FEED BACK PARTICLE FILTER BASED ALGORITHMS AND APPLICATIONS: A CRITICAL REVIEW

Harish Kumar
Electronics and Telecommunication Department, Jabalpur Engineering College, Jabalpur, India
harishvarshney1990@gmail.com

Dr. Agya Mishra
Electronics and Telecommunication Department, Jabalpur Engineering College, Jabalpur, India
agyamishra@gmail.com

Abstract: Feedback particle filter introduced a new approach to approximate non-linear filtering problem of the won ham filter. And Feedback particle filter has significantly lower variance, when compared to Bootstrap particle filter also reducing MSE at lower no. of particle. And it is useful in Neuroscience to involve coupled oscillators. It is also applied in Neuromorphic implementations for Bayesian inference in Human Brain, where the firing of Neurons is modeled as a Poisson process. And it has Application in Target Tracking.

Key Words: Feedback particle filter, speech enhancement algorithm, continuous transition between prior and posterior, optimal transport formulation, tractable algorithm for non-linear filtering.

1. INTRODUCTION

Particle filters or Sequential Monte Carlo (SMC) methods are a set of on-line posterior density estimation algorithms that estimate the posterior density of the state-space by directly implementing the Bayesian recursion equations [8]. SMC methods use a grid-based approach, and use a set of particles to represent the posterior density [8]. Feedback particle filter approximate non-linear filtering problems of the won-ham filter [9]. And Feedback particle filter has significantly lower variance, when compared to Bootstrap particle filter. And it is useful in Neuroscience to involve coupled oscillators. It is also applied in Neuromorphic implementations for Bayesian inference in Human Brain, where the firing of Neurons is modeled as a Poisson process. And it has Application in Target Tracking, Altitude estimation, Visual tracking and Robotic localization [6].

The feedback particle filter introduced a new approach to approximate nonlinear filtering, motivated by techniques from mean-field game theory. The filter is defined by an ensemble of controlled stochastic systems [9].

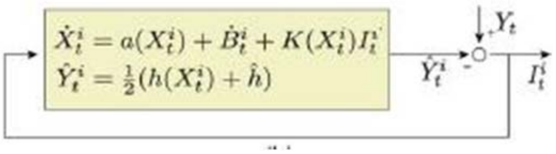

Fig.1. Feedback Particle Filter [9]

The stochastic differential equations (SDEs) are expressed in [9],

$$\mathrm{d}X_t = a(X_t)\mathrm{d}t + \sigma_B \mathrm{d}B_t, \quad (1)$$

$$\mathrm{d}Z_t = h(X_t)\mathrm{d}t + \sigma_W \mathrm{d}W_t \quad (2)$$

Where $X_t \epsilon R$ is the state at time t, $Z_t \epsilon R$ is the observation, a(.), h(.) are C^1 functions, and $\{B_t\}$, $\{W_t\}$are mutually independent standard Wiener processes. The objective of the filtering problem is to compute or approximate the posterior distribution of X_t given the filtration. The posterior P^* is defined so that, for any measurable set A C R [9]

$$\int_{x \in A} p^*(x,t)\mathrm{d}x = \mathsf{P}\{X_t \in A | \mathcal{Z}_t\}. \quad (3)$$

The model for the i^{th} particle is defined by a controlled system

$$\mathrm{d}X_t^i = a\left(X_t^i\right)\mathrm{d}t + \sigma_B \mathrm{d}B_t^i + \mathrm{d}U_t^i \quad (4)$$

Where $X_t^i \epsilon R$ is the state for the ith particle at time t, U_t^i is its control input, and $\{B_t\}$ are mutually independent standard Wiener processes. Throughout the paper, we denote conditional distribution of a particle X_t^i given Z_t by p where, just as in the definition of p*[9]

$$\int_{x \in A} p(x,t)\mathrm{d}x = \mathsf{P}\left\{X_t^i \in A \mid \mathcal{Z}_t\right\} \tag{5}$$

The initial conditions $\{X_0^i\}_{i=1}^N$ are assumed to be independent and identically distributed, and drawn from initial distribution $p^*(x, 0)$of X_0. The control problem is to choose the control input so that p approximates p*

