Proceedings of the 3rd International Conference on Control, Dynamic Systems, and Robotics (CDSR'16)
Ottawa, Canada – May 9 – 10, 2016
Paper No. 103
DOI: 10.11159/cdsr16.103

State Estimation for General Class of Dynamical Systems: An Extension to Particle Filters

Houman Hanachi, Jie Liu
Carleton University
1125 Colonel By Drive, Ottawa, Canada
houman.hanachi@carleton.ca; jie.liu@carleton.ca

Abstract – Many physical systems are nonlinear and non-Gaussian in their state-space models. Particle Filter (PF) is a sequential Monte Carlo method that uses sets of sample scenarios, i.e. "particles" to represent probability densities, and it can be applied for state estimation in nonlinear/non-Gaussian state-spaces models. Conventional variants of PF do not assume any noise for the system input, while the corresponding measurement models disregard the system input as an argument. In reality, physical systems receive inputs contaminated with the measurement noise. In this work, a generalized particle filter algorithm is developed that handles the noisy input of the state-space model in a probabilistic framework. Three advanced variants of PF are then developed to improve the filtering accuracy. Performance of the developed filters are then verified with simulation of univariate and bivariate non-stationary growth models as benchmarks.

Keywords**:** Dynamic system, Particle Filter, State estimation, Noisy input, Non-stationary growth model

1. Introduction

In many engineering problems, internal state of a system is to be estimated using sequential measurements on time-varying inputs and outputs of the system. State-space model of a dynamical system includes at least two parts: a system model that describes the state evolution with time, and a measurement model that shows the relation of the state and the measurements. Sequential Bayesian estimation is a rigorous approach for state estimation in dynamical systems. The optimal algorithm in closed form is called Kalman Filter (KF) [1], which applies only for first order Markovian systems with linear/Gaussian state-space models. Extended Kalman Filter (EKF) is a variant of KF that projects its applications to nonlinear systems [2]. With availability of low cost computational power, Monte Carlo methods have been applied to simulate the state probability distribution using weighted sample scenarios, called particles [3]. The so called Particle Filter (PF) is not restricted to linear/Gaussian systems and therefore, its variants have found a wide range of applications in various fields of science and engineering [4-8]. A dynamical system that receives time-varying input has the following state-space model:

$$\dot{x} = F(x, u) + \tau \tag{1}$$
$$y = G(x, u) + \nu \tag{2}$$

Where x is the state, u is the input and y is the output of the system. τ is the process noise and ν is the measurement noise. In discrete form, for a first order Markov process, state-space model at a time step k can be presented as:

$$x_k = F(x_{k-1}, u_k) + \tau_k \tag{3}$$
$$y_k = G(x_k, u_k) + \nu_k \tag{4}$$

Generic PF overlooks the role of input u_k in the system model [3]. Inclusion of the input into the system model has just been attended in a few recent works. However, in these works, the inputs of the systems have still been treated as deterministic variables [9,10]. In the real systems as illustrated in Fig. 1, the inputs are noisy, and there is a need for the development of PF framework such that it can manage general class of dynamical systems including noisy inputs. In a recent research work, the authors presented a PF based framework to address the stated problems [11]. In this paper, new variants of PF will be developed to manage the stochastic nature of the system input. To verify performance of the developed framework, a comparative study will be presented using one-dimensional and two-dimensional generalized non-stationary growth model. A conclusive summary of the paper will be provided in the last section.

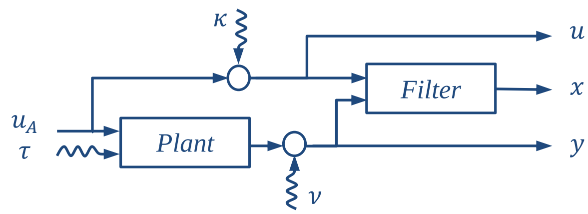

Fig. 1: Configuration of state estimation filter.

