Appl Intell (2018) 48:834–853
DOI 10.1007/s10489-017-1013-1

A novel fault prognostic approach based on particle filters and differential evolution

Luciana B. Cosme[1,2] · Marcos F. S. V. D'Angelo[3] · Walmir M. Caminhas[4] · Shen Yin[5] · Reinaldo M. Palhares[4]

Published online: 4 August 2017

Abstract This paper proposes an improved fault prognostic approach based on a modified particle filter with a built-in differential evolution characteristic. The main methodological contribution of this study is to handle the problem of sample impoverishment faced by particle filters when only a few particles are resampled. This is done by incorporating modified mutation and selection operators for differential evolution into the proposed particle filter. The proposed method is performed to deal with two real applications of condition monitoring and fault prognosis, namely an accelerated degradation of bearings under operating conditions from the platform PRONOSTIA and a high-speed computer numerical control (CNC) milling machine 3-flute cutters.

Keywords Differential evolution · Fault prognostic · Particle filters

✉ Luciana B. Cosme
luciana.balieiro@ifnmg.edu.br

Marcos F. S. V. D'Angelo
marcos.dangelo@unimontes.br

Walmir M. Caminhas
caminhas@cpdee.ufmg.br

Shen Yin
shen.yin@hit.edu.cn

Reinaldo M. Palhares
rpalhares@ufmg.br

[1] Federal Institute of Norte de Minas Gerais, Campus Montes Claros, Montes Claros, Brazil

[2] Graduate Program in Electrical Engineering, Federal University of Minas Gerais, Av. Antônio Carlos 6627, Belo Horizonte, MG, Brazil

[3] Department of Computer Science, UNIMONTES, Av. Rui Braga, sn, Vila Mauricéia, Montes Claros, Brazil

[4] Department of Electronics Engineering, Federal University of Minas Gerais, Belo Horizonte, Brazil

[5] Harbin Institute of Technology, Harbin, China

1 Introduction

Particle filters (PF) have been shown to be an efficient method for estimation and prediction mainly due to their capacity to assess the hidden system states from noisy measurements [2, 10, 34]. Specifically for fault prognostic, the current machine/component condition can be seen as a system state which is accessed based on the observation of some parameters related to the internal tool degradation and the link between PF and fault prognostic is immediate. This kind of strategy has been widely employed in the literature [4, 14, 19, 21, 23, 24, 31, 33, 36, 39] since it allows predicting the remaining useful life (RUL) of industrial components which is an important task contributing to reduced unexpected failures and maintenance costs. In general, PF provides a discrete weight approximation to the correct posterior distribution by performing simulations of a set of independent random variables called particles. These particles are sampled from the state space model associated with their weights which are updated based on the likelihood of new measurements. The reader can find in the literature

good reviews covering fault prognostic models [13, 27], general particle filters [2, 8], and a very recent one focused on particle filter for prognostics [14].

Unfortunately, PF has some drawbacks as, for example, with the increasing number of iterations only a single particle weight becomes significant. Thus, the method fails to represent the posterior distribution which can lead to erroneous state estimation. Another drawback is that a lot of computing effort can be wasted to update trivial weights. A resampling step avoids particles degeneration to overcome some of the drawbacks. Additionally, resampling implies in replace low probability particles with the same number of high probability particles. There are a few different ways to select which particles will survive, a review of these techniques can be found in [8]. However, the resampling step can introduce diversity loss of the particles. This impoverishment problem occurs when the resampling procedure forces the selection of a few large weight particles around a region of high likelihood which may not represent the true state. Several strategies have been proposed in the literature to deal with diversity loss of the particles [7]. On the other hand, from the perspective of evolutionary computation, there are only a few available works in the literature related to PF. Very recently, authors in [37] proposed a genetic algorithm (GA) based approach, called Intelligent Particle Filter (IPF), but which may fail to find a good state estimation if the likelihood around the true state is small. A sequential evolutionary filter (SEF) has been proposed by [38] which also uses specific genetic operators for the problem of loss of particle diversity, but its authors suggest to substitute the resampling step. Other related work in [11] uses a different genetic operator, but the states are represented by a binary code. Another GA strategy using a just crossover and selection operators has also discussed in [16]. Specifically, in [17, 18] an approach is presented based on differential evolution but replacing the resampling step such that the particle selection is made by comparing one particle generated by the PF with its respective vector created from the DE operations. Clearly, there is no guarantee this method may mitigate diversity loss without increasing the variance. Furthermore, [17, 18, 40] use a large number of generations to be executed at each time instant which implies large computational efforts.

The main contribution of this paper is to present an improved fault prognostic approach based on a modified PF in which genetic operators based on differential evolution are proposed to mitigate the particle diversity loss. This work differs from other ones by introducing new mutation and selection operators while the resampling step is kept. This methodology ensures that high particle weights can be selected avoiding degeneracy. Furthermore, to illustrate the performance of the proposed method, comparisons are made with others evolutionary PF and the standard PF. The results indicate that the proposed approach may be more appropriate to deal with cases for which there is little information about the true state. Additionally, this paper presents a health monitoring experiment for an accelerated degradation of bearings under operating conditions from the platform PRONOSTIA [22] and a fault prognostic study of the case for a high-speed computer numerical control (CNC) milling machine 3-flute cutters (Röders Tech RFM760) [25] using the proposed modified PF method.

The remainder parts of this work are organized as follows: Section 2 presents the proposed particle filter and the essential background theory, Section 3 shows the details of how to use the proposed particle for the problem of predicting faults, and finally, the concluding remarks are drawn in Section 4.

2 Proposed method

This section presents the details about the proposed particle filter which is called Particle Filter based on Differential Evolution (PF-DE) and introduces the required theoretical background.

2.1 Particle filters

Particle filters employ independent random variables which are sampled from the state space to characterize the posterior probability. The update step calculates the particle weights from the likelihood of the new measurements. These numerical methods are suitable for nonlinear and non-gaussian scenarios and when compared to conventional filters as, for example, the extended Kalman filter (EKF), the main advantage of particle filters is that they do not rely on any linearization technique or any analytical approximation [8]. Instead, it uses Monte Carlo (MC) simulation to approximate a solution when the exact one is difficult to obtain.

Although there are different kinds of particle filters, we have used the bootstrap filter type as proposed in its simplest version in [5] and the time-discrete state-space system is given by:

$$x_k = f_k(x_{k-1}, w_{k-1}), \tag{1}$$

where $k \in \mathbb{N}$ is the time step, $x_k \in \mathbb{R}^n$ is the state vector, f_k is a nonlinear function of the state x_{k-1} and $w_{k-1} \in \mathbb{R}^{n_w}$ which is the random process noise sequence. The overall aim is to estimate recursively the hidden state x_k from measurements. The measurement model y_k may be represented as follows:

$$y_k = h_k(x_k, n_k), \tag{2}$$

where h_k is a nonlinear function representing the relationship between measurements $y_k \in \mathbb{R}^m$ and states x_k,

and $n_k \in \mathbb{R}^{m_n}$ which is the random measurement noise sequence. Assume that x_0, n_k, and w_k are mutually independent.

System (1) expresses the transition probability of the state $p(x_k|x_{k-1})$ while measurement model (2) describes the probability $p(y_k|x_k)$ [5]. The state estimates, x_k, are obtained based on the set of measurements available $y_{1:k} = y_i, i = 1, ..., k$ up to time k. To illustrate some particle filter operation details, consider that $\{x^i_{0:k}, W^i_k\}^{N_p}_{i=1}$ represent the values that characterize the posterior probability density function (pdf) given by $p(x_{0:k}|y_{1:k})$, where $N_p \in \mathbb{N}$ is the number of samples or particles, $x^i_{0:k}, i = 0, ..., N_p$ is a set of samples with associate weights given by $W^i_k \in \mathbb{R}, i = 1, ..., N_p$ and $x_{0:k} = x_j, j = 0, ..., k$ is the set of states up to time k. The posterior density at time k can be approximated by:

$$p(x_{0:k}|y_{1:k}) \approx \sum_{i=1}^{N_p} W^i_k \delta(x_{0:k} - x^i_{0:k}), \tag{3}$$

where $\delta(.)$ is the Dirac delta measure and the weights W^i_k are updated by [2]:

