American Journal of Engineering Research (AJER)
e-ISSN : 2320-0847 p-ISSN : 2320-0936
Volume-02, Issue-09, pp-228-233
www.ajer.org

Research Paper Open Access

On the choice of importance of Resampling schemes in Particle Filter

Mr. J.Joseph Ignatious, Dr.S.Abraham Lincon,
Department of Electronics and Instrumentation Engineering,
Annamalai University, Annamalai Nagar, India.

Abstract: - Particle filter is a powerful tool for tracking of targets in any system .The systems which are non-linear and non Gaussian are commonly occur in practice. This paper investigates various resampling algorithms which are available on particle filter .Many tracking methods have been developed but still there are difficulties in continuous tracking of the target. This work aims on the preference of resampling algorithms in tracking .The performance of resampling is evaluated in terms of their MSE value of SIR filter with that of resampling schemes.

Index Terms: - *SIR Filter, Mean Square Error(MSE), Resampling, Particle filter*

I. INTRODUCTION

The main strength of the particle filters is that they can be used in filtering even in problems where we cannot compute the distribution analytically. It is only needed to be known proportionally. Importance sampling is a technique for getting samples from analytical distributions. The idea is to get samples from another distribution, like normal distribution, and then assign the weight according to the real distribution. This estimation of the filtering distributions is done in every time instant from the beginning. The weight of the particles are calculated as follows:

$$\omega^{(i)}(k) = \omega^{(i)}(k-1)\frac{p(y(k)|x^{(i)}(k)p(x(k)|x^{(i)}(k))}{p(x^{(i)}(k)|x^{(i)}(k),y(k))}$$

where $\omega^{i}(k)$ is weight of the particle, x(k) one possible system state sampled from the proposal distribution, $p(x(k)|x^{i}(k-1))$ is its prior probability and y(k) are the observation, $p(y(k) \mid x^{i}(k))$ is its likelihood and $p(x^{i}(k)|x^{i}(k-1), y(k))$ is the value of the density function of this particle's proposal distribution. Then the weights are normalized so that their sum equals unity. The new weights are:

$$\hat{w}^{(i)}(k) = \frac{\omega^{(i)}(k)}{\Sigma^{n}_{j=1}\,\omega^{(i)}(k)}$$

From these equations it can be known, how the prior distribution differs from the real one. It would be fine if samples match better with the real distribution. For example regular particle filters such as bootstrap and SIR filters use prior distribution for the particle in the prediction stage. If prior distribution is far from the real one it is very likely that many particles end up to the low likelihood area and will not be in the resampling stage. The main techniques of increasing particles in the resampling stage are increasing the number of the particle sample size, prior editing and auxiliary variable. Easiest way to get more particles to the resampling stage is by simply increasing proposal sample size technique. If we have more proposals it is likely that more of them will get in to high likelihood area .In prior editing, the proposals are computed one by one. If the likelihood of the observation for the proposal is high enough than the threshold value, we accept the proposal. Otherwise it is rejected. This is continued until we have N accepted proposals, which allows us to get more proposals from the particles in high likelihood area and none from the particles in low likelihood area to get more particles to the resampling stage which makes the future estimates more accurate. We compare the estimation and prediction performance of the SIR filters with that of resampling schemes based on the following metrics. MSE is defined by

MSE = $\frac{1}{N}\sum_{n=1}^{N} Xn - \widetilde{\mathrm{Xn}}$

The analysis in this paper is related to the Sample Importance Resampling (SIR) type of PFs. However, the analysis is compared to various resampling schemes. First, in Section 2 we provide a brief review of the resampling operation. Then in section 3 we focus on the performance and analysis of several techniques that have been proposed to implement the resampling step and the summary of our work is outlined in section 4.

II. REVIEW OF RESAMPLING SCHEMES.

Besides prediction stage, particle filter also needs resampling stage . In this stage the particles with high weights are multiplied and the ones with low weight disappear. Even though the resampling decreases the number of the particles at current time instant, it also makes the future estimates more accurate. That is because after resampling ,there are more particles to describe high likelihood area. Resampling can be performed with numerous different algorithms like Multinomial resampling, Systematic resampling, Residual resampling, Stratified sampling or Deterministic sampling. Algorithms differ in terms of computational complexity, variance of the of the number of particles and bias. The above mentioned sampling techniques used for simulations in this paper since they are the best unbiased resampling algorithms . These resampling algorithms determine how many copies of each weighted particles made to represent unweighted particles at the next point in time.