The optimally controlled dynamics of the ith particle have the following it's form [9]

$$\mathrm{d}X_t^i = a\left(X_t^i\right)\mathrm{d}t + \sigma_B \mathrm{d}B_t^i + \underbrace{\mathsf{K}\left(X_t^i, t\right)\mathrm{d}I_t^i + \Omega\left(X_t^i, t\right)\mathrm{d}t}_{\text{optimal control, } \mathrm{d}U_t^{i*}} \tag{6}$$

In which $\Omega(x,t) := (1/2)\sigma_W^2 \mathsf{K}(x,t)\mathsf{K}'(x,t)$, $\mathsf{K}'(x,t) = (\partial \mathsf{K}/\partial x)(x,t)$and I_t^i is a certain modified form of the innovation process that appears in the nonlinear filter

$$\mathrm{d}I_t^i := \mathrm{d}Z_t - \frac{1}{2}\left(h\left(X_t^i\right) + \hat{h}_t\right)\mathrm{d}t \tag{7}$$

Where $\hat{h}_t := \mathsf{E}[h(X_t^i)|\mathcal{Z}_t] = \int h(x)p(x,t)\mathrm{d}x$,we approximate

$$\hat{h}_t \approx \hat{h}_t^{(N)} := \frac{1}{N}\sum_{i=1}^{N} h\left(X_t^i\right) \tag{8}$$

The gain function K is shown to be the solution to the following Euler–Lagrange boundary value problem (E-L BVP) [9]

$$-\frac{\partial}{\partial x}\left(\frac{1}{p(x,t)}\frac{\partial}{\partial x}\{p(x,t)\mathsf{K}(x,t)\}\right) = \frac{1}{\sigma_W^2}h'(x) \tag{9}$$

With boundary conditions $\lim_{x\to\pm\infty} p(x,t)\mathsf{K}(x,t) = 0,$

2. PERFORMANCE ANALYSIS OF EXISTINGALGORITHMS

2.1. Speech enhancement algorithm using FPF [1][8]

The study has been done on a microphone recorded speech signal sampled at 1 KHz, corrupted with white Gaussian noise (SNR ranging from 1-10dB) [1]. The graph shows the estimated output of the feedback particle filter results are simulated over 10dB input noise SNR, with no. of particles taken 100. The output MSE for graph shown was measured to be 4.43 10^{-4} [1]. Speech enhancement improves the quality and intelligibility of speech, ranging from removal of echo to suppression of background [1].

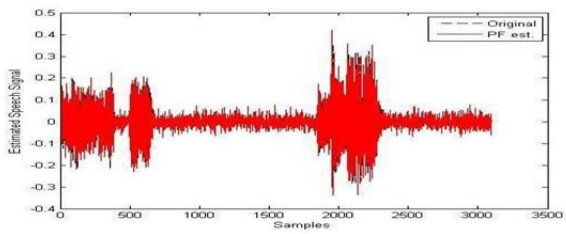

Fig. 2: Samples v/s Estimated Plot, original signal (black), Estimated signal (red) [1]

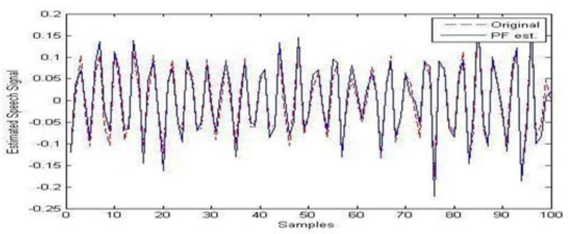

Fig. 3: Samples v/s Estimated Plot, original signal (red), Estimated signal (blue) [1]

The trails have been performed over different SNR ranges varied from 1dB-10dB, keeping no. of particles constant at 100.The table shows the variations of mean square errors over input SNR variations which shows not much variations hence shows the stability of algorithm at lower and higher SNR values. The results are tabulated below

TABLE I. The variation in mean squared error with change in input SNR [1]

S.No.	Variation in MSE with SNR	
	SNR in dB	MSE
1	1	0.0046
2	2	0.0030
3	3	0.0027
4	4	0.0019
5	5	0.0016
6	6	0.0012
7	7	9.44×10^{-4}
8	8	7.66×10^{-4}
9	9	6.18×10^{-4}
10	10	4.43×10^{-4}