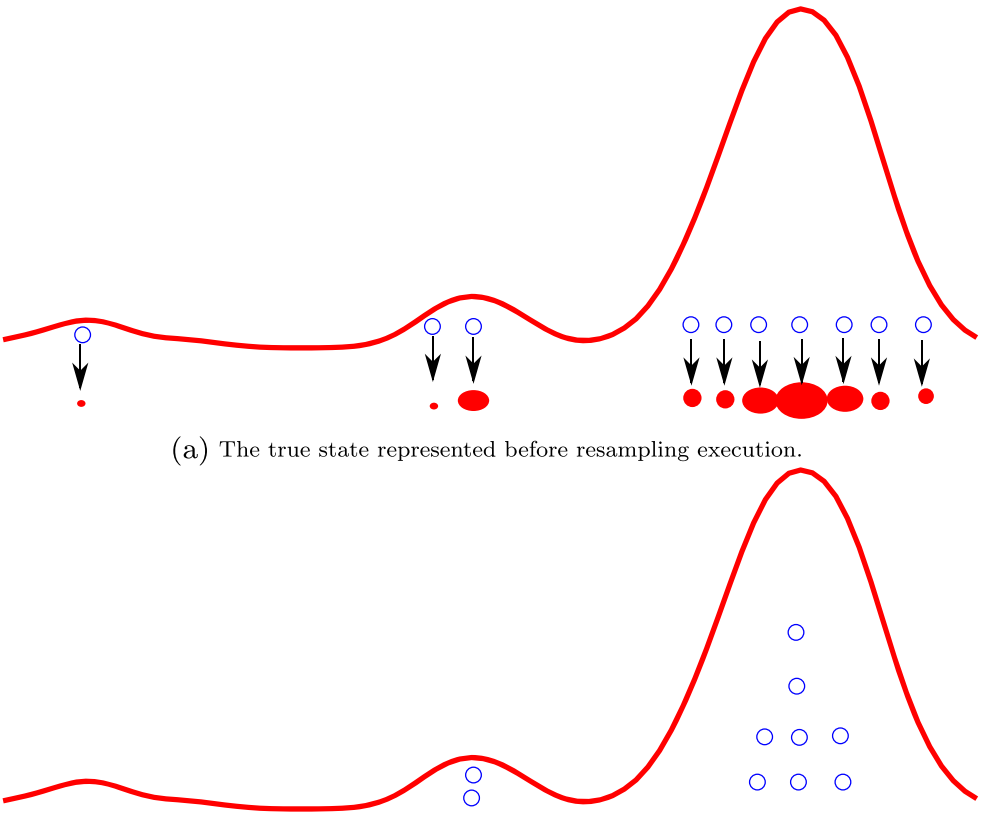
$$W^i_k = W^i_{k-1} p(y_k|x^i_k). \tag{4}$$

The degeneration effect is an important aspect which occurs frequently in the particle filter since most particles have negligible weights due to their increased variance over time. According to [5], the disadvantage lies on the large computational effort to update weights with trivial values. Thus, resampling techniques can prevent the update of particles that contribute little to compute $p(x_k|y_{1:k})$. It may be performed between the sampling steps or, as mentioned in [2, 5], a measure of degeneracy, say $\hat{N}_{eff}$, can be calculated to guide the resampling. A measure of degeneracy is given by

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_p} (W^i_k)^2}, \tag{5}$$

where W^i_k are the normalized weights obtained from (4). If $\hat{N}_{eff}$ is less than a given threshold N_T it indicates that the resampling steps may be working well.

There are several resampling techniques where the main objective is to generate a new set of samples to replace particles with trivial weights and to duplicate samples with larger weights [2, 8]. Figure 1 depicts (in a simple way) the resampling step where the red line represents the estimated state pdf and the blue circles are the particles. Only particles in regions with high probabilities have duplicated copies after the procedure. However, the reasampling step results in particle impoverishment because of diversity loss and this occurs since resampling eliminates low weight particles. The resampling step may lead to wrong estimations when it



(a) The true state represented before resampling execution.

(b) The true state represented after resampling execution.

Fig. 1 Resampling step. *Red line* represents the true pdf and the blue circles are particles at time instant k. In Fig. 1a, the particles are generated by (1). The *red filled circles* represent proportionally the normalized weights of each particle. Figure 1b expresses the resampling step which only selects copies of particles with larger weights

concentrates only on a few points that do not represent the true state.

2.2 Particle Filter based on Differential Evolution – PF-DE

Before presenting the PF-DE, the modified PF proposed in [37]—called IPF—is revisited. Its aim is to enlarge the search space in order to better approximate the posterior distribution. The IPF selection strategy consists on separating the particles generated at time instant k from (1) into two groups, S_L and S_H, where $S_L = \{x_k^l\}_{l=1}^{N_l}$ and $S_H = \{x_k^h\}_{h=1}^{N_h}$ are the sets of small and large weight particles of length N_l and N_h, respectively. In [37] a threshold, the N_{eff}th particle weight, is used for separating the groups:

$$x_k^{l,\text{crossover}} = \alpha x_k^l + (1-\alpha)x_k^h, \tag{6}$$

where $l = 1, .., N_l$ and $h = 1, .., N_h$. For each lth element in S_L, one hth element is randomly selected from S_H. $\alpha \in [0, 1]$ is a scale factor and $x_k^{l,\text{crossover}}$ is the offspring generated from the crossover method.

And, last but not least, the IPF mutation operator is executed:

$$x_k^{l,\text{mutation}} = \begin{cases} 2x_k^h - x_k^{l,\text{crossover}} & \text{, if } r_l < p_m \\ x_k^{l,\text{crossover}} & \text{, otherwise} \end{cases} \tag{7}$$

where r_l is a random uniform value, $r_l \sim U(0, 1)$, $p_m \in [0, 1]$ is the mutation probability and $x_k^{l,\text{mutation}}$ is the offspring generated from mutation method. After GA operations the particle states x_k^l will be replaced by $x_k^{l,\text{mutation}}$.

Considering this procedure, just small weight particles are changed. According to the results presented in [37], a new particle state derived from x_k^l will be located between x_k^l and $2x_k^h - x_k^l$. This new particle will be directed to regions where there are larger weight particles. Their experiment illustrates that impoverishment problem is mitigated. Unfortunately, the IPF may not work correctly in cases where there are no significant large weight particles close to the true state. Figure 2 shows a specific time step (obtained from Section 2.3.1) in which there are a few particles before IPF execution to represent the actual state (Fig. 2a). Thus the IPF leads the particles to close to the best particles about their weights (Fig. 2b). However, this process may not explore particles that represent the actual system state as depicted in Fig. 2.

To overcome the aforementioned problem, new operators are proposed based on Differential Evolution (DE) concepts [29, 30] and its applications [1, 9, 15, 18, 41]. DE has been presented in the current literature as one of the most efficient stochastic algorithms for real-world optimization problems. Among the main advantages of this method, compared to most other evolutionary algorithms, are its simplicity of implementation, better performance regarding accuracy and robustness, a small number of control parameters and low space complexity even compared to the most competitive optimization algorithms [6].

Although this kind of method does not have an inspiration in any natural process and despite the fact it operates with a population that can be a possible solution to optimization problems, its population is compounded from genetic crossover and mutation operations that recombine the "genetic material". A fitness value is associated with each individual that reflects how adapted it is compared to others to guide the evolutionary process. The higher the fitness of an individual, the greater its chance of surviving, and

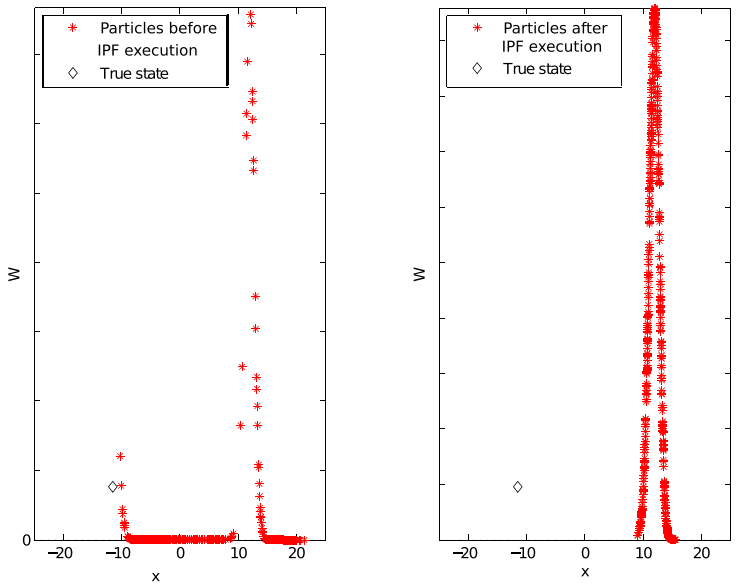


Fig. 2 Particles vs. weight in a specific time instant

(a) The true state poorly represented before IPF execution.