2.1 SIR Particle Filter

The Sampling-Importance Resampling (SIR) is motivated from the bootstrap techniques. Bootstrap technique is a collection of computationally intensive methods that are based on resampling from the observed data [1], [2], [3]. The intuition of bootstrapping is to evaluate the properties of an estimator through the empirical cumulative distribution function (cdf) of the samples instead of the true cdf.. The resampling step is aimed to eliminate the samples with small weights and duplicate the samples with importance weights. The steps of SIR proceeds as follows:

Draw Np random samples $\{x^{(i)}\}\begin{cases} Np \\ i=1 \end{cases}$ from proposal distribution q(x);

Calculate importance weights ω (i) ∝ p(x)/q(x) for each sample x(i);

Normalize the importance weights to obtain $\widehat{\mathrm{w}}$(i);

Resample with replacement N times from the discrete set $\{x^{(i)}\}\begin{cases} Np \\ i=1 \end{cases}$ where the probability of resampling from each x(i) is proportional to $\widehat{\mathrm{w}}$(i). Resampling usually (but not necessarily) occurs between two importance sampling steps. In resampling step, the particles and associated importance weights {x(i), $\widehat{\mathrm{w}}$(i)} are replaced by the new samples with equal importance weights (i.e.) = $\widehat{\mathrm{w}}$(i)=1/Np). Resampling can be taken at every step or only taken if regarded necessary.

• As justified in [4], resampling step plays a critical role in importance sampling since

(i) if importance weights are unevenly distributed, propagating the "trivial" weights through the dynamic system is a waste of computing power;

(ii) when the importance weights are distorted, resampling can provide chances for selecting "important" samples and restore the sampler for the future use, though resampling doesn't necessarily improve the current state estimate because it also introduces extra Monte Carlo variation.

• Resampling schedule can be deterministic or dynamic [5], [6]. In deterministic framework, resampling is taken at every k^{th} time step (usually k = 1). In a dynamic schedule, a sequence of thresholds (that can be constant or time-varying) are set up and the variance of the importance weights are monitored; resampling is taken only when the variance is above the threshold.

2.2 Multinomial resampling

[7]: The procedure reads as follows

- Produce a uniform distribution u ∼ U(0, 1), construct a cdf for importance weights calculate s= $\sum_{j=1}^{i} \widehat{\mathrm{w}}$(i)
- Find s_i such that $s_{i-1} \le u < s_i$, the particle with index *i* is chosen; Given {x(i), $\widehat{\mathrm{w}}$(i) }, for j = 1, ,Np, generate new samples x(j) by duplicating x(i) according to the associated

$\widehat{\mathrm{w}}$(i);

- Reset ω (i) = 1/Np.

Multinomial resampling uniformly generates Np new independent particles from the old particle set. Each particle is replicated Ni times (Ni can be zero), namely each x(i) produces Ni particles.

Note that here $\sum_{i=1}^{Np} Ni = Np$

$E[Ni] = Np\ \hat{w}(i)$, $Var[Ni] = Np\ \hat{w}(i)\ (1 - \hat{w}(i))$.

2.3. Residual resampling

[8]: Liu and Chen [5] suggested a partially deterministic resampling method calculates the number of times each particle is replicated except that it avoids when residual particles need to be resampled. The number of replications of a specific particle is determined by truncating the product of the number of particles and the particle weight The selection procedure is as follows [5]:

- For each i = 1, · · · ,Np, retain $k_i = [Np\ \omega\ (i)]$ copies of $x^{(i)}$ n ;
- Let $Nr = Np - k1 - \ldots - kNp$, obtain Nr from $\{x^{(i)}n\}$ with probabilities proportional to $Np\ \hat{w}(i) - k_i$ (i = 1,… ,Np);
- Reset ω (i) = 1/Np.

Residual resampling procedure is computationally cheaper than the conventional SIR and achieves a lower sample variance, and it doesnot introduce additional bias. Every particle in residual resampling is replicated.

2.4. Systematic resampling (or Minimum variance sampling)

This resampling**[9]** takes the previous method one step further by deterministically linking all the variables drawn in the sub-intervals. The procedure proceeds as follows:

- $u \sim U(0, 1)/Np$; j = 1;m = 0;i = 0;
- do while u < 1
- if m > u then • u = u + 1/Np; output x(i)
- else, pick k in {j,… ,Np}
- i = x(k), m = m + ω (i)
- switch (x(k), ω (k)) with (x(j), ω (j))
- j = j + 1, end if, end do

The systematic resampling treats the weights as continuous random variables in the interval (0, 1), which are randomly ordered. The number of grid points {u+k/Np} in each interval is counted . Every particle is replicated and the new particle set is chosen to minimize $Var[Ni] = E[(Ni - E[Ni])^2]$.

