

# Bearing Only Tracking using a Particle Filter

Joselin George<sup>1</sup>, M.Mathurakani<sup>2</sup>

PG Scholar, Department of Electronics and Communication, Toc H Institute of Science and Technology, Ernakulam, India,  
joselingearge90@gmail.com<sup>1</sup>

Prof. Head of the Department (M Tech), Department of Electronics and Communication, Toc H Institute of Science and Technology,  
Ernakulam, India, Formerly Scientist, DRDO, Ministry of Defence India<sup>2</sup>

**Abstract**— Target Tracking has always been a challenging problem arising in different contexts ranging from military applications to biology. The tracking problem consists of computing the best estimate of the target's trajectories based on noisy measurements (observations). This tracking problem can be modeled as Dynamic State Space Model. Either the system model or the measurement model can be nonlinear in terms of state variables. Also the uncertainty in the process model and/or in the measurement model may be non-Gaussian. In such cases the solutions obtained from the traditional methods are much complex. Particle filter offers a general numerical tool to approximate the posterior density function for the state in nonlinear and non-Gaussian filtering problems. This paper considers the application of particle filtering technique to a target tracking example, in which a radar sends a signal towards a target (aircraft) and estimates the state (position and velocity) of the target using the observation (Bearing angle). State model and measurement model have been derived for the proposed target tracking problem.

**Keywords**—Target Tracking, Dynamic state space models, Nonlinear Systems, Non-Gaussian Systems, Particle Filter, Monte Carlo Simulations and Relative RMSE.

## INTRODUCTION

Target tracking has been an active research area in image/signal processing in recent years. It has many potential applications in the fields of intelligent robots [1], monitoring and surveillance [2], human computer interfaces [3], smart rooms [4]-[5], vehicle tracking [6], biomedical image analysis [7], video compression [8], etc. [9]-[10]. Such application problems require estimation of the state of the system that changes over time using a sequence of noisy measurements made on the system. For such application areas, it is becoming important to include elements of nonlinearity and non-Gaussian in order to model accurately the underlying dynamics of the physical system. Such problems can be written in the form of the so-called Dynamic State Space (DSS) model. Moreover, it is crucial to process data on-line as it arrives, which results in a recursive method of estimation. Numerous approaches have been proposed to track moving objects, such as the Kalman filter, the extended Kalman filter, approximate grid-based filter and the Particle filter.

## PARTICLE FILTERS

Particle Filter [14]-[15] is used to perform filtering for problems that can be described using dynamic state space modeling [14]. In most practical scenarios, these models are non-linear and the densities involved are non-Gaussian. Traditional filters like the Classical Kalman Filter [11], Extended Kalman Filter [12] are known to perform poorly in such scenarios. The performance of PFs on the other hand, is not affected by these conditions. PFs are Bayesian in nature and their goal is to find an approximation to the posterior density of the states of interest (e.g. position of a moving object in tracking, or transmitted symbol in communications) based on observations corrupted by additive noise which are inputs to the filter. This is done using the principle of Importance Sampling (IS) whereby, samples (particles) are drawn from a known density (Importance Function (IF)) and assigned appropriate weights based on the received observations using IS rules [14]. This weighted set of samples represents the posterior density of the state and can be used to find all kinds of estimates of the state (like Minimum Mean Square Error MMSE). PF algorithms allow for recursive propagation of this density as the observations become available. However, performance of this scheme is affected by weight degeneracy [15]. Because of this, after several sampling periods there are only a few particles with significant weights while those of the rest become negligible. This problem is solved by introducing resampling which discards particles with negligible weights and replicates those with large weights while preserving constant number of particles. These operations form the traditional PF algorithm known as the Sampling Importance Resampling Filter (SIRF).

The main steps in the particle filter algorithm include:

1) Initialization

Draw a set of particles for the prior  $p(X_0)$  to obtain  $\{X_0^{(i)}, w_0^{(i)}\}_{i=1}^N$ , let  $k=1$ .