(b) After IPF execution it leads the particles to close to the high particle weights.

thus reproduces and transmits its characteristics to the next generation.

Consider the particle states $x_k^i, i = 1, 2, 3, ..., N_p$ at time instant k where N_p is the number of particles, this set will become the DE population. The particle selection is performed according to [37]. After this step, two particle sets $S_L = \{x_k^l\}_{l=1}^{N_l}$ and $S_H = \{x_k^h\}_{h=1}^{N_h}$ are created and include, respectively, the small and the large weight particles.

The traditional DE's mutation method creates new resulting vectors from the difference between two individuals selected randomly from the population which is then added to a third individual, producing a new mutant solution. In this work, the individuals are selected differently to cover the features of PF. The new mutation operator can be defined according to:

$$v_k^i = x_k^h + F(x_k^{r_1} - x_k^{r_2}), \tag{8}$$

where i is the index of the current mutant vector, h is an index randomly chosen from S_H ($\sim U[1, \ldots, N_h]$) and r_1 and r_2 are indices chosen randomly from the entire population, $\{S_L, S_H\}$ ($\sim U[1, \ldots, N_p]$). F is a value randomly chosen from the uniform distribution $U(0, 2)$ to avoid a parameter tuning. Thus the difference of $x_k^{r_1}$ and $x_k^{r_2}$ particles is scaled by the factor F and the resulting difference is added to x_k^h. It is also necessary to ensure that indices h, r_1, r_2 and i denote different individuals to avoid degenerate combinations. For instance, if $r_1 = r_2$ there is no disturbance applied to the mutant vector.

The resultant mutant vector $v_k^i, i = 1, 2, .., N_p$, provides the basis to increase the diversity using the crossover step and it is performed with probability $C \in [0, 1]$. The parameter C controls the fraction values in u_k^i copied from the vector v_k^i. The offspring crossover, u_k^i, is computed in the same way as the traditional DE using the offspring mutation v_k^i and the parent individual x_k^i and it is given by:

$$u_k^{ji} = \begin{cases} v_k^{ji} & , \text{ if } rand_j \leq C \text{ or } j = j_{rand} \\ x_k^{ji} & , \text{ otherwise} \end{cases} \tag{9}$$

where $j \in \{1, 2, ..., d\}$, d is the dimensional length of x_k^i, $rand_j$ is the jth value of a uniform random number which is distributed inside the interval $[0, 1]$. $C \in [0, 1]$ is the crossover probability, j_{rand} is an index from $\{1, 2, ..., d\}$ chosen randomly. The condition $j = j_{rand}$ will be checked at least once and it ensures that one of the parameters v_k^{ji} will be inherited from the mutant individual.

In DE selection, the trial solution u_k^{ji} is compared to the parent individual x_k^i and the vector that has the better fitness function will survive to the next generation. However, in this work, a new particle set will be selected as follows:

$$\{x_k^i\}_{i=1}^{N_p} = \left\{\{x_k^h\}_{h=1}^{N_h}, \{\tilde{u}_k^l\}_{l=1}^{N_l}\right\}, \tag{10}$$

where $N_p = N_h + N_l$. N_p represents the total number of particles and it is composed by N_h particles with large weights and N_l new particles.

Notice that since the vector u_k^{ji} is the result of a crossover method, $\tilde{u}$ is given as the first N_l elements from u sorted by its recalculated weights W_k^{l*} in decreasing order. Furthermore, all particles are subject to mutation and crossover operators not only small weight particles. Another point is that the highest weighted particles, $S_H = \{x_k^h\}_{h=1}^{N_h}$, are collected and it will integrate the final particles set to ensure they will not be discarded during the proposed method.

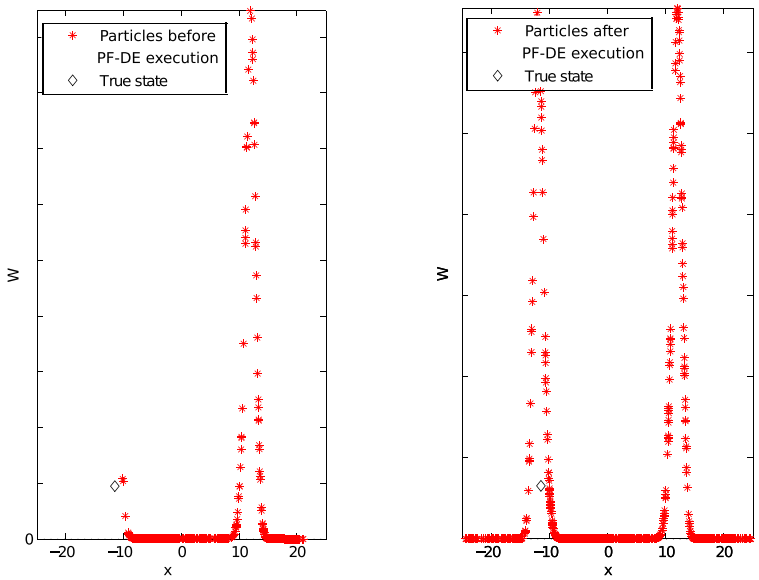


Fig. 3 Particles vs. weight in a specific time instant

(a) The true state poorly represented before PF-DE execution.

(b) After PF-DE execution it leads to close to the true state.

Figure 3 exemplifies a specific time step in which there are few particles to represent the actual state (Fig. 3a). Thus the proposed PF conducts the particles to close to the true state (Fig. 3b).

The proposal consists of two different selection steps. First, samples are chosen in the same way as IPF and the second one serves to take the better weight particles generated from operations based on DE.

Notice that the PF approach generates a particle set concentrated in a single one high-likelihood region. Thus the search space is scanned exclusively in the existing particle directions with large weights and using the genetic operators proposed in [37]. On the other hand, as the difference vectors in the DE are obtained from different amplitudes and directions this allows a complete scan over the search space and then the generated individuals are capable of extrapolating towards a second mode direction. The proposed method is summarized in Algorithm 1.

Algorithm 1 Modified Particle Filter - PF-DE

Data: Draw N_p samples $x_0^i \sim p(x_0)$, according to (1), and set $W_0^i = \frac{1}{N_p}$, $i = 1, ..., N_p$.
Result: $\{\hat{x}\}$
begin
  for *time steps* $k = 1, 2, ..., N$ **do**
    Draw N_p *samples* $x_k^i \sim p(x_k|x_{k-1}^i)$, $i = 1, ..., N_p$, *according to* (1).
    Compute the weights W_k^i *according to* (4).
    Normalize the weights $W_k^i = \frac{W_k^i}{\sum_{j=1}^{N_p} W_k^j}$.
    Calculate $\hat{N}_{eff}$, *according to* (5).
    Separate the particles into two sets, S_L *and* S_H, *according to* [37].
    Perform DE operations in all particles.
    Evaluate the weights of DE offsprings.
    Select the S_H *particles and* N_L *highest weighted offsprings and store in* $\{x_k^i\}_{i=1}^{N_p}$.
    Compute and normalize the weights W_k *of new particles set.*
    Obtain new particle set according to the particle weights from any resampling method.
    Set $W_k^i = \frac{1}{N_p}$.
    Estimate the state $\hat{x}_k = \sum_{i=1}^{N_p} W_k^i x_k^i$.
  end
  return $\{\hat{x}\}$
end

2.3 Illustrative experiments

Two numerical examples are presented to illustrate the issue of diversity loss.