Lastly here the comparison of particles in a particle filter in [8] gives us by the tables below

TABLE II. Variation of the output SNR versus the number of particles (Input SNR = 5db) [8]

Particles number	10	100	250	400	500	900	1000	1500
output $SNR(dB)$ voiced sequence	5.2	6.5	8.1	11.2	12.3	15.6	25.1	25.1
output $SNR(dB)$ unvoiced sequence	5.7	7.9	13.2	19.7	26.4	29.8	30.3	30.3

TABLE III. Variation of the output SNR versus the number of particles (Input SNR = 10db) [8]

Particles number	10	100	250	400	500	800	1000	1500
output SNR (dB) voiced sequence	5.1	5.9	7.7	10.1	11.9	14.7	22.1	22.1
output SNR (dB) unvoiced sequence	5.5	6.9	12.8	18.9	24.4	27.8	28.2	28.2

2.2. Proper orthogonal decomposition algorithm (FPF with POD based gain computation) [2]

We assess the performance of Algorithm using a planar two-body problem, which involves estimating the motion of a satellite that orbits around earth, and compare against a Rao-Blackwellized particle filter (RBPF) [2].

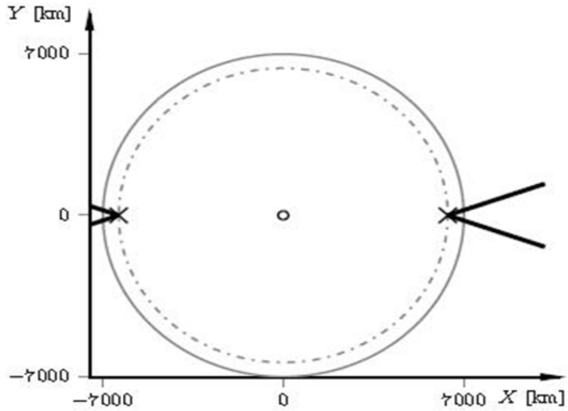

Fig. 4: The two-body problem with two bearing sensors (crosses) that measure the respective angle to the satellite from the earth-fixed X-axis. The earth surface is indicated with the dash-dotted circle and the small solid circle indicates the earth center. The true satellite path for one orbit realization is the gray circle [2].

We proposed a data-driven approach for choosing basis functions that approximate the gain function present in the FPF, which is the main difficulty when implementing the FPF. This is the first algorithm that proposes a general method for how to choose the basis functions. The key idea is that the evolution of the particle cloud gives information about how to locally adjust the particles. Because the method is data driven, it is applicable to a range of estimation problems. This is resampling free technique [2].

2.3. Optimal transport formulation of FPF [3]

The objective of this algorithm is to design a unique optimal SDE, whose solution denoted as S_t has conditional probability distribution equal to the conditional probability distribution of X_t[3].

$$P(S_t|\mathscr{Z}_t) = P(X_t|\mathscr{Z}_t), \quad \text{for} \quad t \in [0, T] \tag{10}$$

In the first figure (Monte-Carlo SDE) [3], a system of N particles are simulated independently according to,

$$\mathrm{d}S_t^i = \mathrm{d}B_t^i, \quad \text{for} \quad i = 1, \ldots, N, \tag{11}$$

Where $\{B_t\}$ are independent Wiener processes, and S^i_0 are samples drawn from initial distribution ***N*** (0, 1).

In the second figure (optimal transport SDE) [3], a system of N particles are simulated according to the optimal transport SDE

$$\mathrm{d}S_t^i = \frac{1}{2\hat{\Sigma}_t^{(N)}} \left(S_t - \hat{S}^{(N)}\right) \mathrm{d}t, \tag{12}$$

For i=1,……,N, where $S^i_0$are i.i.d samples drawn from initial distribution ***N***(0, 1), and

$$\hat{S}_t^{(N)} := \frac{1}{N}\sum_{i=1}^{N} S_t^i, \quad \hat{\Sigma}_t^{(N)} := \frac{1}{N}\sum_{i=1}^{N} (S_t^i - \hat{S}_t^{(N)})^2 \tag{13}$$

Optimal transport formulation of FPF algorithm served to decrease the simulation variance. The formulation is general and the extension to the non-linear and non-Gaussian setting is the subject of future work [3].