2. Sequential Bayesian State Estimation

In a dynamical system with discrete state-space model, system model F estimates the state x_k with an uncertainty of $f_x(x_k|x_{k-1}, u_k)$. At the same time, measurement model G estimates the output y_k with a marginal distribution of $f_y(y_k|x_k, u_k)$. We assume that uncertainty of F and marginal distribution of G are available. We also assume known densities for the initial state $f_x(x_0)$ and the system input $f_u(u_k)$. If the historical data on the input $Y_k \triangleq \{y_1, y_2, \dots, y_k\}$ and the output $U_k \triangleq \{u_1, u_2, \dots, u_k\}$ become available, the marginal filtering density can be found using Bayes' theorem:

$$f_x(x_k|Y_k, U_k) = f_{y,u}(y_k, u_k|x_k) f_x(x_k|Y_{k-1}, U_{k-1}) / f_{y,u}(y_k, u_k|Y_{k-1}, U_{k-1}) \tag{5}$$

Where,

$$f_x(x_k|Y_{k-1}, U_{k-1}) = \int f_x(x_k|x_{k-1})\, f_x(x_{k-1}|Y_{k-1}, U_{k-1})\, dx_{k-1} \tag{6}$$

$$f_{y,u}(y_k, u_k|x_k) = f_y(y_k|x_k, u_k)/f_x(x_k) \int f_x(x_k|x_{k-1}, u_k)\, f_x(x_{k-1})\, dx_{k-1} \tag{7}$$

and

$$f_{y,u}(y_k, u_k|Y_{k-1}, U_{k-1}) = \int f_{y,u}(y_k, u_k|x_k)\, f_x(x_k|Y_{k-1}, U_{k-1})\, dx_k \tag{8}$$

The system output u_k is independent from the previous state of the system, therefore, the probability transition density in Eq. 6 reduces to,

$$f_x(x_k|x_{k-1}) = \int f_x(x_k|x_{k-1}, u_k)\, f_u(u_k)\, \mathrm{d}u_k \tag{9}$$

The state probability $f_x(x_k)$ in Eq. 7 can be found sequentially by integrating the transition density of Eq. 9,

$$f_x(x_k) = \int f_x(x_k|x_{k-1})\, f_x(x_{k-1})\, \mathrm{d}x_{k-1} \tag{10}$$

In this way, the posterior filtering density at time step k is calculated by Eq. 5.

3. State Estimation by PF

Instead of a continuous mathematical function, the posterior density in Eq. 5 can be numerically represented by a set of weighted scenarios, known as particles.

$$f_x(x_k|Y_k, U_k) \approx \sum_{i=1}^{m} \omega_k^i \, \delta\left(x_k - x_k^i\right) \tag{11}$$

Where $x_k^1, \dots, x_k^m$ and $\omega_k^1, \dots, \omega_k^m$ are the particles and the corresponding weights respectively, such that, $\sum_{i=1}^{m} \omega_k^i = 1$. To find the weights ω_k^i in Eq.(11), when new measurements on the input u_k and the output y_k become available, the weights can be calculated using the values from last time step.

$$\omega_k^i \propto \omega_{k-1}^i f_{y,u}\left(y_k, u_k \middle| x_k^i\right) f_x\left(x_k^i \middle| x_{k-1}^i\right) / g_x\left(x_k^i \middle| x_{k-1}^i, y_k, u_k\right) \tag{12}$$

Where $g_x\left(x_k \middle| x_{k-1}^i, y_k, u_k\right)$ is the importance density with an optimal value to minimize the variance of the weights, that reduces Eq. 12 to:

$$\omega_k^i = \omega_{k-1}^i \int f_{y,u}(y_k, u_k|x_k) \, f_x\left(x_k \middle| x_{k-1}^i\right) \mathrm{d}x_k \tag{13}$$

The optimal result in Eq. (13) cannot be found, because the posterior is not yet available. That is why the choice of importance density g_x remains a filter design decision that has no unique global answer, and it is addressed differently in several variants of PF [12]. In the following, four variants of PF are developed with the ability to receive stochastic input.

3.1. Generic Particle Filter

A simple candidate for the importance density is the prior:

$$g_x\left(x_k \middle| x_{k-1}^i, y_k, u_k\right) = f_x\left(x_k \middle| x_{k-1}^i\right) \tag{14}$$

that reduces Eq. 12 to

$$\omega_k^i \propto \omega_{k-1}^i f_{y,u}\left(y_k, u_k \middle| x_k^i\right) \tag{15}$$

The particles x_k^i are propagated before the measurements on the input u_k are available, and they are consequently independent at each time step, i.e. $f_u\left(u_k \middle| x_k^i\right) = f_u(u_k)$. We can therefore rewrite Eq. 15 as,

$$\omega_k^i \propto \omega_{k-1}^i f_y\left(y_k \middle| x_k^i, u_k\right) \tag{16}$$

The propagated particles x_k^i and their corresponding weights ω_k^i represent the posterior density as laid out in Eq. (11). A common problem with the generic PF is that after some iteration, one particle holds the entire share of the weights and the other particles retain no weight. This so-called degeneracy can be avoided by resampling a set of new particles from the posterior density, in case the effective sample size N_{eff} falls below a predefined threshold N_T [13]. Table 1 presents the generic PF process for state estimation.

