Praise Worthy Prize

International Review of Automatic Control (I.RE.A.CO.), Vol. 4, N. 3
May 2011

Enhanced Particle Filtering for Nonlinear State Estimation via Invasive Weed Optimization

Mohamadreza Ahmadi, Hamed Mojallali

Abstract – *In this paper, an enhanced version of the particle filtering (PF) technique using the invasive weed optimization (IWO) algorithm is proposed. Due to the fact that sampling from the posterior distribution in PF algorithm is a sub-optimal approach, it is vulnerable to estimation errors. To avoid such approximation errors, this paper suggests incorporating the IWO algorithm by translating the sampling step into a nonlinear optimization problem. By defining an appropriate fitness function, the optimization problem is readily handled. The functionality of the proposed method is evaluated against several examples through simulation analysis. It is demonstrated that applying the suggested IWO enhanced PF algorithm (PFIWO) would lead to superior estimation performance.*

Keywords*: Particle Filter, Nonlinear Filtering, State Estimation, Invasive Weed Optimization*

Nomenclature

x_k	The state vector with probability distribution of $p(x_k \mid x_{k-1})$ which is not directly measurable
y_k	The noise corrupted observation with likelihood $p(y_k \mid x_k)$
$\tilde{x}^i$	i th drawn sample from a probability distribution
F_i	The fitness of the i th weed or particle
k	The discrete time instance
N	The number of particles
n	The nonlinear modulation index in the IWO algorithm
N_{eff}	The effective sample size associated with the weights

I. Introduction

A large group of models in signal processing can be represented by a state-space form in which prior knowledge of the system is available.

This prior knowledge allows us to exploit a Bayesian approach. Within this statistical framework, one can perform inference on the unknown states according to the posterior distribution. In most cases, the observations arrive sequentially in time, and one is interested in recursively estimating the *hidden* states from the time-varying posterior distribution. This problem is referred as the *optimal filtering* problem [1],[2].

Owing to the mathematical complexity, only few specific models (including linear Gaussian state-space models and finite state-space hidden Markov models (HMM) [3]) can be adopted to reach an analytical solution. The popular Kalman filter (KF) [1] and the renowned HMM filter [3] provide close form solutions to the latter models.

In many real-life applications, however, the models possess nonlinearity and non-Gaussian behavior. Therefore, an optimal solution to the filtering problem cannot be achieved. Over the last decades, several sub-optimal filtering methods such as the extended Kalman filter (EKF), and the unscented Kalman filter (UKF) have been proposed in open literature [4]. But, these filtering algorithms suffer from the *curse of dimensionality*, that is, they perform poorly as the dimension of the model states increases. Furthermore, the rate of convergence of the approximation error decreases dramatically for large state dimensions, say 4 [5].

The particle filter (PF) first brought forward by Gordon et al. [5] utilizes a set of N random samples (or particles) to approximate the posterior distribution. The particles are evolved over time via a combination of importance sampling and resampling steps. In a few words, the resampling step statistically multiplies and/or discards particles at each time step to adaptively concentrate particles in the regions of high posterior probability [6].

Recently, there exists an active research on the subject of integrating meta-heuristic algorithms in PF. In [7], Tong *et al.* proposed an optimized PF based on particle swarm optimization (PSO) algorithm which exposed improved estimation accuracy. Many subsequent studies also followed the same trend using the PSO method; e.g., refer to [8],[9].

Manuscript received and revised April 2011, accepted May 2011

This paper considers the implementation of the IWO algorithm as a means to optimizing the PF method. Since sampling in PF is performed in a sub-optimal manner, it can bring about some performance defects such as *sample impoverishment* [4]. Using a suitable fitness function for particles, such problems are circumvented and an enhanced PF algorithm is achieved thanks to the IWO approach. The functionality of the combined method is verified using two nonlinear state estimation problems.

The rest of this paper is organized as follows. Section II considers a concise description of the filtering problem and the basic particle filtering algorithm. The IWO algorithm is outlined in section III. The proposed PFIWO method is discussed in section IV. Results based on the PFIWO algorithm are addressed in section V. Section VI concludes the paper.