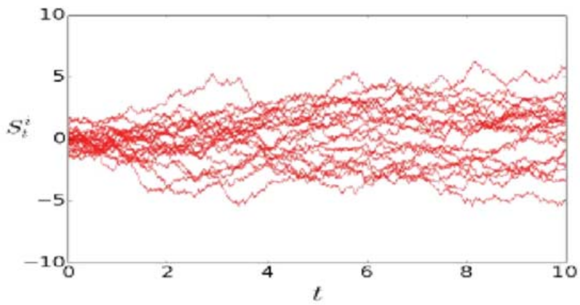

Fig. 5: Particle trajectories obtained by Monte-Carlo SDE [3]

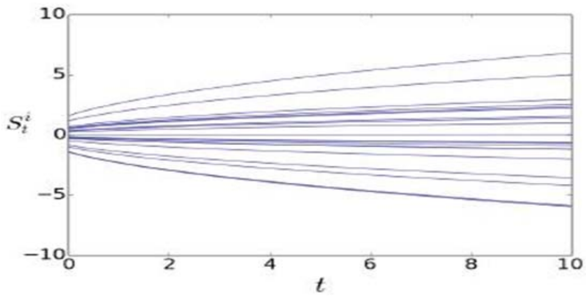

Fig. 6: Particle trajectories obtained by optimal transport SDE [3]

2.4. Implement Bayes' rule using a continuous transition between prior & posterior [4][7]

The numerical study contains two examples. One performance Measure, we use is the root-mean-square error (RMSE). Let $\hat{x}_{k,j}$ Denote the estimated mean at time t_k for the j^{th} of M Monte-Carlo simulations. Then the RMSE is computed as [4]

$$\mathrm{RMSE} = \sqrt{\frac{1}{M}\sum_{j=1}^{M}(x_{k,j} - \hat{x}_{k,j})^2}. \quad (14)$$

The RMSE is not necessarily the best performance measure for nonlinear, non-Gaussian systems, but is used because it is the standard method for comparing estimation performance.

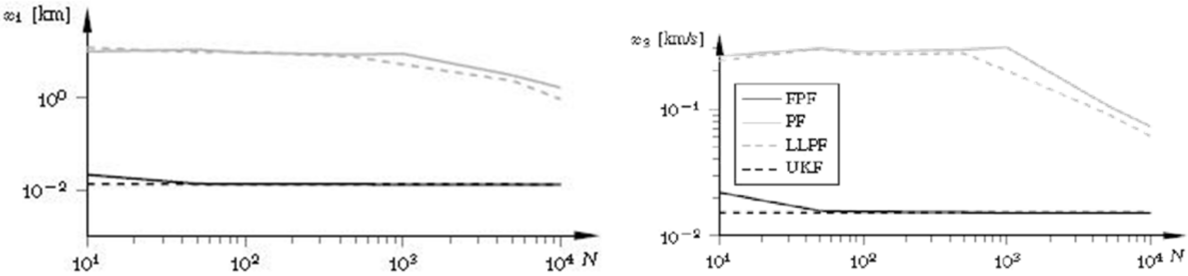

Fig.7: Time-averaged RMSEs for x_1, x_3, and x_5 as function of the number of particles for the reentry problem. The differences are small, but FPF has slightly smaller RMSE for both x1 and x3 for N ≥ 100. The RMSE values are computed for N = 10, 50, 100, 500, 1000, 5000, 10000 [4].

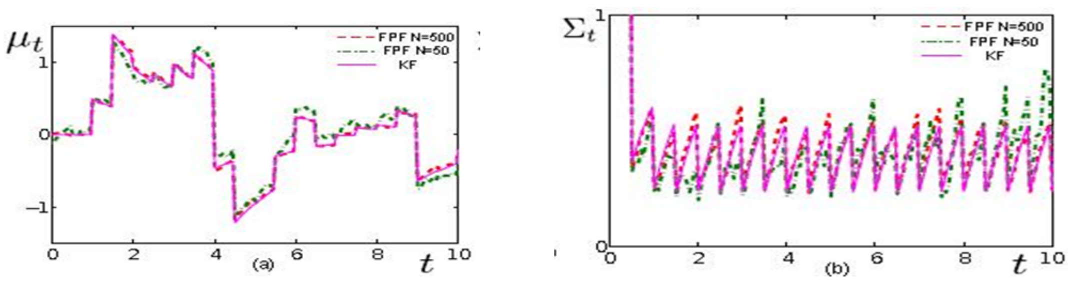

Fig.8: Comparison of the Kalman filter and the feedback particle filter with different number of particles: (a) conditional mean; (b) conditional variance [7].