Table 1: PF process.

$\{(x_k^i, \omega_k^i)\}_{i=1:m} = \mathrm{PF}[\{(x_{k-1}^i, \omega_{k-1}^i)\}_{i=1:m}, y_k, u_k]$	
• Draw input samples:	$u_k^i \sim f_u(u_k)$
• Propagate priors:	$x_k^i \sim f_x\left(x_k \middle| x_{k-1}^i, u_k^i\right)$
• Update the weights:	$\omega_k^{i\prime} = \omega_{k-1}^i f_y\left(y_k \middle| x_k^i, u_k^i\right)$
• Normalize the weights:	$\omega_k^i = \omega_k^{i\prime} / \sum_{i=1}^m \omega_k^{i\prime}$
• Construct posterior density:	$f_x(x_k\vert y_k, u_k) \approx \sum_{i=1}^m \omega_k^i \delta\left(x_k - x_k^i\right)$
• Check effective sample size:	$N_{eff} \approx 1/\sum_{i=1}^m (\omega_k^i)^2$
• If $N_{eff} \leq N_T$, Resample:	$\{(x_k^i, \omega_k^i = 1/m)\}_{i=1:m} \sim f_x(x_k\vert y_k, u_k)$

3.2. Auxiliary Particle Filter (APF)

As stated earlier, the choice of importance density is crucial in a filter design. If the importance density only corresponds to the prior density, like generic PF, the propagated particles may land in the regions with low likelihoods where the corresponding weights get small values. To improve the filtering performance, we may give more chance to the particles, which will find more likelihood after propagation. For this objective, in auxiliary particle filter (APF), the particles are drawn from the joint distribution of the prior and the likelihood [5]. Therefore, the importance density is defined the joint density of the state x_k and the index of the particle j in the previous step, i.e. $g_{x,j}(x_k, j|Y_{k-1}, U_{k-1})$. From Bayes' rule the joint density of x_k and j is found as

$$\begin{aligned} f_{x,j}(x_k, j|Y_k, U_k) &\propto f_{y,u}(y_k, u_k|x_k) g_{x,j}(x_k, j|Y_{k-1}, U_{k-1}) \\ &= \omega_{k-1}^j f_{y,u}(y_k, u_k|x_k)\, f_x\left(x_k \middle| x_{k-1}^j\right) \end{aligned} \tag{17}$$

x_k is still unknown and it can be estimated by a representative value μ_k^j from the prior, which may be chosen the mean, the mode or a random draw from the prior. Therefore,

$$f_{x,j}(x_k, j|Y_k, U_k) \approx \omega_{k-1}^j f_{y,u}\left(y_k, u_k \middle| \mu_k^j\right) f_x\left(x_k \middle| x_{k-1}^j\right) \tag{18}$$

That reduces Eq. (12) to

$$\omega_k^i \propto f_{y,u}\left(y_k, u_k \middle| x_k^i\right) / f_{y,u}\left(y_k, u_k \middle| \mu_k^{j^i}\right) \tag{19}$$

And similar to Eq. (16)

$$\omega_k^i \propto f_y\left(y_k \middle| x_k^i, u_k\right) / f_y\left(y_k \middle| \mu_k^{j^i}, u_k\right) \tag{20}$$

The density function to draw candidate particles from the last step and propagate to the current step is therefore,

$$f_x(x_{k-1}|y_k, u_k) \approx \sum_{j=1}^m \omega_{k-1}^j f_y\left(y_k \middle| \mu_k^j, u_k\right) \delta\left(x_{k-1} - x_{k-1}^j\right) \tag{21}$$

Table 2 summarizes the process of APF for state estimation.

Table 2: APF process.