II. The Particle Filter

II.1. The Filtering Problem

Consider the general class of nonlinear non-Gaussian systems with state-space model as described below:

$$x_k = f\left(x_{k-1}, u_{k-1}, v_{k-1}\right), \quad x_k \sim p\left(x_k \left| x_{k-1}\right.\right) \qquad (1)$$

$$y_k = g\left(x_k, u_k, w_k\right), \quad y_k \sim p\left(y_k \left| x_k\right.\right) \qquad (2)$$

where the subscript k denotes the time instance. $f(\cdot)$ and $g(\cdot)$ are generally nonlinear functions. x_k is the system state with probability distribution of $p\left(x_k \left| x_{k-1}\right.\right)$ which is not directly measurable, and y_k is the noise corrupted observation with likelihood $p\left(y_k \left| x_k\right.\right)$. This structure is illustrated in Fig. 1.

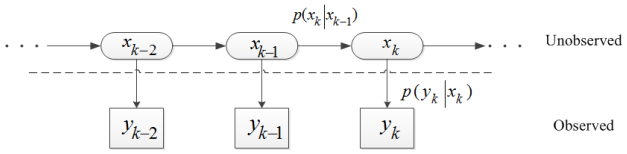


Fig. 1. A graphical representation of the state-space model described by Eq. (1)

u stands for known inputs. v and w represent the process and measurement noise, respectively. *Filtering* is the task of sequentially estimating the states (parameters or hidden variables) of a system as a set of observations become available on-line [1], [3]. In other words, filtering is aimed at estimating the posterior distribution $p\left(x_k \left| y_k\right.\right)$ *as* a set of observations $Y_k = \left(y_1, y_2, \ldots, y_k\right)^T$ becomes available.

The Bayesian solution to the filtering problem consists of two stages [1], [3], [4]:

1) Prediction: using the prior density function and the Chapman-Kolomogrov equation we have:

$$p\left(x_k \left| y_{k-1}\right.\right) = \int p\left(x_k \left| x_{k-1}\right.\right) p\left(x_{k-1} \left| y_{k-1}\right.\right) dx_{k-1} \qquad (3)$$

2) Correction: based on the Bayes' formula:

$$p\left(x_k \left| y_k\right.\right) = \frac{p\left(y_k \left| x_k\right.\right) p\left(x_k \left| y_{k-1}\right.\right)}{p\left(y_k \left| y_{k-1}\right.\right)} \qquad (4)$$

The algorithm is initialized with $p\left(x_0 \left| y_0\right.\right) = p\left(x_0\right)$ and $p\left(x_1 \left| y_0\right.\right) = p\left(x_1\right)$. One step operation of the Bayesian filtering is portrayed in Fig. 2. However, it is obvious that achieving a closed form analytical solution to the untraceable integral in Eq. (3) and therefore the solution to Eq. (4) is a cumbersome task. The problem becomes even more severe as the state dimensions increase. Thus, an optimal solution cannot be attained except under very restricting conditions (linear transition functions and Gaussian noise) using the well-known KF. The interested reader can refer to [1],[3] for more information on optimal solutions which provide a comprehensive theoretical overview of available methods. Sub-optimal solutions exist for rather general models with nonlinear evolution functions and non-Gaussian noises [4]. Nevertheless, due to the nature of these methods (e.g. EKF and UKF) which are based on local linearization, the estimation performance is, more or less, limited. Estimation techniques established upon sequential Monte Carlo methods, namely the PF, are a promising alternative to local linearization algorithms.

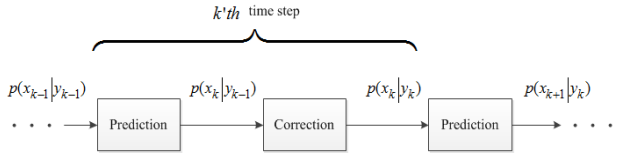


Fig. 2. The Bayesian approach to filtering problem

II.2. The Monte Carlo Method

In the Monte Carlo technique, one is concerned with estimating the properties of some highly complex probability distribution $p(x)$, e.g. *expectation*:

$$E\left(s(x)\right) = \int s(x)\, p(x)\, dx \qquad (5)$$

where $s(x)$ is some useful function for estimation. In cases where this cannot be obtained analytically, the approximation problem can be handled indirectly. It is possible to represent $p(x)$ by a set of random samples $\tilde{x}^i, i = 1,2,\ldots,N$. Consequently, the Monte Carlo representation is [10]:

$$p(x)=\frac{1}{N}\sum_{i=1}^{N}\delta\left(x-\tilde{x}^{i}\right) \tag{6}$$

where $\delta(\cdot)$ is the Dirac delta function. Then, the expectation can be reformulated as:

$$\begin{aligned} E\left(s(x)\right) &= \int s(x)\,p(x)\,dx \approx \\ &\approx \int s(x)\frac{1}{N}\sum_{i=1}^{N}\delta\left(x-\tilde{x}^{i}\right)dx \\ &= \frac{1}{N}\sum_{i=1}^{N}s\left(\tilde{x}^{i}\right) \end{aligned} \tag{7}$$

Alternatively, suppose that the samples $\tilde{x}^{i}$ are drawn from a distribution $q(x)$ instead of $p(x)$. Now, the expectation can be estimated using importance sampling as follows [10]:

$$\begin{aligned} E\left(s(x)\right) &= \int s(x)\,p(x)\,dx = \int s(x)\frac{q(x)\,p(x)}{q(x)}dx \\ &\approx \int s(x)\frac{p(x)}{q(x)}\frac{1}{N}\sum_{i=1}^{N}\delta\left(x-\tilde{x}^{i}\right)dx \\ &= \frac{1}{N}\sum_{i=1}^{N}s\left(\tilde{x}^{i}\right)\frac{p\left(\tilde{x}^{i}\right)}{q\left(\tilde{x}^{i}\right)} = \frac{1}{N}\sum_{i=1}^{N}w_i s\left(\tilde{x}^{i}\right) \end{aligned} \tag{8}$$

where $w_i \propto \dfrac{p\left(\tilde{x}^{i}\right)}{q\left(\tilde{x}^{i}\right)}$ is the importance weight. So, $p(x)$ can be estimated as:

$$p(x)=\sum_{i=1}^{N}w_i\delta\left(x-\tilde{x}^{i}\right),\quad s.t.\ \sum_{i=1}^{N}w_i=1 \tag{9}$$

II.3. The Basic Particle Filter

Consider equations (3) and (4). This discussion is proceeded by reformulating the latter equations based on the Monte Carlo approximation described in the previous section. Let $\tilde{x}^{i}$, $i=1,2,\ldots,N$ be the drawn samples from the posterior distribution $p\left(x_k\left|y_k\right.\right)$. The filter is initialized as [7]:

$$\tilde{x}_0^{i}\sim p\left(x_0\left|y_0\right.\right),\ \ i=1,2,\ldots,N \tag{10}$$

Then, for $k=1,2,\ldots$ we have:

$$p\left(x_k\left|y_k\right.\right)=\sum_{i=1}^{N}w_i^{k}\delta\left(x-\tilde{x}_k^{i}\right),\ s.t.\ \sum_{i=1}^{N}w_i^{k}=1 \tag{11}$$

For $i=1,2,\ldots,N$ sample from the proposal distribution $q\left(x_k\left|\tilde{x}_{k-1}^{i}\right.\right)$ as:

$$\tilde{x}_k^{i}\sim q\left(x_k\left|\tilde{x}_{k-1}^{i}\right.\right) \tag{12}$$

Subsequently, update the importance weights:

$$w_i^{k}=w_i^{k-1}\frac{p\left(y_k\left|\tilde{x}_{k-1}^{i}\right.\right)p\left(\tilde{x}_k^{i}\left|\tilde{x}_{k-1}^{i}\right.\right)}{q\left(y_k\left|\tilde{x}_{k-1}^{i}\right.\right)} \tag{13}$$