Table 1 Values and standard deviation (SD) of Absolute Error (AE) for the scalar system example

Metrics	PF	IPF	PFSDE	PFSDEWR	SEF	PF-DE
$N_p = 50$						
AE	4.58	4.70	4.46	6.13	4.41	3.96
SD	0.64	0.51	0.47	0.85	0.59	0.53
$N_p = 500$						
AE	3.81	3.71	4.26	6.05	3.97	3.55
SD	0.20	0.14	0.17	0.87	0.26	0.08

To investigate the performance of the proposed method, we compare the results obtained by six methods: traditional FP, IPF [37], FP with traditional DE operators (PFSDE), PFSDE without resampling (PFSDEWR) [17], SEF [38] and PF-DE. Regarding the PF parameter settings, the same values are used in all filters. For example, IPF uses the resampling step at each run, so for a fair comparison among all these methods, we built them in the same way. Likewise, we use only one generation for all filters, to make the computational cost equivalent, differently from how it is proposed by [17, 18, 40] in which several generations have been used. In the specific case of the DE parameters, we consider $F \sim U(0, 2)$ and $C = 0.5$ and no adjustment for each problem is necessary. For the parameters of the IPF and SEF we use the values suggested by their authors for the numerical problems and these same values are adopted in the applications for fault monitoring and prognostic. To measure the performance of the filters, the Absolute Estimation Error (AE) is used similarly as in [37]. For this, the computation of the error in the jth execution is performed and it is given by:

$$e_j = \frac{1}{N}\sum_{k=1}^{N} |x_k - \hat{x}_k| \tag{11}$$

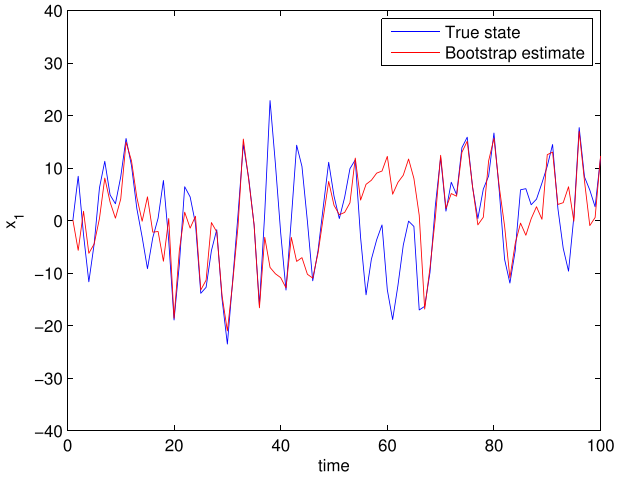
where N is the time step number, x_k is the true state and $\hat{x}_k$ is the filter predicted state at time instants $k = 1, .., N$. Thus the absolute error is:

$$AE = \frac{1}{S}\sum_{j=1}^{S} e_j \tag{12}$$

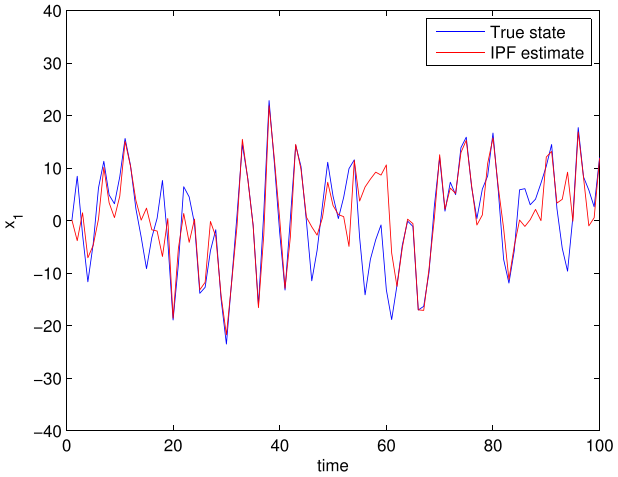
where S is the total number of simulations. Thirty independent runs of each algorithm is executed and the AE and its

Table 2 Wilcoxon signed ranks test results for the scalar system example with a level of significance $\alpha = 0.05$ taking in consideration of the AE

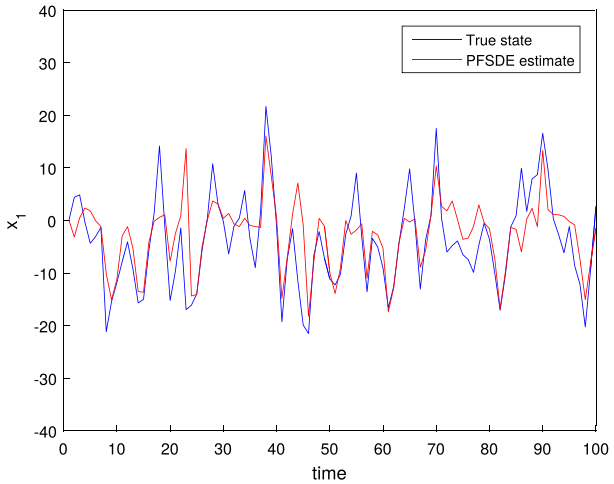
	PF/ PF-DE	IPF/ PF-DE	SDE/ PF-DE	RW/ PF-DE	SEF/ PF-DE
p-value	0.0017	0.0002	0.0011	0	0.0026



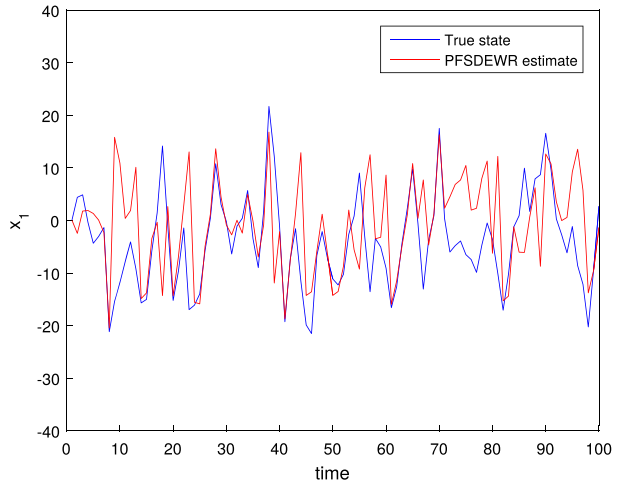
(a) PF state estimation along the time.



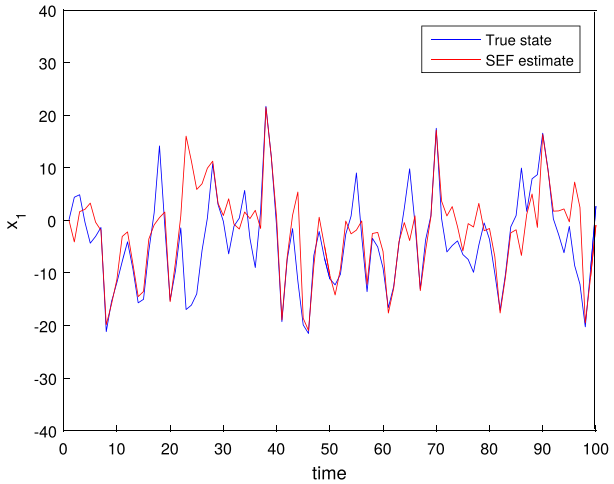
(b) IPF state estimation along the time.



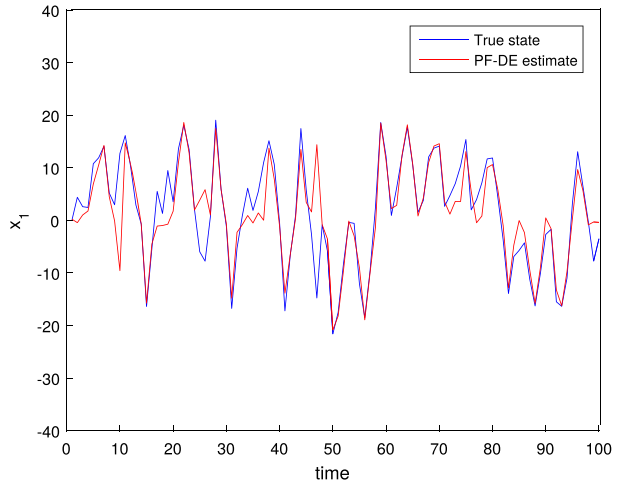
(c) PFSDE state estimation along the time.



(d) PFSDEWR state estimation along the time.



(e) SEF state estimation along the time.



(f) PFDE state estimation along the time.

Fig. 4 State estimation along the time for the scalar system example

standard deviation resulting is registered. In this work, we use $S = 30$.

To analyze the achieved results, the Wilcoxon signed rank test [35] is used to compare two different methods of state estimation in the single-problem analysis. This test belongs to the category of non-parametric statistical methods which aims to detect significant differences between two paired samples or in this case the performance of two algorithms. The tests are performed comparing PF-DE with five different methods which are grouped two by two groups. In this study, the significance level chosen is $\alpha = 0.05$ taking in consideration of the absolute error metric obtained in all runs, therefore a p-value greater than α indicates that there is no difference in the performance of the methods, otherwise, there are significant differences among the results.