$\{(x_k^i, \omega_k^i)\}_{i=1:m} = APF[\{(x_{k-1}^i, \omega_{k-1}^i)\}_{i=1:m}, y_k, u_k]$	
• Draw input samples:	$u_k^i \sim f_u(u_k)$
• Generate representatives:	$\mu_k^i = E\left(x_k \middle| x_{k-1}^i, u_k^i\right)$
• Find representative weights:	$\omega_{\mu_k}^i \propto \omega_{k-1}^i f_y\left(y_k \middle| \mu_k^i, u_k^i\right)$
• Construct representative density:	$f_x(x_{k-1}\|y_k, u_k) \approx \sum_{i=1}^m \omega_{\mu_k}^i \delta\left(x_{k-1} - x_{k-1}^i\right)$
• Draw particles from last step:	$x_{k-1}^i \sim f_x(x_{k-1}\|y_k, u_k)$
• Redraw input samples:	$u_k^i \sim f_u(u_k)$
• Propagate priors:	$x_k^i \sim f_x\left(x_k \middle| x_{k-1}^i, u_k^i\right)$
• Regenerate representatives:	$\mu_k^i = E\left(x_k \middle| x_{k-1}^i, u_k^i\right)$
• Update the weights:	$\omega_k^i \propto f_y\left(y_k \middle| x_k^i, u_k^i\right) ./ f_y\left(y_k \middle| \mu_k^i, u_k^i\right)$
• Construct posterior density:	$f_x(x_k\|y_k, u_k) \approx \sum_{i=1}^m \omega_k^i \delta\left(x_k - x_k^i\right)$
• Check effective sample size:	$N_{eff} \approx 1/\sum_{i=1}^m (\omega_k^i)^2$
• If $N_{eff} \leq N_T$, Resample:	$\{(x_k^i, \omega_k^i = 1/m)\}_{i=1:m} \sim f_x(x_k\|y_k, u_k)$

3.3. Regularized Particle Filter (RPF)

When the particles are drawn from a discrete distribution function, the particles with smaller weights have lower chance, while the highly weighted particles will repeatedly be chosen, causing loss of diversity. This problem can be prevented with constructing a continuous distribution function for the posterior density. Regularization is usually done using rescaled kernel density $K_h(x) = h^{-D} K(x/h)$, where $h > 0$ is the bandwidth with an optimal value of h_{opt} for a D dimensional state of x [14]. Equation 11 can therefore be restated in the continuous form as

$$f_x(x_k|Y_k, U_k) \approx \sum_{i=1}^{m} \omega_k^i K_h\left(x_k - x_k^i\right) \tag{22}$$

There are different choices of kernel functions among which Epanechnikov and Gaussian functions are commonly used for regularization [15]. The process of RPF technique for state estimation is outlined in Table 3.

Table 3: RPF process.

$\{(x_k^i, \omega_k^i)\}_{i=1:m} = RPF[\{(x_{k-1}^i, \omega_{k-1}^i)\}_{i=1:m}, y_k, u_k]$	
• Draw input samples:	$u_k^i \sim f_u(u_k)$
• Propagate priors:	$x_k^i \sim f_x\left(x_k \middle| x_{k-1}^i, u_k^i\right)$
• Update the weights:	$\omega_k^{i\,\prime} = \omega_{k-1}^i f_y\left(y_k \middle| x_k^i, u_k^i\right)$
• Normalize the weights:	$\omega_k^i = \omega_k^{i\,\prime} / \sum_{i=1}^m \omega_k^{i\,\prime}$
• Regularize posterior density:	$f_x(x_k\|y_k, u_k) \approx \sum_{i=1}^m \omega_k^i K_h\left(x_k - x_k^i\right)$

3.4. Regularized Auxiliary Particle Filter (RAPF)

While APF improves the sampling process by drawing the particles of larger likelihoods in the next step, it still suffers from loss of diversity and sensitivity to the outliers in the proposal particles. If the process noise is small, sampling the particles with higher probability and propagating them to the next step will lead to congestion of the priors within a narrow neighbourhood, which does not represent the distribution of the state. This deficiency is addressed in RAPF technique by regularizing the empirical density $f_x\left(x_{k-1}^j \middle| y_k, u_k\right)$ and sampling from the resulting continuous distribution [16]. Similar to RPF, regularization of discrete density functions is doable with rescaled Kernel density:

$$f_x(x_{k-1}|y_k,u_k) \approx \sum_{j=1}^{m} \omega_{k-1}^{j}\, f_y\left(y_k\middle|\mu_k^j,u_k\right) K_h\left(x_{k-1}-x_{k-1}^{j}\right) \tag{22}$$

The likelihood of the representative particle $f_y\left(y_k\middle|\mu_k^j,u_k\right)$ may end up with small values, leading to large weights ω_k^i for the corresponding particles through Eq. (20). This will nullify the effect of the other particles with small weights, whereas they are equally consuming the computational efforts. To control the variance of the weights and improve effectiveness of the sampled particles, a rejection algorithm can be employed by defining an acceptance threshold for the samples, i.e., $> \omega_k^i > 1/w$, where $w > 1$ is a chosen design factor for rejection. RAPF technique with rejection algorithm is outlined in table 4.