2.5. Tractable algorithm for non-linear filtering and FPF algorithm for approximate the won ham filter [5]

The resulting algorithm represents an extension of FPF algorithm for diffusion process to a continuous-time Markov chain. And this model is useful in certain application- neuromorphic implementation for Bayesian inference in human brain [5].

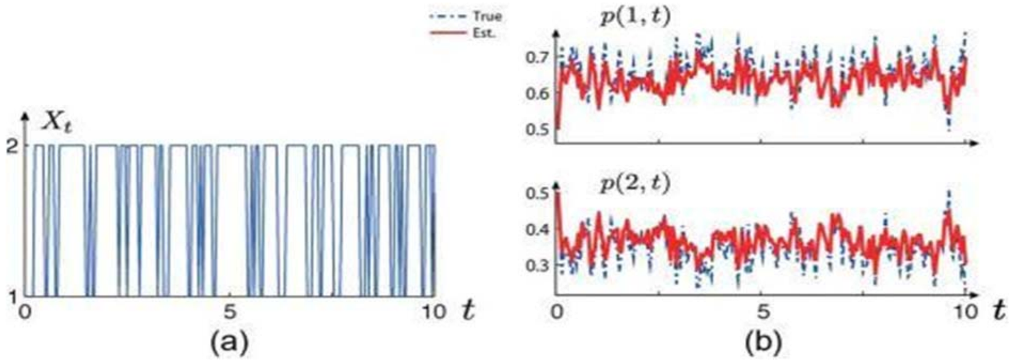

Fig.9: Simulation results for Example (a) Sample path *Xt*. (b) Comparison of *p*(·, *t*) obtained using the feedback particle filter (Est.) and the Won ham filter (True) [5].

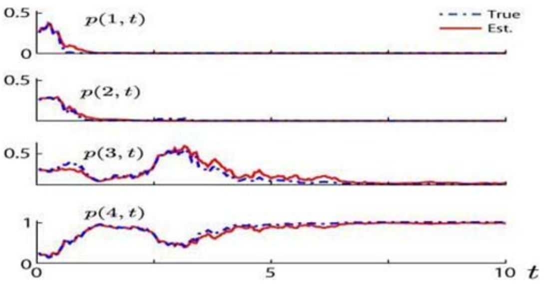

Fig.10: Simulation results for Comparison of *p*(·,*t*) obtained using the feedback particle filter (Est.) and the Won ham filter (True) [5].

2.6 Feedback Particle Filter on Matrix Lie Group [6]

2.6.1 FPF on SO(2) :- SO(2) is a 1-dimensional Lie group of rotation matrices R such that $RR^T = I$ and det(R) = 1. An arbitrary element is expressed as [6],

$$R = R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{15}$$

Where $\theta \in S^1$ is defined as the phase coordinate. And E is a basis of the Lie algebra SO(2),

$$E = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{16}$$

Finally, the gain function is obtained as,

$$\mathrm{k}(\theta) = \kappa_1 \cos(\theta) - \kappa_2 \sin(\theta) \tag{17}$$

2.6.2 FPF on SO(3) :- The action of the basis E1, E2, E3 is easily given as [6]

TABLE IV. Action of E_n on Basis function [6]

	$E_1\cdot$	$E_2\cdot$	$E_3\cdot$
ψ_1	$(R_{22}+R_{33})/2$	$-R_{21}/2$	$-R_{31}/2$
ψ_2	$-R_{12}/2$	$(R_{11}+R_{33})/2$	$-R_{32}/2$
ψ_3	$-R_{13}/2$	$-R_{23}/2$	$(R_{11}+R_{22})/2$
ψ_4	$(R_{23}-R_{32})/2$	$(R_{31}-R_{13})/2$	$(R_{12}-R_{21})/2$