Provided that $p\left(x_k\left|x_{k-1}\right.\right)=q\left(x_k\left|x_{k-1}\right.\right)$, equations (12) and (13) converts to:

$$\tilde{x}_k^{i}\sim p\left(x_k\left|\tilde{x}_{k-1}^{i}\right.\right) \tag{14a}$$

$$w_i^{k}=w_i^{k-1}p\left(y_k\left|\tilde{x}_{k-1}^{i}\right.\right) \tag{14b}$$

Afterwards, for $i=1,2,\ldots,N$ normalize the weights:

$$w_i^{k}=\frac{w_i^{k}}{\sum_{j=1}^{N}w_j^{k}} \tag{15}$$

A prevalent problem with PF is the degeneracy phenomenon, wherein after few iterations, all but few particles will have trivial weights. A measure of degeneracy is the effective sample size N_{eff} which can be empirically evaluated as:

$$\hat{N}_{eff}=\frac{1}{\sum_{i=1}^{N}\left(w_i^{k}\right)^{2}} \tag{16}$$

The conventional approach to solve around the problem of sample degeneracy is to define a degeneracy threshold N_{th}. If $\hat{N}_{eff}<N_{th}$, *resampling* should be initiated [4].

III. The Invasive Weed Optimization

The bio-inspired IWO algorithm was introduced by Mehrabian and Lucas [11] which imitates the colonial behavior of invasive weeds in nature. The IWO algorithm has shown to be virtuous in converging to optimal solution by employing some basic characteristics of weed colonization, e.g. seeding, growth and competition [12].

III.1. Key Terms

Prior to describing the IWO algorithm, the key terms are explained as follows:

a. *Seed*: Each unit in the colony (here the particles) which encompasses a value for each variable in the optimization problem before fitness evaluation.
b. *Weed/Plant*: any seed that is evaluated grows to a weed or plant.
c. *Fitness*: a value corresponding to the goodness of each unit after being evaluated.
d. *Field*: the search/solution space.
e. *Maximum weed population*: a parameter preset representing the maximum number of possible weeds in the field after fitness assessment.

III.2. The IWO Algorithm

The process flow of the IWO algorithm is outlined below [11], [12]:

1. Initialize the seeds $S_i = (s_1, s_2, \ldots, s_n)^T$, where n is the number of selected variables, over the search space. Consequently, each seed contains random values for each variable in the $n-D$ solution space.
2. The fitness of each individual seed is calculated according to the optimization problem, and the seeds grow to weeds able to produce new units.
3. Each individual is ranked based on its fitness with respect to other weeds. Subsequently, each weed produces new seeds depending on its rank in the population. The number of seeds to be created by each weed alters linearly from $N_{\min}$ to $N_{\max}$ which can be computed using the equation given below:

$$\text{Number of seeds} = \frac{F_i - F_{worst}}{F_{best} - F_{worst}}\left(N_{max} - N_{min}\right) + N_{min} \tag{17}$$

in which F_i is the fitness of ith weed. F_{worst}, and F_{best} denote the best and the worst fitness in the weed population. This step ensures that each weed take part in the reproduction process.

4. The generated seeds are normally distributed over the field with zero mean and a varying standard deviation of σ_{iter} described by:

$$\sigma_{iter} = \left(\frac{iter_{max} - iter}{iter_{max}}\right)^n \left(\sigma_f - \sigma_0\right) + \sigma_f \tag{18}$$

where $iter_{max}$ and $iter$ are the maximum number of iteration cycles assigned by the user, and the current iteration number respectively. σ_0 and σ_f represent the pre-defined initial and final standard deviations. n is called the nonlinear modulation index. In order to obtain a full and swift scan of possible values of standard deviation, it has been examined that the most appropriate value for nonlinear modulation index is 3 [12]. The fitness of each seed is calculated along with their parents and the whole population is ranked. Those weeds with less fitness are eliminated through competition and only a number of weeds remain which are equal to *Maximum Weed Population.*

5. The procedure is repeated at step 2 until the maximum number of iterations allowed by the user is reached.

IV. The Proposed PFIWO Algorithm

In this section, we will first highlight some exclusive features of the IWO algorithm which convinced us to choose this meta-heuristic strategy as an optimizer for PF; afterwards, we will discuss the proposed PFIWO scheme.

The IWO algorithm certifies that all possible candidates would participate in the reproduction process. In contrast, most meta-heuristic algorithms would not allow the less-fitted individuals to produce offspring such as the GA. Besides, the IWO algorithm is straightforward and it includes less deal of computational burden unlike other methods. As a good illustration, one can consider the PSO algorithm. PSO needs to update both the position and velocity of individuals in each iteration round which require some extra calculations to find the best position in the neighborhood of each particle as well as the whole population. The mentioned incentives persuaded us to integrate the IWO algorithm in PF. The modifications are delineated next.