4. Verification of Performance

To evaluate and compare performance of the developed filters, the well-known non-stationary growth model (NGM) [17] will be utilized. We have enhanced this model with non-uniform steps Δt and including the input u for performance evaluation of the systems with stochastic inputs [11,18].

$$\begin{gathered} u_k = u_{A_k} + \kappa_k \\ x_k = \frac{1}{2}x_{k-1}\Delta t_k + \frac{25\,(x_{k-1}+u_{A_k})\Delta t_k}{1+(x_{k-1}+u_{A_k})^2{\Delta t_k}^2} + 8\,\Delta t_k u_{A_k}\cos(1.2(k-1))/|u_{A_k}| + \tau_k \\ y_k = \frac{1}{20}x_k(x_k+u_{A_k}) + \nu_k \end{gathered} \tag{24}$$

We assume the input noise κ and the output noise ν have unknown distributions. Instead, the input noise and the output noise are represented by sets of s redundant measurements in each case. The model exemplifies a system with unknown noise characteristics for the sensors, where, there are several readings from the sensors at each time step.

Table 4: RAPF process.

$\{(x_k^i,\omega_k^i)\}_{i=1:m} = RAPF[\{(x_{k-1}^i,\omega_{k-1}^i)\}_{i=1:m},y_k,u_k]$	
• Draw input samples:	$u_k^i \sim f_u(u_k)$
• Generate representatives:	$\mu_k^i = E\left(x_k\middle|x_{k-1}^i,u_k^i\right)$
• Find representative weights:	$\omega_{\mu_k}^i \propto \omega_{k-1}^i f_y\left(y_k\middle|\mu_k^i,u_k^i\right)$
• Regularize representative density:	$f_x(x_{k-1}|y_k,u_k) \approx \sum_{i=1}^{m}\omega_{\mu_k}^i K_h\left(x_{k-1}-x_{k-1}^i\right)$
• Draw particles from last step:	$x_{k-1}^i \sim f_x(x_{k-1}|y_k,u_k)$
• Redraw input samples:	$u_k^i \sim f_u(u_k)$
• Propagate priors:	$x_k^i \sim f_x\left(x_k\middle|x_{k-1}^i,u_k^i\right)$
• Regenerate regularized representatives:	$\mu_k^i = E\left(x_k\middle|x_{k-1}^i,u_k^i\right)$
• Update the weights:	$\omega_k^i \propto f_y\left(y_k\middle|x_k^i,u_k^i\right)./f_y\left(y_k\middle|\mu_k^i,u_k^i\right)$
• If $\omega_k^i > w$ or $\omega_k^i < 1/w$, reject the proposal x_k^i, return to "Draw particles from last step".	
• Regularize posterior density:	$f_x(x_k|y_k,u_k) \approx \sum_{i=1}^{m}\omega_k^i K_h\left(x_k-x_k^i\right)$

4.1. Univariate Non-stationary Growth Model (UNGM)

We assume a univariate non-stationary growth model (UNGM) with a single input with actual value of $u_{A_k} = 8(sin(1.2\,k) + sin(k^2))$ and a single output with the following assumptions: $\Delta t_k \sim \mathcal{U}(0,2), \tau_k \sim \mathcal{N}(0,5)$ and random measurement samples generated from an arbitrary distribution such that $\{\nu_k^l|E\left(\nu_k^l\right)=0, Var\left(\nu_k^l\right)=2\}_{l=1:50}$ and $\{\kappa_k^l|E\left(\kappa_k^l\right)=0, Var\left(\kappa_k^l\right)=2\}_{l=1:50}$. Figure 2 shows the actual and the observed input, the actual state and the observed output of the system in 50 time-steps. State estimation has been repeated for 20 times by each of the developed filters using 100 particles. Figure 3 shows the estimation results for 10 iterations, and table 5 compares the root mean square error of state estimation for the filters. The largest error associates with generic PF and it is unable to track the system state at many time steps, as shown in Fig. 3. RAPF has a slightly better performance than APF for this model, while it is computationally more costly than APF. Among the developed techniques, RPF shows relatively the best performance with the least averaged error over the 50 time steps.

