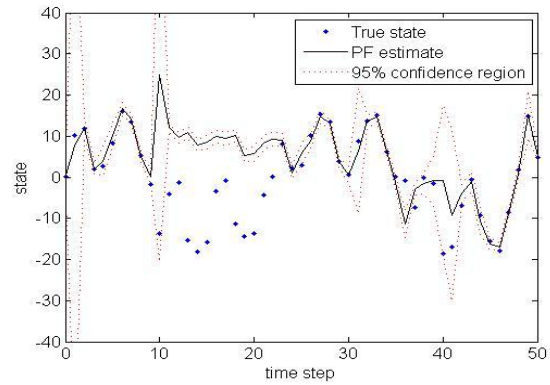
$$\begin{aligned}
 x_n &= \alpha x_{n-1} + \beta \frac{x_{n-1}}{1 + x_{n-1}^2} + \gamma \cos(1.2(n-1)) + u_n \\
 y_n &= \frac{x_n^2}{20} + v_n, n = 1, \dots, N
 \end{aligned}
 \tag{9}$$

The cosine term in the state transition equation simulates the effect of time-varying noise. From Eq no.9, we choose  $\alpha=0.5, \beta=25, \gamma=8$ .

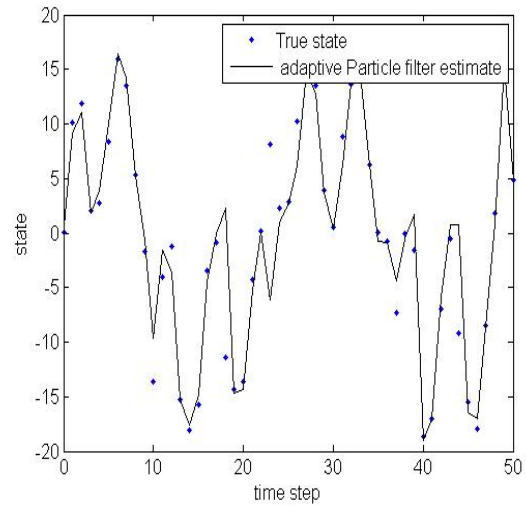
For  $N=100$  Particles, the root mean square error (RMSE) curves of the estimated results are 11.1394 (particle filter) and



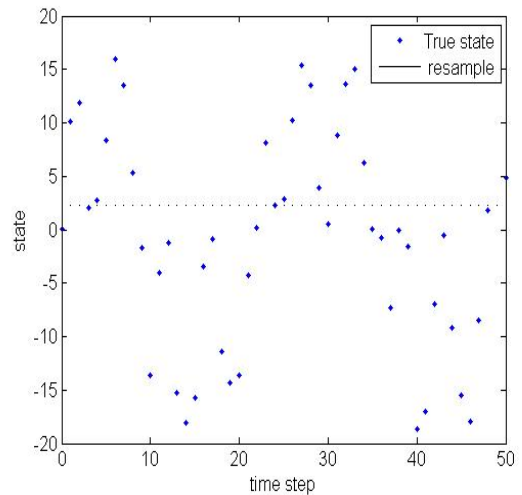
2.8254 (Adaptive particle filter). Fig. 2 and Table 1 shows that APF has higher accuracy than particle filter in the estimation of this nonlinear system with non-Gaussian noise.



(b)



(c)



(d)

Fig.2. (a) Estimation of Adaptive pdf.  
 (b) Estimation of Particle filter.  
 (c) Estimation of Adaptive Particle filter.  
 (d) Resample of Adaptive Particle filter.

By simulating the state transition equation of adaptive particle filter based on univariate non-stationary growth model, we obtain the results which is shown in Figure 2 and the RMSE value of particle filter and adaptive particle filter are shown in the Table 1

Table 1 RMSE Performance of Two Algorithms

Runs	Particle filter	Adaptive particle filter
100	11.1394	2.8254
200	21.9882	9.0909
300	11.0651	3.4106
500	18.1648	2.3371

[8] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," IEE Proceedings F: Radar and Signal Processing, vol. 140, no. 2, pp. 107-113, 1993.  
[9] M. S. 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, 2002.

## 5. CONCLUSION

In this paper, we have developed an Adaptive particle filter that overcomes the major problem of particle degeneracy. Here we have adopted the idea of resampling operation to overcome the problem of particle degeneracy but it leads to the problem of sample impoverishment. So we proposed an algorithm based on variance reduction technique which reduces the variance of the particles. It overcomes both the problem and produce the particle with more accuracy compared to the prior methods. More experiments are necessary in order to validate the algorithm. Further improvements of the algorithm are necessary to reduce error completely.

## REFERENCES

[1] Gordon N J, Salmond D J, Smith A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F: Radar and Signal Processing, 1993, 140(2): 107-113  
[2] Giremus A, Tourneret J Y, Djuric P M. An improved regularized particle filter for GPS/INS integration. In: Proceedings of the 6th International Workshop on Signal Processing Advances in Wireless Communications. New York, USA: IEEE 2005. 1013-1017  
[3] D. B. Ward, E. A. Lehmann, and R. C. Williamson, "Particle filtering algorithms for tracking an acoustic source in a reverberant environment," IEEE Transactions on Speech and Audio Processing, vol. 11, no. 6, pp. 826-836, 2003.  
[4] Pitt M, Shephard N. Filtering via simulation: auxiliary particle filters. Journal of the American Statistical Association, 1999, 94(446): 590-599  
[5] Higuchi T. Monte Carlo filtering using the genetic algorithm operators. Journal of Statistical Computation and Simulation, 1997, 59(1): 1-23  
[6] Orguner U, Gustafsson F. Risk sensitive particle filters for mitigating sample impoverishment. IEEE Transactions on Signal Processing, 2008, 56(10): 5001-5012  
[7] Cheng S Y, Zhang J Y. Fission bootstrap particle filtering. Acta Electronica Sinica, 2008, 36(3): 500-504