2.3.1 A scalar system example

Consider the state space model:

$$x_k = \frac{1}{2}x_{k-1} + \frac{25x_{k-1}}{1+x_{k-1}^2} + 8\cos[1.2(k-1)] + w_k, \quad (13)$$

$$y_k = \frac{1}{20}x_k^2 + n_k. \quad (14)$$

where x_k is the state, y_k is the measurement sequence and $w_k \sim \mathcal{N}(0, 10)$ and $n_k \sim \mathcal{N}(0, 1)$ are sequences denoting zero-mean Gaussian white noises. This experiment is a common benchmark in literature [2, 28, 37]. Notice that for measurements $y_k > 0$ the likelihood has two modes at $\pm(20y_k)^{1/2}$ which increasingly difficult to solve to problem. The initial state is drawn considering $x_0 \sim \mathcal{N}(0.1, 2)$. The simulations have been executed using 50 and 500 particles and time step number $N = 100$.

As the filtering aims to estimate the state x_k from noise measurements y_k obtained at k, to analyze the effect caused by diversity loss when the process noise is underestimated, suppose the noise process in (13) is expected to be $w_k \sim \mathcal{N}(0, 2)$ along the filter estimation, however it is simulated as being $w_k \sim \mathcal{N}(0, 10)$.

Table 1 presents the absolute error (AE) and its standard deviation (SD) for each filter scheme considering that the process noise is underestimated and also considering $N_p = 50$ and $N_p = 500$ particles. The DE parameters are $F \sim U(0, 2)$ and $C = 0.5$, for the IPF it is selected $\alpha = 0.1$ and $p_m = 0.5$ and for SEF $p_m = 0.5$ and $\lambda = 1$ is used.

When comparing the filter estimates, the proposed PF-DE approach improves accuracy even in the presence of underestimated noise and few samples which suggest that the algorithm is able to find more new particles close to the true state. As can also be observed, the proposed algorithm has achieved a smaller absolute error (AE) even with a smaller number of particles, which contributes to a better estimate.

Table 3 Values and SD of Absolute Error (AE) for the multidimensional example

Metrics	PF	IPF	PFSDE	PFSDEWR	SEF	PF-DE
$AE_{x_1(t)}$	51350.80	481.35	–	–	532.89	344.49
$SD_{x_1(t)}$	10021.55	364.90	–	–	63.65	22.75
$AE_{x_2(t)}$	3431.66	1139.75	–	–	823.25	712.95
$SD_{x_2(t)}$	539.99	165.91	–	–	210.23	134.66
$AE_{x_3(t)}$	0.11	0.28	–	–	0.21	0.17
$SD_{x_3(t)}$	0.04	0.06	–	–	0.03	0.09

For this one-dimensional problem, the good performance of the PF-DE shows that the different amplitudes and directions generated by the DE applied to the particles with higher weights can generate better solutions when the real state is not well represented, as illustrated in Fig. 3. On the other hand, PFSDE fails to achieve good results by using the conventional DE strategy in which any particle is selected for the mutation. The results of PFSDEWR are significantly worse than the others due to the increase of the variance imposed by the FP without resampling and by the operations of the DE. The SEF method obtains similar results to the IPF because it also implements genetic operators oriented to the higher weight particles. Especially for smaller number of particles, the IPF presents the worst performance among all methods since this condition generates more instants in which the real state is not well represented, as shown in Fig. 2.

Table 2 presents the p-values obtained for all five comparisons concerning PF-DE: PF versus PF-DE (PF/PF-DE), IPF versus PF-DE (IPF/PF-DE), PFSDE versus PF-DE (SDE/PF-DE), PFSDEWR versus PF-DE (RW/PF-DE) and SEF versus PF-DE (SEF/PF-DE). As shown in Table 2, at the level of significance $\alpha = 0.05$ the difference between PF-DE and the other methods is statistically significant.

For illustration purposes, Fig. 2 depicts the state pdf estimation obtained by the IPF filter at time instant $k = 54$ and $N_p = 500$. In that case, IPF approach disregarded

Table 4 Wilcoxon signed ranks test results for the multidimensional example with a level of significance $\alpha = 0.05$ taking in consideration of the AE

	PF/ PF-DE	IPF/ PF-DE	SDE/ PF-DE	RW/ PF-DE	SEF/ PF-DE
p-value$_{x_1}$	0	0	–	–	0
p-value$_{x_2}$	0	0	–	–	0.014
p-value$_{x_3}$	0.006	0	–	–	0

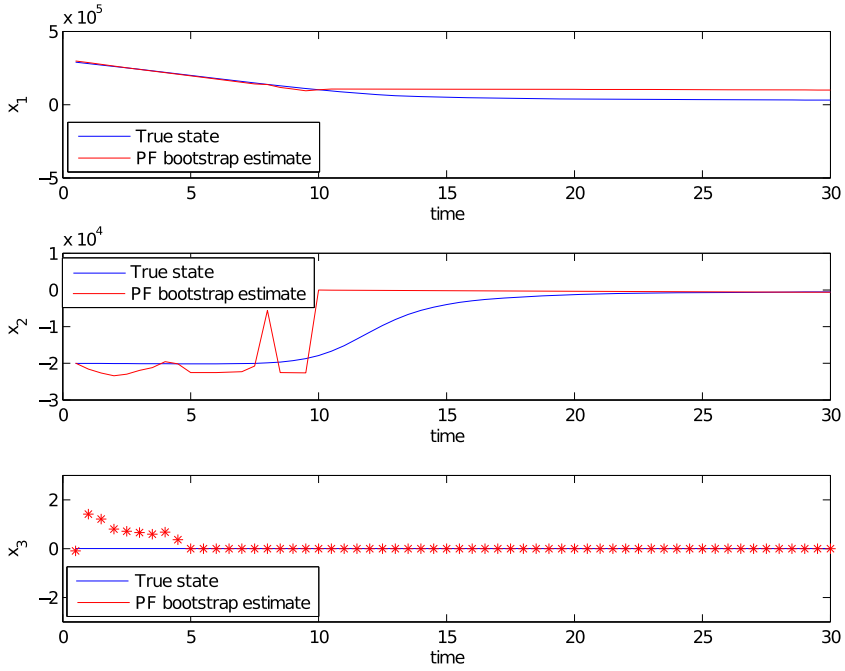


Fig. 5 State estimation results of the PF with 500 particles

particles with small weights close to the true state. On the other hand, considering the same instant $k = 54$, Fig. 3 illustrates that the proposed filter PF-DE can achieve better estimates. Additionally, the state estimation is also depicted in Fig. 4a for the standard PF, Fig. 4b for the IPF, Fig. 4c for the PFSDE, Fig. 4d for the PFSDEWR, Fig. 4e for the SEF and in Fig. 4f for the proposed method PF-DE.

2.3.2 Falling body experiment

Consider the same system studied in [28, 37] to estimate the altitude x_1,velocity x_2 and constant ballistic coefficient x_3 of a falling body towards earth. The system's equations are given in (15). Fourth-order Runge-Kutta integration with a step size of 0.05 s is used to simulate the system for