Owing to the fact that the sampling step of the conventional PF is sub-optimal, the IWO is suggested as a means to enhance the sampling step. Here, the goal of the IWO in the sampling step is to trace the particles which correspond to greater weights. Therefore, it is convenient to calculate the fitness of i'th particle as:

$$fitness^i = \frac{p\left(y_k \middle| \tilde{x}_{k-1}^i\right) p\left(\tilde{x}_k^i \middle| \tilde{x}_{k-1}^i\right)}{q\left(y_k \middle| \tilde{x}_{k-1}^i\right)} \tag{19}$$

which in case of $p\left(x_k \middle| x_{k-1}\right) = q\left(x_k \middle| x_{k-1}\right)$ reduces to:

$$fitness^i = p\left(y_k \middle| \tilde{x}_{k-1}^i\right) \tag{20}$$

Consequently, the IWO algorithm's task would be to maximize the fitness function. The sampling step is modified as follows:

1. The fitness of each particle is evaluated, and the particles are ranked based on their fitness in the population; i.e., those particles which correspond to greater weights are of higher rank.

2. Perform steps 3, 4, and 5 of the IWO algorithm as described in section III until a predefined number of iteration cycles is reached. It is worth noting that since the basic PF is considerably time-consuming the maximum number of iteration cycles should be chosen as a compromise between estimation performance and algorithm run-time.
3. Subsequently, the weights are updated and normalized using equations (14) and (15).
4. In order to reproduce and pick out the particles with larger weights the resampling step is implemented. That is:

$$\left\{ \tilde{x}_k^i, w_i^k \right\}_{i=1}^N = \left\{ \tilde{x}_k^i, \frac{1}{N} \right\}_{i=1}^N \tag{21}$$

V. Simulation Results

V.1. Numerical Example

This section discusses the results based on the method presented in this study. Both the PF and PFIWO algorithms are applied to estimate the hidden states in case of a nonlinear time-varying system perturbed by non-Gaussian noise and nonlinear time-varying sensor model. Suppose a Gamman probability distribution function given by:

$$p(x,\alpha,\beta) = x^{\alpha-1} \frac{e^{-\frac{x}{\beta}}}{\beta^{\alpha}\Gamma(\alpha)} \sim gamma(\alpha,\beta) \tag{22}$$

where $\Gamma(\cdot)$ represent the Gamma function, α is the shape parameter, and β is the scale parameter. Let $\alpha = 2$, and $\beta = 3$. This density function is used to characterize the noise associated with the model dynamics. Consider the following state-space system model:

$$\begin{bmatrix} x_1^k \\ x_2^k \\ x_3^k \end{bmatrix} = \begin{bmatrix} 1 + sin(0.04\pi t) + 0.5x_1^{k-1} \\ cos\left(0.05\pi x_2^{k-1}\right) \\ x_1^{k-1}x_2^{k-1} + sin\left(x_3^{k-1}\right) \end{bmatrix} + gamma(3,2) \tag{23}$$

and the sensor model is:

$$z^k = y^k + N(0,0.5) \tag{24a}$$

in which:

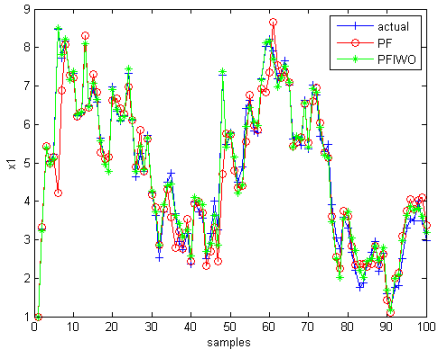
$$y^k = \begin{cases} \frac{\left(x_3^k\right)^2}{5} & t \le 30 \\ \frac{x_3^k - 4}{2} & t > 30 \end{cases} \tag{24b}$$

wherein superscript k accounts for discrete time, and $N(0,0.5)$ is a zero mean Gaussian white noise with a variance of 0.5. The particle filter as discussed in section II is applied ($N = 50$ and $N_{th} = 30$). Additionally, the PFIWO method as described in section IV is similarly utilized. The corresponding values for different parameters regarding PFIWO algorithm are listed in Table I.