The four basis functions are,

$$\psi_1(q) = 2q_1q_0, \quad \psi_2(q) = 2q_1q_0,$$
$$\psi_3(q) = 2q_1q_0, \quad \psi_4(q) = 2q_0^2 - 1 \tag{18}$$

In order to compute the matrix A and the vector b, the formulae for the action of E1, E2, E3 on these basis functions are [6]

TABLE V. Action of En on Basis functions using Quaternion [6]

	$E_1\cdot$	$E_2\cdot$	$E_3\cdot$
ψ_1	$q_0^2 - q_1^2$	$-q_1q_2 - q_3q_0$	$-q_1q_3 + q_2q_0$
ψ_2	$-q_1q_2 + q_3q_0$	$q_0^2 - q_2^2$	$-q_2q_3 - q_1q_0$
ψ_3	$-q_1q_3 - q_2q_0$	$-q_2q_3 + q_1q_0$	$q_0^2 - q_3^2$
ψ_4	$-2q_1q_0$	$-2q_2q_0$	$-2q_3q_0$

Finally, the filter in the quaternion coordinates has the following form,

$$dq_t^i = \frac{1}{2}\Lambda\big(V(q_t^i)\big)q_t^i + \frac{1}{2}\Lambda\big(\mathsf{K}(q_t^i,t)\big)q_t^i \circ \Big(\mathrm{d}Z_t - \frac{h(q_t^i)+\hat{h}_t}{2}\,\mathrm{d}t\Big), \tag{19}$$

Where $\mathsf{K}(q_t^i,t),\ V(q_t^i) \in so(3),\ V(q_t^i) = \Omega\,\mathrm{d}t + V_1 \circ \mathrm{d}B_t^i$ and the 4 x 4 matrix $\Lambda(\mathsf{K})$ is given by,

$$\Lambda(\mathsf{K}) := \begin{bmatrix} 0 & -k_1 & -k_2 & -k_3 \\ k_1 & 0 & k_3 & -k_2 \\ k_2 & -k_3 & 0 & k_1 \\ k_3 & k_2 & -k_1 & 0 \end{bmatrix} \tag{20}$$

3. COMPARISON TABLE:

Table VI- Comparison of FPF based algorithm and application

S. No.	Reference Paper	Algorithm Used	Measurement Parameters	Advantage	Limitation
1.	[1]	Noise Cancellation By Noise Removal Methodology Using FPF	MSE(Speech Data 0.000443) (Song Data0.00034) At 10dB SNR At Input & Keeping No. of Particles Constant	Improving Over All Quality of Speech Signal, Improving Intelligibility of Speech Ranging, Removal Of Background Noise By Removal Of Echo Phase Recovery Can Be Done	It Is Applicable Up To 100 No. Of Particles
2.	[2]	Proper Orthogonal Decomposition (FPF With POD Based Gain Computation)	Posterior Filter Density $P(x/y_k)$, Control Gain (K_k^i)	This Is Resampling Free Technique, Estimating The Motion Of Satellite, That Orbit Around The Earth	The Difficulty, When Choosing The Basis Function That Approximate The Gain Function
3.	[3]	Optimal Transport Formulation of Linear FPF	Estimating Mean $E(X_t) \simeq S_t^{\wedge(N)}$, Simulation Variance $(S_t^{\wedge(N)})$,	Decrease The Simulation Variance	Replacement of Noise Term With A Deterministic.