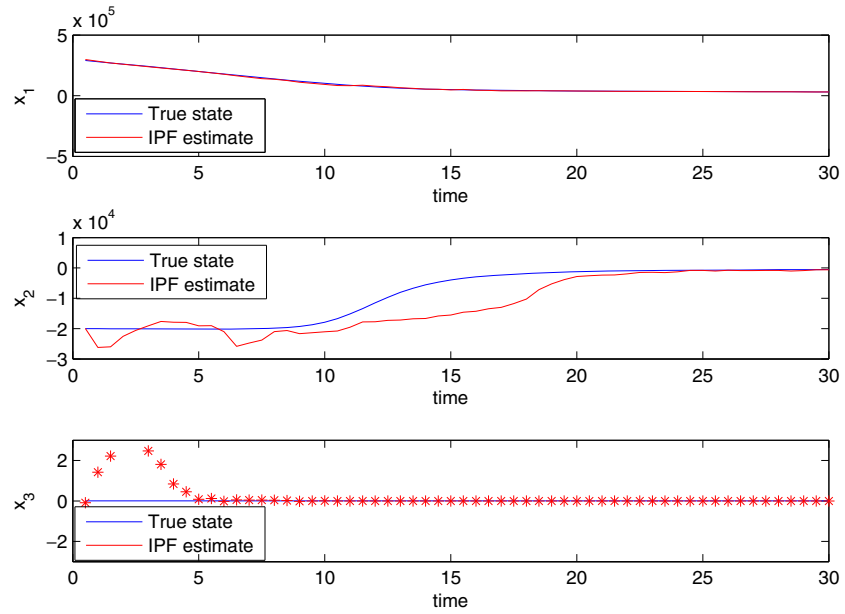


Fig. 6 State estimation results of the IPF with 500 particles

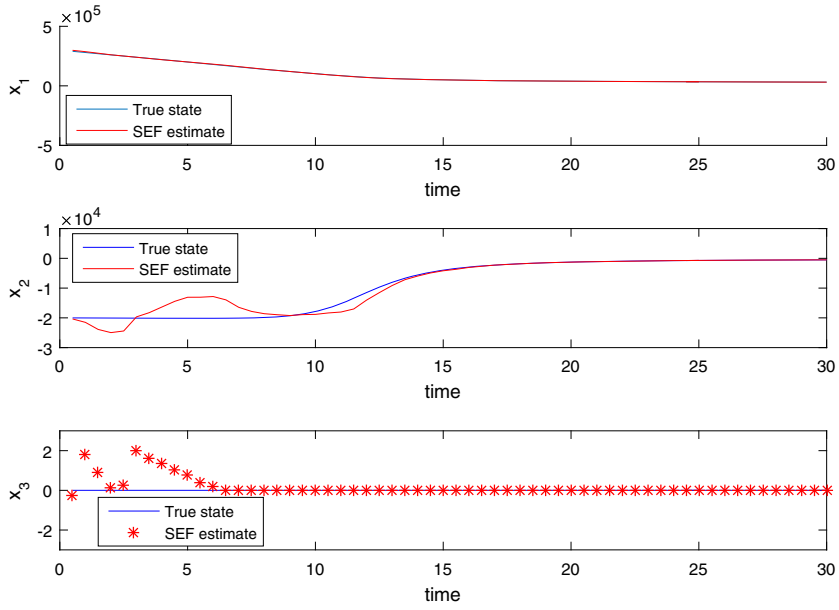


Fig. 7 State estimation results of the SEF with 500 particles

30 s. In this case, a standard particle filter implementation is not efficient. To improve the PF performance, the roughening procedure is used to prevent sample impoverishment. This procedure also adds random noise to particles after the resampling step. We used the tuning parameter $K = 0.2$ as suggested in [28]. The model is:

$$\begin{aligned}
\dot{x}_1(t) &= x_2(t) + w_1(t) \\
\dot{x}_2(t) &= \rho \exp(-x_1(t)/c) x_2(t)^2 x_3(t)/2 - g + w_2(t) \\
\dot{x}_3(t) &= x_3(t) + w_3(t) \\
y &= \sqrt{M^2 + (x_1 - a)^2} + n
\end{aligned} \tag{15}$$

where $w_i(t), i = 1, 2, 3$ is the process noise and n is the measurement noise. In each particle, x_3 must be a nonnegative value. ρ is the air density at sea level, k determines the relation between air density and altitude, and g is the gravity acceleration. Suppose range measurements are taken every 0.5 s and time step number $N = 60$. The parameters and conditions of the system at $k = 0$ are given as in [37].

This system is chosen to demonstrate the misleading results caused from sample impoverishment when there is a small process noise. Table 3 presents the values and SD of absolute error for each filter scheme considering 500 particles. The DE parameters are $F \sim U(0, 2)$ and $C = 0.5$, the IPF is executed using $\alpha = 0.9$ and $p_m = 0.9$, and for SEF $p_m = 0.5$ and $\lambda = 1$.

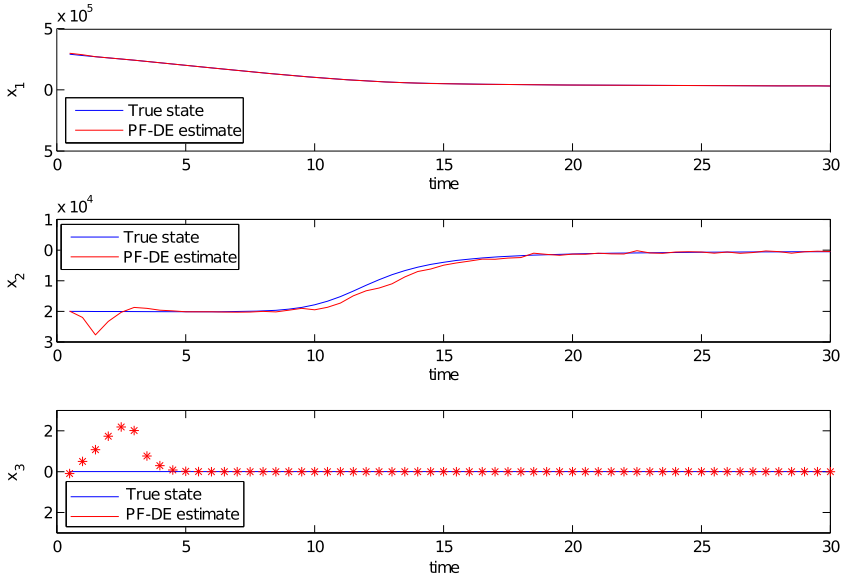


Fig. 8 State estimation results of the PF-DE with 500 particles

A traditional PF implementation is not able to ensure the convergence of (15) because the particles rapidly collapse to a single point. However, only the use of common DE operator in PFSDE and PFSDEWR is not enough to deal with this problem. Besides that, these methods are not able to cope with the increase of the variance in x_2 generated by the DE approach without specific genetic operators for PF, which lead to non-convergence, marked as (–) in Table 3. The proposed filter achieves smaller error levels than other filters, particularly for x_1 and x_2. As expected, the operators of the PF-DE provided a complete scan over state space and thus it can better represent the posterior distribution. As for the variable x_3, the proposed method found worse estimates than PF. However, this may be caused because x_3 is a constant value and it does not imply in any nonlinear dynamics in system or measurement equations. Compared to [37], PF-DE reaches similar errors using half of the number of particles. Although increasing the number of samples makes PF equivalent to the true posterior pdf [2], this is not always reasonable because it involves a large computational effort which avoids real time applications.

Table 4 presents the p-values obtained for all five comparisons concerning PF-DE: PF versus PF-DE (PF/PF-DE), IPF versus PF-DE (IPF/PF-DE), PFSDE versus PF-DE (SDE/PF-DE), PFSDEWR versus PF-DE (RW/PF-DE) and SEF versus PF-DE (SEF/PF-DE). As shown in Table 4, at level of significance $\alpha = 0.05$, the differences between PF-DE and the other methods for x_1 and x_2 are statistically significant. The Wilcoxon test also confirms that PF is better than PF-DE for x_3. As can be seen in Figs. 5 and 8, the error of x_3 presented by both methods are similar and it happens only at the beginning of the estimation. After the first few seconds, x_3 becomes constant as expected. However, on x_1 and x_2, PF-DE obtains much better estimates than PF.

Notice that the state estimations for x_1, x_2 and x_3 are depicted in Fig. 5 for the standard bootstrap PF, Fig. 6 for the IPF, Fig. 7 for the SEF and in Fig. 8 for the proposed method PF-DE. Clearly the proposed filter obtains better accurate estimates.

3 Health monitoring and prognostic experiments

This section presents details about the proposed particle filter usage for the problems of predicting faults as well as estimation of the health condition of two real applications:

- A high-speed computer numerical control (CNC) milling machine 3-flute cutters [25] and;
- An accelerated degradation of bearings under operating conditions from the platform PRONOSTIA [22].

Specifically, the experiments using the proposed method for condition monitoring and fault prognosis are performed on databases made available by the Prognostics and Health Management Society (PHM Society) [26]. The main objectives of those experiments are to infer the health condition of equipment from online measurements since the tool wear is an unmeasured tool wear.