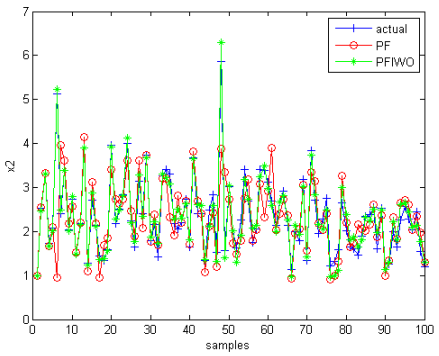
TABLE I
PARAMETERS USED IN THE PFIWO METHOD

$iter_{max}$	σ_0	σ_f	N_{max}	N_{min}	Max. Weed Number
20	0.1	0.00001	5	1	50

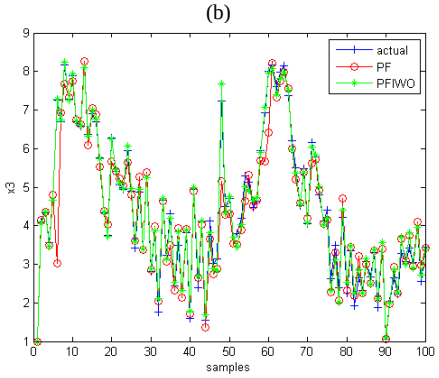
Note that the *Max. Weed Number* should be equal to the number of particles N. The state estimation results are also portrayed in Figs. 3.



(a)



(b)



(c)

Figs. 3. The estimation performance of the PF and PFIWO Algorithms. (a) x_1 (b) x_2 (c) x_3

Accordingly, the implementation of the proposed PFIWO scheme has led to improved state approximation performance. For the purpose of comparison, the results obtained with different number of particles for the system given by Eqs. (23) and (24) are also provided in Table II which includes run-time and mean square error (MSE). As it can be deduced from the table, the PFIWO algorithm is slightly slower than the PF algorithm; nonetheless, it contributes to considerable higher estimation accuracy. Also, it is worth noting that increasing the number of particles not necessarily leads to greater estimation performance. Furthermore, one has to consider the computational burden imposed by increasing the number of particles.

TABLE II
OVERVIEW OF THE SIMULATION RESULTS OBTAINED USING PF AND PFIWO

Algorithm	N	Run-time (seconds)	MSE		
			x_1	x_2	x_3
PF	15	0.3782	0.1797	0.2162	0.1351
	25	0.5213	0.1554	0.2039	0.0926
	50	0.7936	0.1819	0.2275	0.1078
PFIWO	15	0.5641	0.0340	0.0876	0.0267
	25	0.8229	0.0213	0.0313	0.0311
	50	1.1795	0.0224	0.0352	0.0329

V.2. Angular Velocity Estimation of an Induction Machine

To further evaluate the usefulness of the proposed PFIWO algorithm, both the PF and PFIWO are applied to estimate the angular velocity of an induction machine which exhibits nonlinear dynamics [13]-[15]. The state-space model describing the dynamics of a three-phase induction machine can be expressed as [15]:

$$\dot{x}_1(t) = k_1 x_1(t) + u_1(t) x_2(t) + k_2 x_3(t) + u_2(t) \quad (25a)$$

$$\dot{x}_2(t) = -u_1(t) x_1(t) + k_1 x_2(t) + k_2 x_4(t) \quad (25b)$$

$$\dot{x}_3(t) = k_3 x_1(t) + k_4 x_3(t) + \left(u_1(t) - x_5(t)\right) x_4(t) \quad (25c)$$

$$\dot{x}_4(t) = k_3 x_2(t) + k_4 x_4(t) + \left(x_5(t) - u_1(t)\right) x_3(t) \quad (25d)$$

$$\dot{x}_5(t) = k_6 u_3(t) + k_5 \left(x_1(t) x_4(t) - x_2(t) x_3(t)\right) \quad (25e)$$