2.5 Stratified Sampling

[10];The idea of stratified sampling is to distribute the samples evenly (or unevenly according to their respective variance) to the sub regions dividing the whole space. Let g (statistics of interest) denote the Monte Carlo sample average of a generic function $g(x) \in RNx$, which is attained from importance sampling. Suppose the state space is decomposed into two equal, disjoint strata (subvolumes), denoted as a and b, for stratified sampling, the total number of Np samples are drawn from two strata separately and we have the stratified mean $g = 1/2\ (g_a + g_b)$, and the stratified variance

$$Var[g] = \frac{Var_a[g] + Var_b[g]}{4}$$

$$= \frac{Var_a[g] + Var_b[g]}{2NP}$$

where the second equality uses the facts that Vara[g] =2/Np Vara[g] and [Vara[g] = 2/Np Varb[g]. In addition, it can be proved that NpVar[g] = Var[g]

$$= N_P Var[\dot{g}] + \frac{(Ea[g] + Eb[g])^2}{4}$$

$$\geq N_P Var[\dot{g}]$$

Hence, the variance of stratified sampling Var g` is never bigger than that of conventional Monte Carlo sampling Var[g], whenever $Ea[g] \neq Eb[g]$. In general, provided the numbers of simulated samples from stratified a and b are Na and Nb ≡ Np−Na, respectively, becomes

$$Var[\dot{g}] = \frac{1}{4}\left(\frac{Var_a[g]}{Na} + \frac{Var_b[g]}{N_P}\right)$$

the variance is minimized when

$Na/Np = \sigma a / (\sigma a + \sigma b)$ and the achieved minimum variance is

$$Var[\dot{g}]min = \frac{(\sigma a + \sigma b)^2}{4Na}$$

Table –I shows different popular montecarlo methods.

Table-I

Author	Sampling method	Applied
Rubin	SIR	On/off line
-	Stratified sampling	On/off line
Gordon	Bootstrap	Online
-	QMC	On/offline
Bolic	RSR	On/offline

III. UNIVARIATE NON-STATIONARY GROWTH MODEL AND RESULTS

To illustrate some of the advantages of SIR Particle Filter with various resampling schemes mentioned in this paper.,Let us now consider an example[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 its highly nonlinear and bimodal, so it is really challenging for traditional filtering techniques. The dynamic state space model for UNGM can be written as

$$x_n = \alpha x_{n-1} + \beta \frac{x_n - 1}{1+x^2_{n-1}} + \cos(1.2(n)) + w_n$$

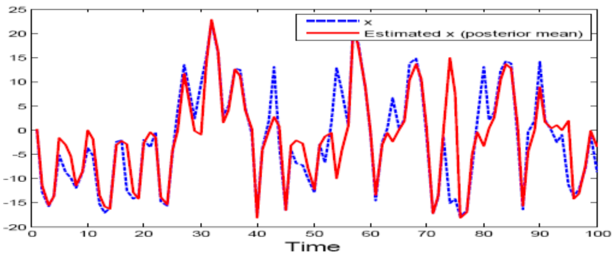
$$Y_n = x^2_n / 20 + v_{n,} \qquad n = 1,\ldots\ldots\ldots,N$$

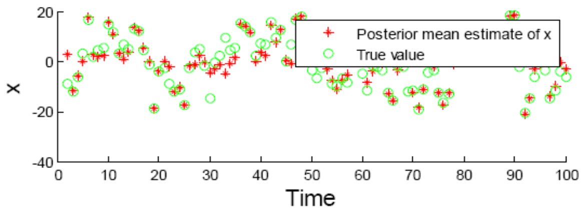
Table-II shows the comparison of root mean square error(RMSE) of SIR particle filter with that of The cosine term in the state transition equation simulates the effect of time-varying noise. From the above equation we choose α=0.5,β=25,γ=8.For N=100 particles ,the mean square error(MSE) curves of the estimated results of SIR particle filter is compared with different resampling schemes are shown in Figures and Table below.

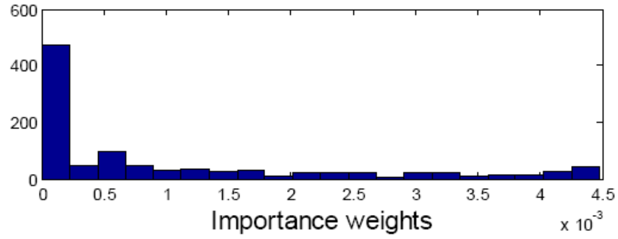
Table-II

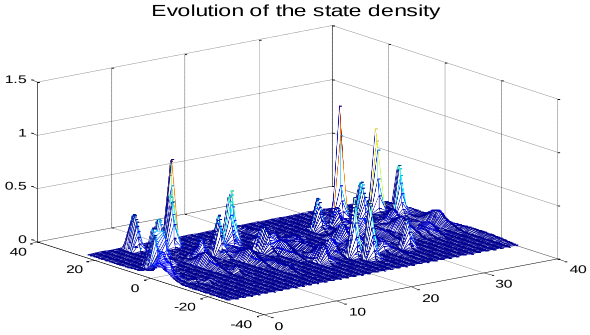
Resampling Schemes	ob	fob
SIR	2.839	2.132
Multinomial	-1.535	0.3845
Residual	-0.712	0.5523
Deterministic	0.802	0.594
Systematic	0.196	0.647