3.1 Particle filters and health monitoring

One important aspect of the PF theory is that at least two models are required: a model describing the evolution of the system states and a model related to the noisy measurements of the state [2]. Usually, mathematical models have been established to describe them, however, their design can

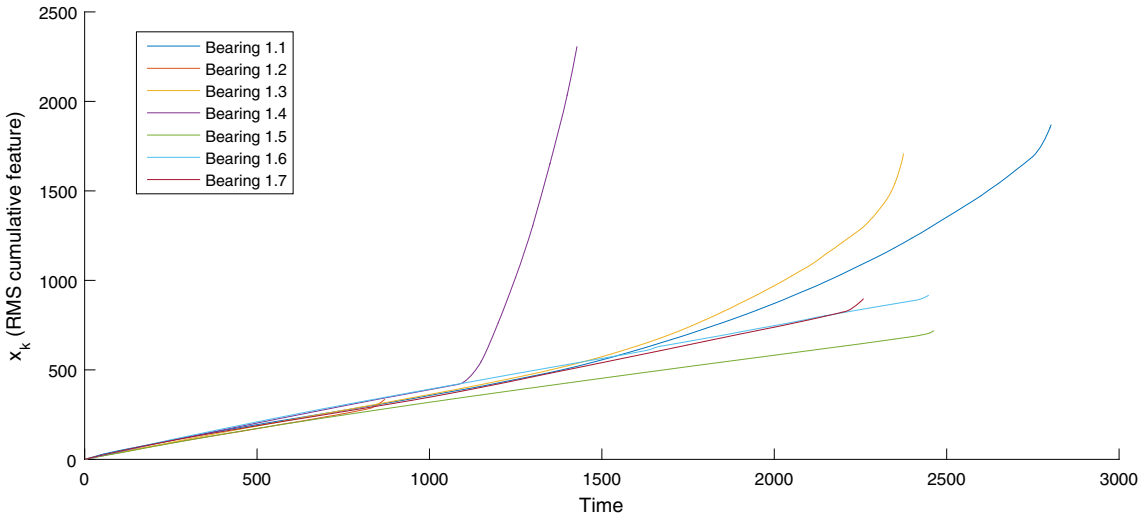


Fig. 9 RMS cumulative features as degradation model

Table 5 Operating conditions of PRONOSTIA dataset

	Operating conditions 1800 rpm and 4000 N
Learning set	Bearing1_1
	Bearing1_2
Test set	Bearing1_3
	Bearing1_4
	Bearing1_5
	Bearing1_6
	Bearing1_7

be very complex and can require expert knowledge about the degradation process. The work in [3] trains an adaptive neuro-fuzzy inference system (ANFIS) model [12] from data pairs to build the fault prognostic system. This kind of approach has been used to approximate dynamic systems without deriving complex models in a 4th order particle filter. The disadvantage of a 4th order PF relies on the fact that it needs to keep more states than the usual PF. This work uses a first-order ANFIS model. According to the previous values of the particles, one-step-ahead condition can be predicted from

$$x_k = g_k(x_{k-1}, w_{k-1}), \tag{16}$$

where the state x_k is the tool wear, w_{k-1} is the noise process and g_k is a nonlinear ANFIS function trained from data.

In the CNC machine, x_k means the level of tool wear provided by the database. A directly measured wear level on PRONOSTIA case does not exist, so a degradation model is created based on the feature selected. More details are presented in Section 3.2.1.

To estimate the measurement probability $p(y_k|x_k)$ and its corresponding relation to a given degradation state x_k, one needs to have a measurement mathematical model. For CNC machine an ANFIS model is created, with the role of a condition monitoring index, to quantify the health hidden state from measures. In PRONOSTIA data set, the measurement equation is given by $z_k = h(x_k) = x_k$.

3.2 Health monitoring experiment—PRONOSTIA

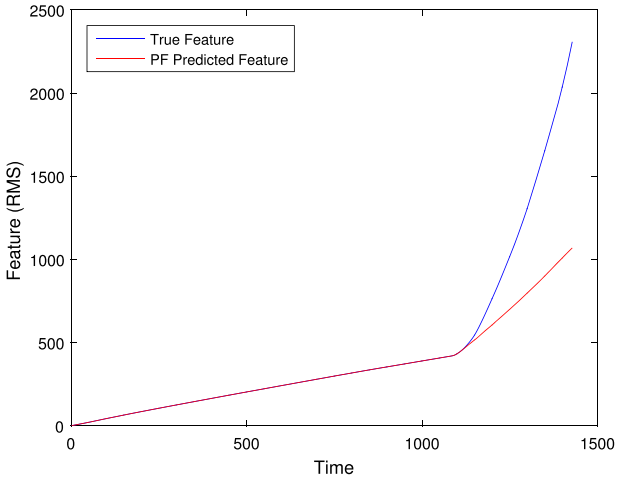
3.2.1 The data set

The PRONOSTIA platform is a data set focused on health condition estimations of bearings. Its experiments present an accelerated bearings degradation under operating conditions and it provides real-world experimental monitoring data. Furthermore, according to [22], theoretical framework (L10, BPFI, BPFE, etc.) mismatches the experimental observations.

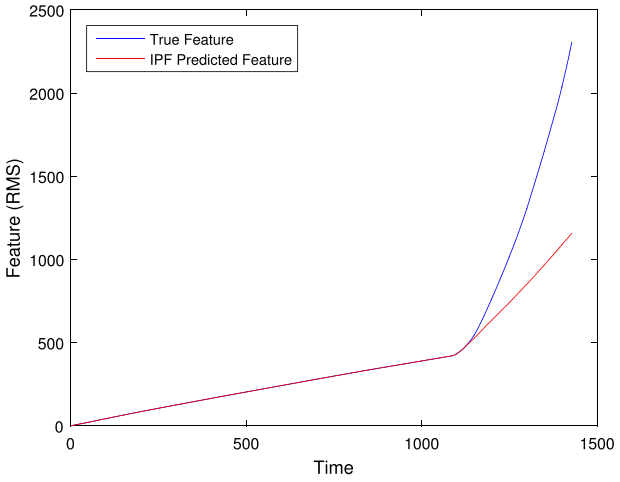
The selected characterization of the bearing's degradation is based on vibration sensor which is used for feature extraction. To proceed with the experiment, we have extracted the root mean square (RMS) from vibration data of the sensor placed on the vertical axis. In next step, RMS value is also processed using an exponential moving average with the number of previous data points equal to 12 as smoothing operation. To build a monotonic degradation process in (16) we have used a cumulative sum of RMS feature after the smoothing task. Figure 9 depicts the extracted RMS feature for different degradation trends of ball bearings along their operational life. Table 5 presents the operating

Table 6 Values and SD of Absolute Error (AE) for PRONOSTIA dataset

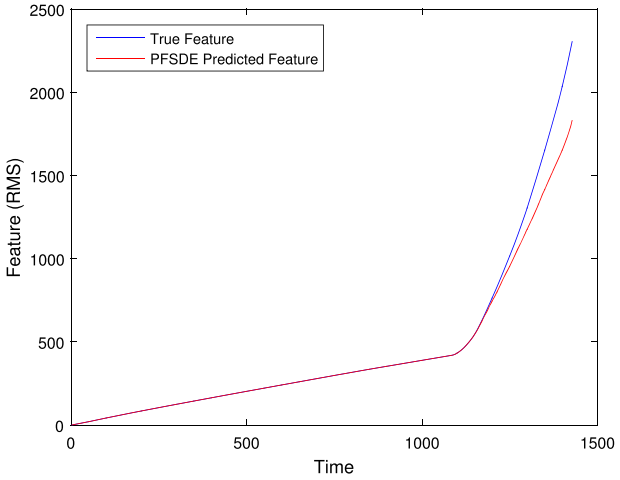
Metrics	PF	IPF	PFSDE	PFSDEWR	SEF	FP-DE
Bearing 1-3						
AE	0.99	0.81	0.17	83.54	0.22	0.06
SD	0.03	0.02	0.03	11.69	0.11	0.01
Bearing 1-4						
AE	102.22	93.17	31.54	680.14	12.76	7.03
SD	0.47	0.48	1.48	36.83	8.30	0.99
Bearing 1-5						
AE	0.0153	0.0136	0.0130	0.2274	0.0285	0.0118
SD	0.0002	0.0002	0.0002	0.0029	0.0029	0.0001
Bearing 1-6						
AE	0.0156	0.0137	0.0131	0.2364	0.0289	0.0119
SD	0.0002	0.0003	0.0002	0.0036	0.0005	0.0001
Bearing 1-7						
AE	0.0150	0.0132	0.0126	0.2170	0.0278	0.0117
SD	0.0003	0.0002	0.0002	0.0038	0.0005	0.0002



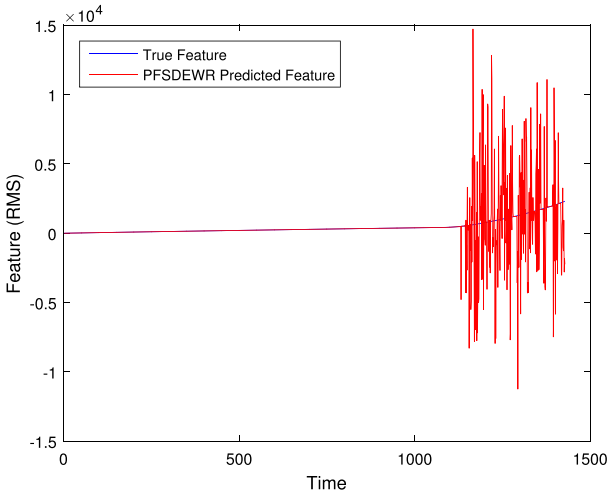
(a) State estimation results of the PF.