wherein k_i, $h = 1,2,\ldots,6$ are parameters associated with the machine drive. $x_1(t)$, $x_2(t)$, $x_3(t)$, $x_4(t)$ are the respective components of the stator and rotor flux in the plane perpendicular to the rotation axis. $x_5(t)$ denotes the angular velocity. The inputs to the system are described next. $u_1(t)$ is the frequency of the stator voltage, $u_2(t)$ is the amplitude of the stator voltage, and $u_3(t)$ signifies the load torque. The outputs of the system are the stator currents given by:

$$z_1(t) = k_7 x_1(t) + k_8 x_3(t) \quad (26a)$$

$$z_2(t) = k_7 x_2(t) + k_8 x_4(t) \quad (26b)$$

in which k_7 and k_8 are constant parameters. The system is simulated using 500 steps of the Euler-Maruyama method with $\Delta t = 0.1s$. The system initial conditions are set as $x(0) = [0.2\ -0.6\ -0.4\ 0.1\ 0.3]^T$, and the values for different parameters are chosen as $k_1 = -0.186$, $k_2 = 0.178$, $k_3 = 0.225$, $k_4 = -0.234$, $k_5 = -0.081$, $k_6 = -0.018$, $k_7 = 4.643$, and $k_8 = -4.448$. Also, the system inputs are given as $u_1(t) = 1$, $u_2(t)$, and $u_3(t)$. The PF algorithm was applied with ($N = 100$ and $N_{th} = 85$). The parameters regarding the suggested PFIWO algorithm are provided in Table III, as well.

TABLE III
PARAMETERS USED IN THE PFIWO METHOD

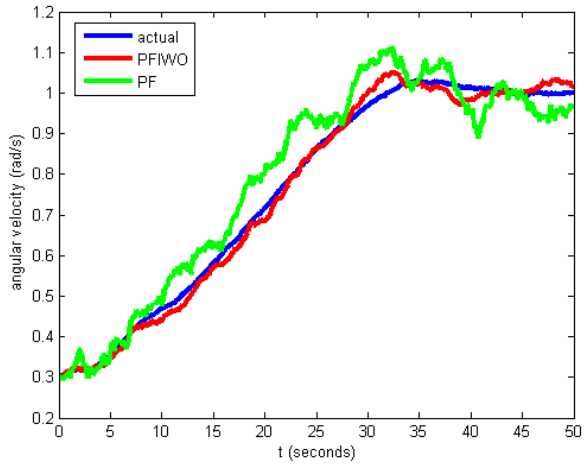
$iter_{max}$	σ_0	σ_f	N_{max}	N_{min}	Max. Weed Number
15	0.1	0.00001	3	1	100

It should be noted N_{min} should always be selected greater than 0 to ensure the participation of all particles in the reproduction stage of the IWO algorithm. It follows from the results and discussions presented in this section that the user should make a compromise between estimation accuracy and run-time which extremely depends on the application.

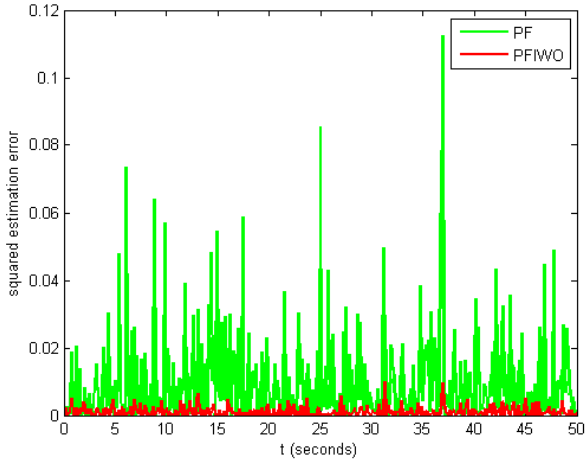
The angular velocity approximation results are given in Figs. 4. It can be perceived from the figure that with an equal number of particles the PFIWO methodology brings about superior estimation performance even in the presence of severe nonlinear model structures.

VI. Conclusion

An enhanced PF algorithm established upon the IWO scheme is proposed. Firstly, the sampling step is transformed to an optimization problem by defining an apt fitness function. Then, the IWO algorithm is exploited to deal with the optimization problem efficiently. The simulation results based on the proposed methodology are supplemented which verifies the algorithm's accuracy. It is demonstrated through simulations that the PFIWO is slightly slower and imposes somewhat more computational burden than the generic PF, yet it would contribute to noticeably greater estimation precision. The proposed algorithm can be used for state estimation of highly nonlinear plants, and can significantly deal with the shortcomings of the conventional PF.