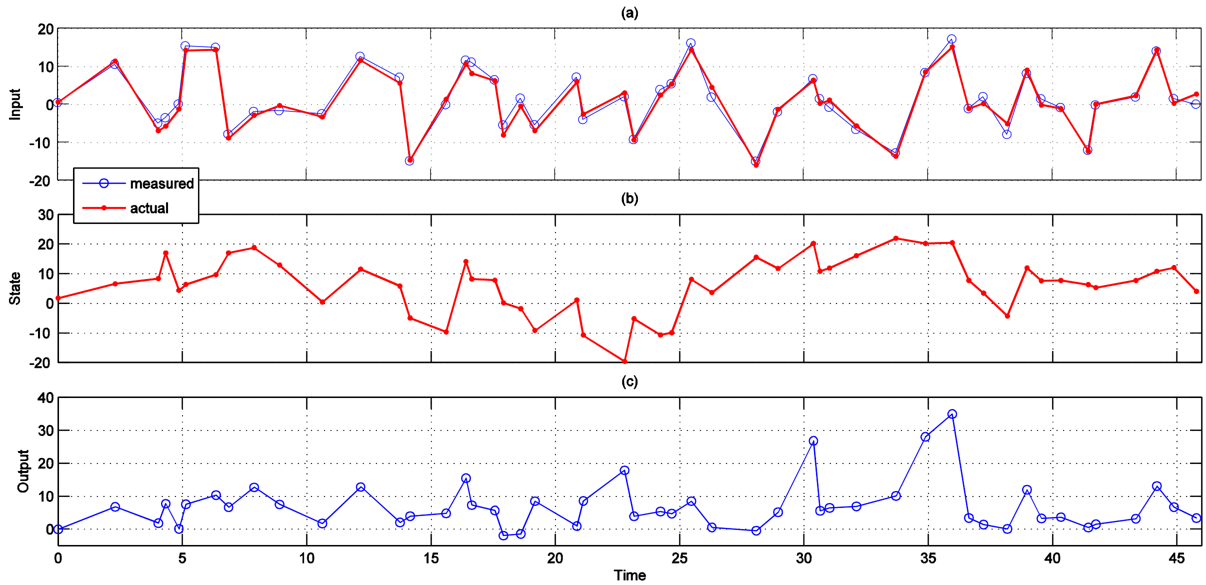

Fig. 2: One-dimensional state-space; (a) system input, (b) internal state, and (c) measured output.

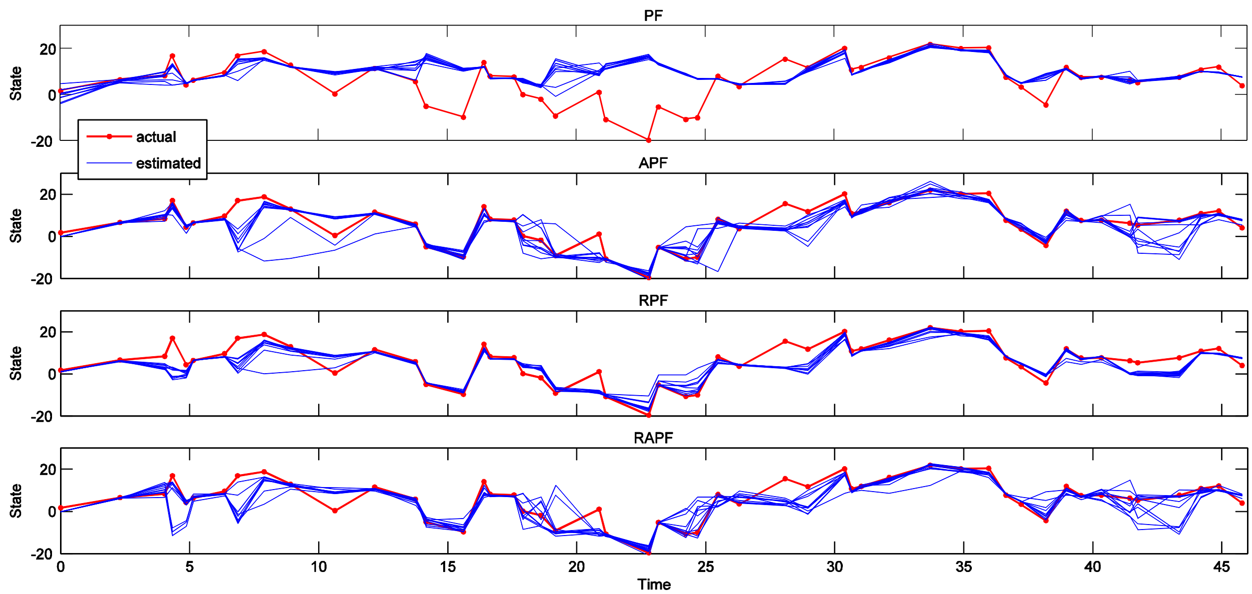

Fig. 3: State estimation results for PF, APF, RPF and RAPF.