2) Sampling

(a) For  $i=1, \dots, N$

Sample  $X_k^{(i)}$  from the proposal distribution  $p(X_k^{(i)}|X_{k-1}^{(i)})$ .

(b) Evaluate the new weights

$$w_k^{(i)} = p(Z_k|X_k^{(i)}), i = 1, 2, \dots, N$$

(c) Normalize the weights

(1a)

$$w_k^{(i)} = \frac{w_k^{(i)}}{\sum_{j=1}^N w_k^{(j)}}, i = 1, 2, \dots, N \tag{1b}$$

3) Output

Output a set of particles  $\{X_k^{(i)}, w_k^{(i)}\}_{i=1}^N$  that can be used to approximate the posterior distribution as

$$p(X_k | Z^k) \approx \sum_{i=1}^N w_k^{(i)} \delta(X_k - X_k^{(i)})$$

and the estimate as

$$E_{p(g|Z^k)}(f_k(X_k)) \approx \sum_{i=1}^N w_k^{(i)} f_k(X_k^{(i)}), \text{ where } \delta(g) \text{ is the Dirac delta function.}$$

4) Resampling

Resample particles  $X_k^{(i)}$  with probability  $w_k^{(i)}$  to obtain  $N$  independent and identically distributed random particles  $X_k^{(j)}$ , approximately distributed according to  $p(X_k | Z^k)$ .

5)  $k=k+1$ , go to step2.

**TARGET TRACKING EXAMPLE**

This paper considers a typical target tracking example in which the position and velocity of a target is estimated using a 2D constant velocity model. In this case the bearing angle is the measurement which is applied to the filter. From the dynamic state space model, it can be seen that the model has linear state equation and nonlinear measurement equation.

The dynamic state space model of the above tracking example is given below.

$$x_{t+1} = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x_t + w_t \tag{2a}$$

$$y_t = (\varphi) = \arctan(p_y | p_x) + e_t \tag{2b}$$

where the state vector  $x_t = [p_x \ p_y \ v_x \ v_y]^T$  i.e. position and velocity. We have discarded the height component, since a level flight is considered. The sample time is taken as constant and denoted by  $T=1$ sec. The measurement noise  $e_t$  is Gaussian with zero mean and variance  $R=2$ . The process noises are assumed Gaussian with zero mean and covariances  $Q=Diag(5, 5, 0.01, 0.01)$ .

The simulation for the tracking of the target using particle filter is done using MATLAB. The various steps done in this simulation is given below.

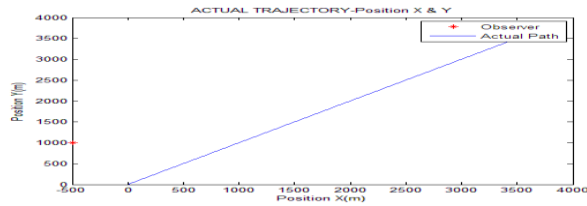
- i) Generation of the trajectory.
- ii) Calculation of corresponding measurement (bearing angle) from the trajectory and corrupting with noise.
- iii) Simulating particle filter using the noise corrupted measurement generated for 100 Monte Carlo Simulations.
- iv) Comparing the estimate with that of the actual path.

Here a linear trajectory is considered. Trajectory is generated using the equations of motion and all the values of position and velocity in the x and y co-ordinate at each time instant along with the corresponding angle (in degrees with respect to x axis) is stored. Then the measurement is corrupted with noise before giving to the filter.

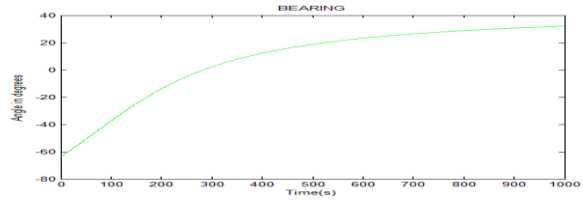
Details of parameters used in the simulation of linear trajectory are given below.