(a)



(b)

Figs. 4. Stimation of angular velocity using the PF and PFIWO Algorithms (a) estimation results (b) estimation errors

References

[1] B.D.O. Anderson, J.B. Moore, *Optimal Filtering*, (Prentice Hall, 1979).

[2] T. Kailath, A. Sayed, B. Hasibi, *Linear estimation*, (Prentice Hall, 2000).

[3] R.J. Elliott, L. Aggoun, J.B. Moore, *Hidden Markov Models: Estimation and Control*, 3rd Edition, (Springer-Verlag, 2008).

[4] B. Ristic, M. Arulampalam, A. Gordon, *Beyond Kalman Filters: Particle filters for tracking applications*, (Artech House, 2004).

[5] N. Gordon, D. Salmond, A.F. Smith, A novel approach to nonlinear/non-Gaussian Bayesian estimation, *IEEE Proc. F, Radar Signal Process., vol. 140*, 1993, pp.107-113.

[6] R. Douc, O. Cappe, E. Moulines, "Comparison of resampling schemes for particle filtering", in *4th international Symposium on Image and Signal Process. and Analysis (ISPA*), Zagreb, Croatia, Sept. 2005.

[7] G. Tong, Z. Fang, X. Xu,"A particle swarm optimized particle filter for nonlinear system state estimation", *in IEEE Congress on Evolutionary Computing ~CEC 2006~*, 2006, pp. 438-442.

[8] J. Zhao, Z. Li, Particle Filter based on Partilce Swarm Optimization resampling for vision tracking, *Expert Systems with Applications, vol. 37*, December 2010, pp. 8910-8914.

[9] W. Jing, H. Zhao, C. Song, D. Liu,"An optimized particle filter based on PSO algorithm", *International Conference on Future BioMedical Information Engineering*, 2009, pp. 122-125.

[10] H.R. Kunsch, Recursive Monte Carlo filters: algorithms and theoretical analysis, *Annuals of Statistics*., vol. 33, no. 5, 2005, pp. 1983-2001.

[11] A.R. Mehrabian, C. Lucas, A novel numerical optimization algorithm inspired from weed colonization, *Ecological Informatics, vol. 1*, 2006, pp. 355 – 366.

[12] S. Karimkashi, A.A. Kishk, Invasive Weed Optimization and its Features in Electromagnetics, *IEEE Transactions on Antennas and Propagation, vol. 58, no. 4*, 2010, pp. 1269-1278.

[13] R. Kandpu, B. Foss, L. Imsland, Applying the unscented Kalman filter for nonlinear state estimation, *Journal of Process Control, vol. 18*, 2008, pp. 753-768.

[14] K. Reif, F. Sonnemann, R. Unbehauen, An EKF-based nonlinear observer with a prescribed degree of stability, *Automatica, vol. 34, no. 9*, 1998, pp. 1119-1123.

[15] L. Salvatore, S. Stasi, L. Tarchioni, A new EKF-based algorithm for flux estimation in induction machines, *IEEE Transactions on Industrial Electronics, vol. 40*, 1993, pp. 496 – 504.

Authors' information

Electrical Engineering Department,
Faculty of Engineering,
University of Guilan, Rasht, Iran.

Mohamadreza Ahmadi was born in Rasht, Iran, in 1988. He has received his BSc. in electrical engineering with first class honours from University of Guilan, Iran, in 2009. Currently, he is pursuing his masters studies in electrical engineering at the same university. He is a gold member of the IEEE. His current research interests include system identification, modeling, estimation, nonlinear filtering, and hybrid systems.

Hamed Mojallali was born in Fouman, Iran, in 1974. He Received the Ph.D. degree in Control Engineering from Iran University of Science and Technology (IUST), Iran, in 2006. He is currently assistant professor at Electrical Engineering Department, University of Guilan, Rasht, Iran. His current research activities include modeling and system identification, evolutionary algorithms and hybrid systems.