4.	[4]	Implement Bayes' Rule Using A Continuous Transition Between Prior &Posterior	Sensitivity to the Choice of Gain, Root mean Square Error (RMSE), Aerodynamic Parameter (x_5)	Decrease The Tracking Error With More Than One Magnitude Improving Quality Of Particle Distribution	
5.	[5]	Tractable Algorithm For Non-Linear Filtering, FPF Algorithm For Approximation Of The Won ham Filter, Generate A Poisson Counter With Time-dependent Rate, FPF For A Continuous Time Markov Chain, Synthesis Of The Gain Function..	Simulate The Particle Filter For2 & 4-State Markov Chain Respectively $X_t € S = \{e_1, e_2\}$ N_t^{12} – Poisson Counter X_t € S = {e1,e2,e3,e4} With Dynamic dx = 0	Neuromorphic Implementations For Bayesian Inference In Human Brain, Maneuvering Target Tracking	
6.	[6]	FPF Algorithm As A Solution For Continuous-Time Non-Linear Filtering of Stochastic Processes For Matrix Lie Groups, algorithm For Feedback Structure Of Kalman-Filter To General Non-Linear Non-Gaussian Filtering	Special Orthogonal Group So(2), So(3), Non-Linear filter Problem On Lie-groups Problem Statement, Filtering Equation, FPF On Lie-Group Particle Dynamic &Control Architecture	Altitude Estimation, Visual Tracking, Robotic Localization, Main-Fold & Preserve The Error Correction Based Feedback Structure of The Original FPF	
7.	[7]	FPF Algorithm For Continuous-Discrete Time Non-Linear Filtering (Particle Flow Algorithm)	Solution Of An Euler-Lagrange Boundary Value Parameter (E-L BVP), Evolution Of Posterior $P^*(x,t_n)$, General Equation $dS_n^i(\lambda)/d\lambda$	Compute The Condition Probability Density Function.	With Fewer Particles (N=50), The FPF Performance Degrades.
8.	[8]	Comparison Between Various Particle Filter Algorithm	Measurement Parameters of Different Algorithm of PF	Given Quick Idea About Particle Filter	

4. CONCLUSION

In this work, Feedback particle filter and their needs in various applications is critically reviewed. This review paper compares all the existing algorithms in comparison table. Here we studied that the transformations of feedback particle filter from continuous time filtering problem to discrete time filtering problem. To get better results, this paper may be used on Matrix lie group. This transformation is more efficient not only in terms of SNR or MSE, but also in terms of preserving the error correction based feedback structure of the original feedback particle filter that improves the conditional probability density function, with increase particles, which simultaneously improves the feedback particle filter performance.

Feedback particle filter and its combinations with other algorithms may be applicable in visual tracking, robotic localization and altitude estimation with greater accuracy. And may be used in image and video quality enhancement.

REFERENCES

[1] Dr. Agya Mishra & Rashi Singh, "Speech Enhancement Using Feedback Particle Filter," IEEE transactions on International conference on computational Intelligence & Communication Networks CICN., oct.2015, pp. 682-685.
[2] Karl Berntorp & Piyush Grover, "Data-Driven Gain Computation in the Feedback Particle Filter," On American Control Conference (ACC), Boston, MA, USA, July 6-8, 2016, pp. 2711-2716.
[3] Amirhossein Taghvaei & Prashant G. Mehta, "An Optimal Transport Formulation of the Linear Feedback Particle Filter," On American Control Conference (ACC), Boston, MA, USA, July 6-8, 2016, pp. 3614-3619.
[4] Karl Berntrop, "Feedback Particle Filter: Application and Evaluation," on International Conference on Information Fusion, Washington, DC- July 6-9, 2015, pp. 1633-1640.
[5] Tao Yang, Prashant G. Mehta & Sean P. Meyn, "Feedback Particle Filter for a Continuous-Time Markov Chain," On IEEE transactions on automatic control, vol. 61, No. 2, Feb. 2016, pp. 556-561.
[6] Chi Zhang, Amirhossein Taghvaei & Prashant G. Mehta, "Feedback Particle Filter On Matrix Lie Groups," On American Control Conference (ACC), Boston, MA, USA, July 6-8, 2016, pp. 2723-2728.
[7] Tao Yang, Henk A.P. Blom & Prashant G. Mehta, "The Continuous-Discrete Time Feedback Particle Filter," On American Control Conference (ACC), Portland, Oregon, USA, June 4-6, 2014, pp. 648-653.
[8] Agya Mishra & Rashi Singh, "Speech Enhancement Using Feedback Particle Filter: A Critical Review," on Indian Journal of Computer Science and Engineering (IJCSE), vol. 6, No. 3, Jun-Jul 2015, pp. 83-90.
[9] Tao Yang, Prashant G. Mehta & Sean P. Meyn, "Feedback Particle Filter" On IEEE transactions on automatic control, vol. 58, No. 10, Oct. 2013, pp. 2465-2480.