Parameter	Value
Number of Monte Carlo Simulations	100
Initial Position $[p_x \ p_y]$ (m)	[500 -1000]
Constant Velocity(m/s)	5
Initial state Covariance Q	Diag (5, 5, 0.01, 0.01)
Measurement noise variance R	2

The trajectory of aircraft that are used for simulation using particle filter are shown below. The bearing is measured from the generated trajectory.



(a)

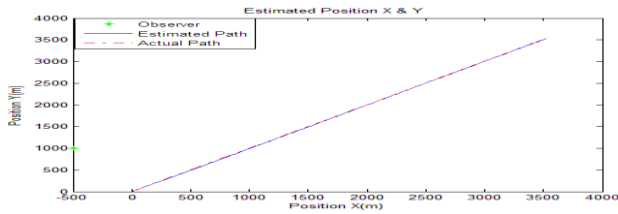


(b)

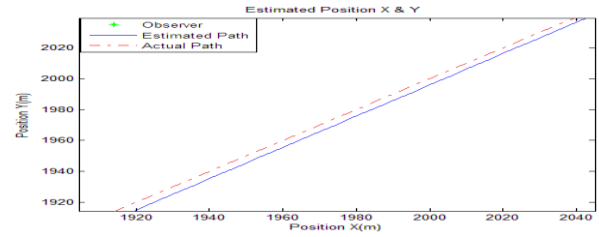
Figure 1: (a) shows the linear trajectory generated. (b) shows the measured bearing angle from the generated trajectory.

## SIMULATION RESULTS

In this section, Estimation of position and velocity using Particle Filter for linear trajectory with  $N=1000$  particles are shown below.

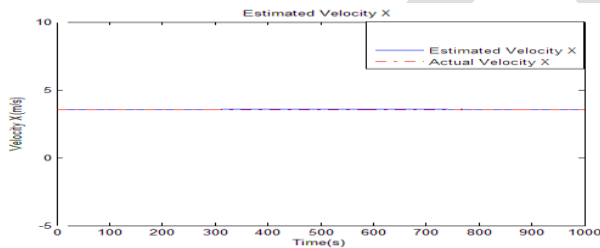


(a)

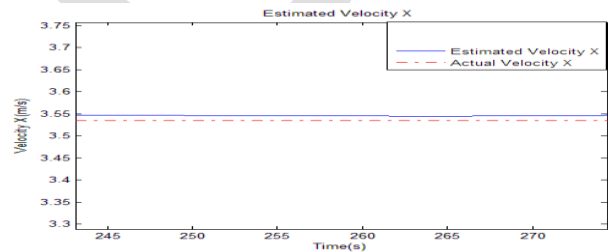


(b)

Figure 2: (a) Estimation of Position of linear trajectory using 100 Monte Carlo simulations with  $N=1000$  particles for complete time samples. (b) Zoomed version.

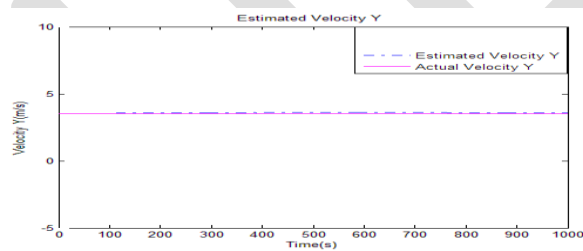


(a)

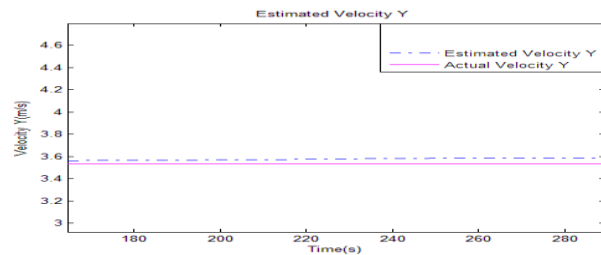


(b)

Figure 3: (a) Estimation of Velocity X of linear trajectory using 100 Monte Carlo simulations with  $N=1000$  particles for complete time samples. (b) Zoomed version.