Table 5: Root mean square error for state estimation for UNGM.

Technique	Error
PF	8.88
APF	7.05
RPF	5.38
RAPF	6.48

4.2. Bivariate Non-stationary Growth Model (BNGM)

To evaluate filtering performance in a higher dimensional state-space, we use complex numbers in the model in Eq. 24 that creates a bivariate system with dependent variables. In this simulation, the actual input value is assumed to be $u_{A_k} = 8(1+i)(sin(1.2\,k) - i\,sin(k^2))$ and for the output: $\tau_k = \tau_{k_R} + i\,\tau_{k_I}$, where $\tau_{k_R}, \tau_{k_I} \sim \mathcal{N}(0{,}5)$. Random measurement samples are generated from an arbitrary distribution such that $\{v_k^l = v_{k_R}^l + i\,v_{k_I}^l | E(v_k^l) = 0, Var(v_{k_R}^l) = Var(v_{k_I}^l) = 2\}_{l=1:50}$ and $\{\kappa_k^l = \kappa_{k_R}^l + i\,\kappa_{k_I}^l | E(\kappa_k^l) = 0, Var(\kappa_{k_R}^l) = Var(\kappa_{k_I}^l) = 2\}_{l=1:50}$. In Fig. 4 the actual and the observed input, the actual state and the observed output of the system are shown in 50 time-steps. Likewise the UNGM, state estimation has been repeated for 20 times by each of the developed filters with 100 particles. To compare estimation error of the filters, table 6 provides the root mean square errors over the entire estimated states. Unlike UNGM, generic PF shows a better performance in comparison with APF. Performance of RAPF filter is almost the same as that of PF, and similar to UNGM, the best accuracy is achieved by RPF.

Table 6: Root mean square error for state estimation for BNGM.

Technique	Error		
	Real part	Imag. Part	Total
PF	5.50	4.53	7.13
APF	6.01	5.64	8.24
RPF	4.74	3.71	6.02
RAPF	5.49	4.68	7.21

5. Summary and Conclusion

Monte Carlo method has been applied to sequential Bayesian state estimation framework for general class of dynamical systems with first order Markovian model. The resulting filter, i.e. PF can estimate the state considering the noisy input of the systems. In addition to the generic PF, three other variants of this filter, i.e. APF, RPF and RAPF were developed and presented. To verify the performance of the filters in a one-dimensional state-space, an extended form of the well-known NGM has been employed for simulating a highly nonlinear system behaviour. To verify the filtering performance with a higher dimension, the NGM was utilized such that the input, the state and the output variables were taken complex variables. Results of applying the developed filters on the NGM model shows that RPF provides the highest accuracy for the simulation model under study, for both one-dimensional and two-dimensional models. It should be noted that filtering performance is depended on the individual systems, so that a filter with superior performance in one system may show poor results on another system [19].

The work shows that PF framework can effectively be extended to more complex dynamical systems where the system receives noisy inputs.