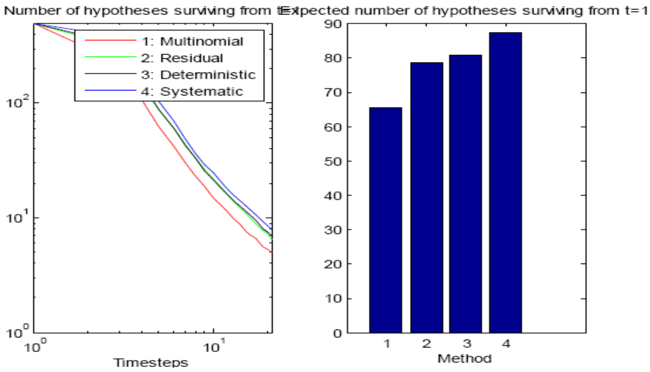
*ob- observation ,*fob-filtered observation











IV. CONCLUSION

The purpose of this work is to examine the application of particle filters in radar systems that use different resampling schemes. The performance of the algorithms were investigated in Matlab simulation . In practical applications of sequential Monte Carlo methods, residual, stratified, multinominal and systematic resampling are generally found to provide comparable results. Due to lack of complete theoretical analysis of its behavior, systematic resampling is often preferred because it is the easiest method to implement. From a theoretical point of view however only the residual and stratified resampling methods (as well as the combination of both) may be shown to dominate the basic multinomial resampling approach, in the sense of having lower covariance for all configurations of the weights.

REFERENCES

[1] G. A. Einicke and L. B. White, “Robust extended Kalman filter,” IEEE Trans. Signal Processing, vol. 47, no. 9, pp. 2596– 2599, Sept. 1999.

[2] W. R. Gilks and C. Berzuini, “Following a moving target – Monte Carlo inference for dynamic Bayesian models,” J. Roy. Statist. Soc., Ser. B, vol. 63, pp. 127–1546, 2001.

[3] B. Manly, Randomization, Bootstrap and Monte Carlo Methods in Biology, 2nd ed., CRC Press, 1997

[4] J. S. Liu and R. Chen, “Blind deconvolution via sequential imputation,” J. Amer. Statist. Assoc., vol. 90, pp. 567–576, 1995.

[5] “Sequential Monte Carlo methods for dynamical systems,” J. Amer. Statist. Assoc., vol. 93, pp. 1032–1044, 1998.

[6] J. S. Liu, R. Chen, and T. Logvinenko, “A theoretical framework for sequential importance sampling with resampling,” in Sequential Monte Carlo Methods in Practice, A. Doucet, J. F. G.

[7] de Freitas, N. J. Gordon, Eds. Berlin: Springer Verlag, 2001.

[8] Multiple Imputation for Nonresponse in Surveys, New York: Wiley, 1987.

[9] T. Higuchi, “Monte Carlo filter using the genetic algorithm operators,” J. Statist. Comput. Simul., vol. 59, no. 1, pp. 1–23, 1997.

[10] “Monte Carlo filter and smoother for non-gaussian nonlinear state space models,” J. Comput. Graph. Statist., vol. 5, no. 1, pp. 1–25, 1996.

[11] W. H. Press and G. R. Farrar, “Recursive stratified sampling for multidimensional Monte Carlo integration,” Computers in Physics, vol. 4, pp. 190–195, 1990

[12] 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

J.Joseph Ignatious received his BE degree in Electronics and communication Engineering in April. 2005, in Nov. 2002 he received the ME degree in Process control and Instrumentation from Annamalai University, Chidambaram. He has since Feb. 2011 been pursuing the PhD degree at the division of Instrumentation Engineering Annamalai University .

S. Abraham Lincon obtained his B.E degree is Electronics and Instrumentation Engineering in the year 1984 and received M.E. degree in Power System Engineering in 1987 and another M.E degree in Process Control and Instrumentation Engineering in 2000 from the Annamalai university, Chidambaram. Presently he is working as a professor of the Department of Instrumentation Engineering in Annamalai University. His areas of research are Process Control, Fault Detection and Diagnosis and Multivariable Control.