(a)



(b)

Figure 4: (a) Estimation of Velocity Y of linear trajectory using 100 Monte Carlo simulations with  $N=1000$  particles for complete time samples. (b) Zoomed version.

## ACKNOWLEDGEMENT

The authors would like to thank Toc H Institute of Science and Technology for their support.

## CONCLUSION

In this paper, we verified particle filtering technique for a target tracking example using a linear state model and a non-linear measurement model with additive white Gaussian noise in MATLAB.

## REFERENCES

- [1] C. Kwok, D. Fox, M. Meila, "Adaptive Real-Time Particle Filter for Robot Localization," *Proceedings of Robotics and Automation*, Vol. 2, pp. 2836-2841, Sept. 2003.
- [2] Y. Cui, S. Samarasekera, Q. Huang, M Greiffenhangen, "Indoor Monitoring Via the Collaboration Between a Peripheral Sensor and a Foveal Sensor," *Proceedings of Visual Surveillance*, pp. 2-9, Jan. 1998.
- [3] G. R. Bradski, "Real Time Face and Object Tracking as a Component of a Perceptual User Interface," *IEEE workshop on Applications of Computer Vision*, Princeton, pp. 214-219, 1998.
- [4] S. S. Intille, J. W. Davis, A. F. Bobick, "Real-Time Closed-World Tracking," *Proceedings of Vision and Pattern Recognition*, pp. 697-703, 1997.
- [5] C. R. Wren, A. Azarbayejani, T. Darrel, A. P. Pentland, "Pfinder: Real- Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine intelligence*, Vol. 19, pp. 780-785, July 1997.
- [6] M. Betke, E. Haritaoglu, S. S. Davis, "Multiple Vehicle Detection and Tracking in Hard Real-Time," *Proceedings of Intelligent Vehicles Symposium*, pp. 351-356, Sept. 1996.
- [7] L. Pan, J. L. Prince, J. A. C. Lima, N. F. Osman, "Fast Tracking of Cardiac Motion Using 3D-HARP," *IEEE Transactions on Biomedical Engineering*, Vol. 52, pp. 1425-1435, Aug. 2005.
- [8] A. Eleftheriadis, A. Jacquin, "Automatic Face Location Detection and Tracking for Model-Assisted Coding of Video Teleconference Sequence at Low Bit Rates," *Signal Processing - Image Communication*, Vol. 7, pp. 231-248, 1995.
- [9] D. Comaniciu, V. Ramesh, P. Meer, "Real-Time Tracking of Non-Rigid Objects Using Mean Shift," *IEEE Conference on Computer Vision and Pattern Recognition*, South Carolina, Vol. 2, pp. 142-149, 2000
- [10] P. Li, T. Zhang, "Visual Contour Tracking Based on Particle Filters," *Proceedings of Generative-Model-Based Vision*, Copenhagen, pp. 61-70, June 2002.
- [11] Monson H. Hayes "Statistical Digital Signal Processing And Modeling", John Wiley & Sons, Inc. 1996, 371- 377.
- [12] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. "A tutorial on particle filters for online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Trans. Signal Process.*, Vol 50 no.2, pp.174-188. Feb.2002.
- [13] Papoulis, S. Unnikrishna Pillai, "Probability, Random Variables, and Stochastic Processes" Tata McGraw-Hill Education 2002.
- [14] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential Monte Carlo methods for Bayesian filtering," *Statistics and Computing*, pp. 197-208, 2000.
- [15] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer Verlag, New York, 2001