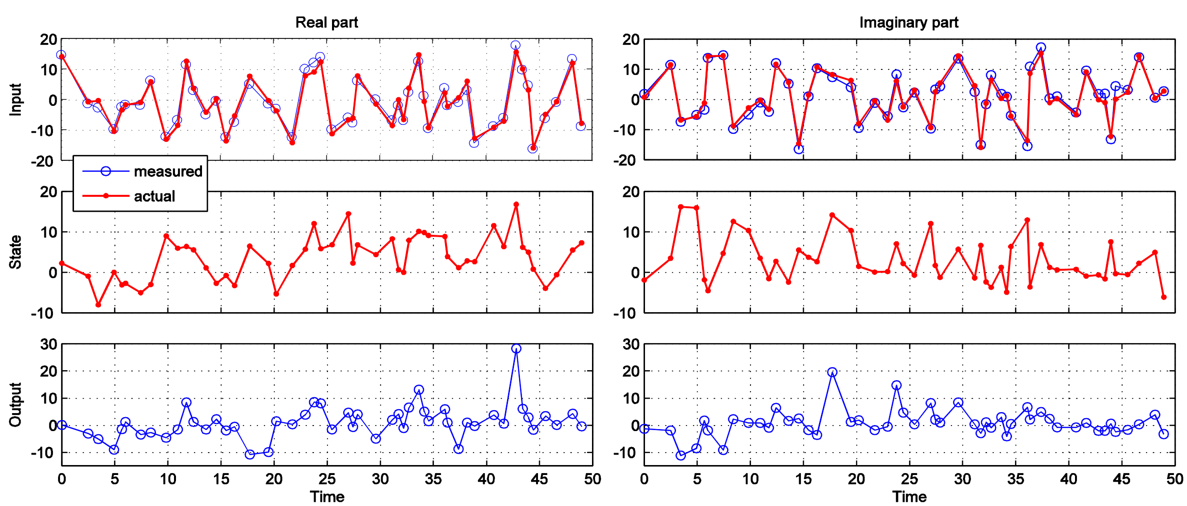

Fig. 4: Two-dimensional state-space; top: system input, middle: internal state, and bottom: measured output.

Acknowledgements

This project was financially supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, and Life Prediction Technologies Inc. (LPTi), Ottawa, Canada.

References

[1] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *J. Basic Eng.*, vol. 82, no. 1, p. 35, 1960.
[2] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *AeroSense '97*, pp. 182-193,1997.
[3] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc. F Radar Signal Process.*, vol. 140, no. 2, p. 107, 1993.
[4] A. Doucet, N. Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York, NY: Springer New York, 2001.
[5] M. K. Pitt and N. Shephard, "Filtering via Simulation: Auxiliary Particle Filters," *J. Am. Stat. Assoc.*, Feb. 2012.
[6] J. K. Kimotho, T. Meyer, and W. Sextro, "PEM fuel cell prognostics using particle filter with model parameter adaptation," in *2014 International Conference on Prognostics and Health Management*, 2014, pp. 1-6.
[7] R. Havangi, M. A. Nekoui, M. Teshnehlab, and H. D. Taghirad, "A SLAM based on auxiliary marginalised particle filter and differential evolution," *Int. J. Syst. Sci.*, vol. 45, no. 9, pp. 1913-1926, 2013.
[8] N. Daroogheh, N. Meskin, and K. Khorasani, "A novel particle filter parameter prediction scheme for failure prognosis," in *2014 American Control Conference*, pp. 1735-1742, 2014.
[9] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 53-82, 2010.
[10] S. P. Won, W. W. Melek, and F. Golnaraghi, "A Kalman/Particle Filter-Based Position and Orientation Estimation Method Using a Position Sensor/Inertial Measurement Unit Hybrid System," *IEEE Trans. Ind. Electron.*, vol. 57, no. 5, pp. 1787-1798, 2010.
[11] H. Hanachi, J. Liu, A. Banerjee, and Y. Chen, "Sequential state estimation of nonlinear/non-Gaussian systems with stochastic input for turbine degradation estimation," *Mech. Syst. Signal Process.*, 2015.
[12] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174-188, 2002.
[13] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved particle filter for nonlinear problems," *IEE Proc. - Radar, Sonar Navig.*, vol. 146, no. 1, pp. 2, 1999.
[14] N. Oudjane and C. Musso, "Progressive correction for regularized particle filters," in *Proceedings of the Third International Conference on Information Fusion*, 2000, vol. 2, pp. THB2/10-THB2/17.
[15] P. Van Kerm, "Adaptive kernel density estimation," *Stata J.*, 2003.
[16] J. Liu, W. Wang, and F. Ma, "A regularized auxiliary particle filtering approach for system state estimation and battery life prediction," *Smart Mater. Struct.*, vol. 20, no. 7, pp. 075021, 2011.
[17] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley & Sons, 2006.
[18] H. Hanachi, J. Liu, A. Banerjee, and Y. Chen, "A framework with nonlinear system model and nonparametric noise for gas turbine degradation state estimation," *Meas. Sci. Technol.*, vol. 26, no. 6, pp. 065604, 2015.
[19] L. B. Jackson, *Digital Filters and Signal Processing: With MATLAB® Exercises*. Springer Science & Business Media, 2013.