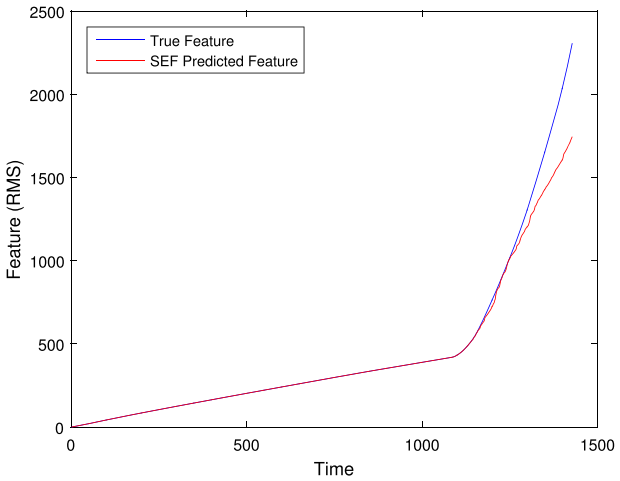
(b) State estimation results of the IPF.



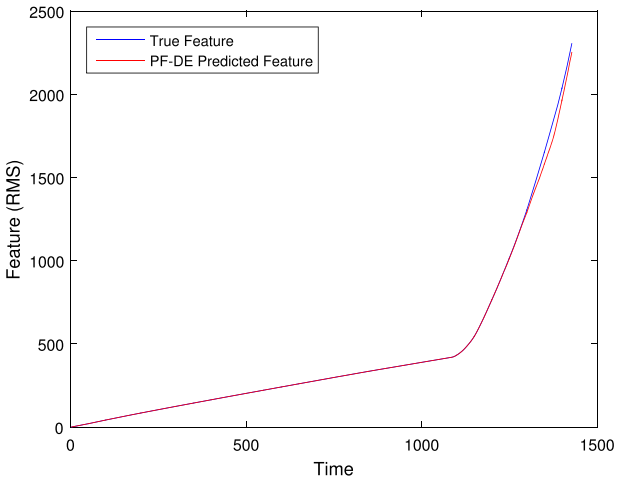
(c) State estimation results of the PFSDE.



(d) State estimation results of the PFSDERW.



(e) State estimation results of the SEF.



(f) State estimation results of the PF-DE.

Fig. 10 State estimation results of the methods

conditions as well as the training and validation sets according to [22].

3.2.2 Results

This section shows the results obtained in the database described in Section 3.2.1 with an ANFIS degradation model. Thirty independent experiments are performed. Multinomial resampling has been chosen and the initial conditions selected are $x \sim N(0, \sigma_x)$. The DE parameters are $F \sim U(0, 2)$ and $C = 0.5$. The IPF is executed using $\alpha = 0.1$ and $p_m = 0.5$ and SEF had $p_m = 0.5$ and $\lambda = 1$ parameters. The filter simulation parameters are 100 particles, $\sigma_x = 0.5$ and $\sigma_v = 0.1$, to process and measurement noises, respectively.

The learning and test sets, presented in Table 5, are used to train the ANFIS process model and for the validation step, respectively. The particle filters are used to monitor the bearings operation until their total failure. To evaluate each method performance, the absolute error (AE) and its standard deviation are computed for the data sets and the results are presented in Table 6.

Notice that from the results in Table 6 as well Fig. 9, bearings 1-3 and 1-4 have higher growth in their curves which differ from the training sets. The less accurate estimation results presented in Table 6 derive from this fact. Because the training is conducted based on the training bearings 1-1 and 1-2, the ANFIS degradation model has not generated particles capable of representing the true state in the end-of-life bearing 1-4. However, since the proposed method expands the search space, it has been able to generate new particles in areas not visited by the model and thus it has achieved better results than the other filters. Specifically for bearing 1-4, the second minor absolute error which is found by the SEF is 80% higher than PF-DE. Overall, the PF-DE method performs better estimates for all test sets. Regarding the methods based on genetic algorithms, the dynamic adjustment of the SEF parameters produces better results than the IPF for bearings 1-3 and 1-4. PFSDERW systematically does not achieve good results, especially for bearing 1-4. The variance increases severely compromising the estimate due to the absence of the resampling step, as can be seen in Fig. 10d. Figure 10 depicts more details of the health condition estimation for the bearing 1-4. The proposed method PF-DE presents a better performance, as illustrated in Fig. 10f.

For the PRONOSTIA dataset, the Wilcoxon test confirms that all results are statistically relevant (p-value < 0.05). The p-value obtained is 0 for all five comparisons except for the bearing 1-4. In this case, in the comparison between SEF and PF-DE the p-value is 0.004. Thus, it is possible to assert that there is a statistically difference between the methods and PF-DE has achieved better state estimates than the other methods.

Table 7 Values and SD of absolute error (AE) and accuracy (AC) for the CNC milling machine

Metrics	PF	IPF	PFSDE	PFSDEWR	SEF	FP-DE
AE	14.19	14.23	14.27	14.26	14.80	13.20
SD	1.13	0.88	0.12	0.07	0.28	0.50
AC	0.77	0.79	0.89	0.87	0.73	0.93
SD	0.09	0.05	0.04	0.06	0.05	0.08

3.3 Prognostic experiment and RUL equipment

According to [24], prognostics may be essentially understood as the generation of long-term predictions for a fault indicator, made with the purpose of estimating the remaining useful life (RUL) of a failing component. In PF literature describes different methods to calculate the earliest time at which the degradation state exceeds the failure threshold [3, 20]. It is possible to compute the update state model successively from (1) from every future time instant.

The long-term (p-step-ahead) condition prediction used in this work follows [3]. RUL is found by calculating the condition from instant k to instant $k + p$ when a threshold value thr is reached from each particle as initial condition at time k, as shown in

$$RUL_{k+p} = \sum_{i=1}^{N_p} r^i W_k^i, \tag{17}$$

where N_p is the total number of particles, r^i is the RUL of the ith particle, and W_k^i is the weight of the ith particle at time instant k. It has been assumed that the current weights, W_k^i, are a good representation of the state pdf at time k, then it is possible to approximate the predicted state pdf at $k + p$, considering the particle weights for future time instants is negligible with respect to other sources of error [24].

Table 8 Wilcoxon signed ranks test results for the CNC milling machine with a level of significance $\alpha = 0.05$ taking in consideration of the AE

	PF/ PF-DE	IPF/ PF-DE	SDE/ PF-DE	RW/ PF-DE	SEF/ PF-DE
p-value$_{AE}$	0.013	0.019	0.002	0.002	0.002
p-value$_{AC}$	0.008	0.002	0.010	0.014	0.002

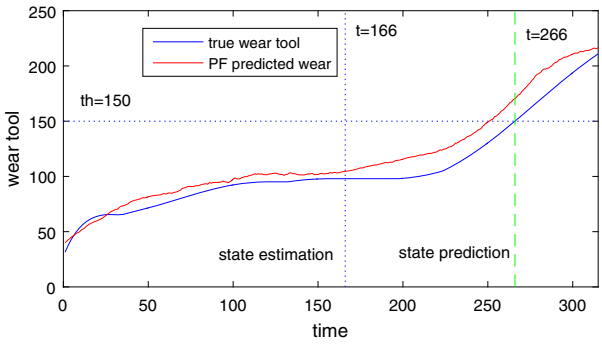
In order to evaluate the performance of the proposed method, the accuracy metric of the prediction is used [32]:

$$Accuracy(k_p) = \frac{1}{E}\sum_{j=1}^{E} \exp^{-|RUL_{k_p}(j) - \hat{RUL}_{k_p}(j)|/RUL_{k_p}(j)}, \tag{18}$$

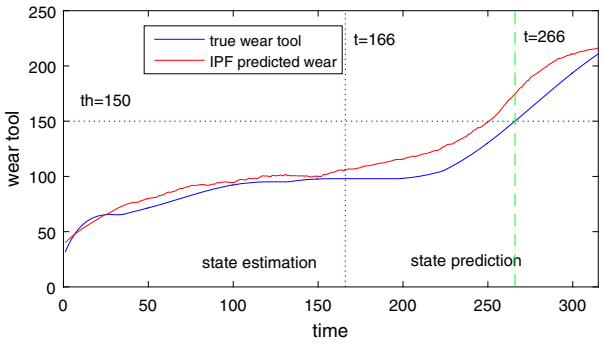
in which k_p is the time step at which the prognostic routine is executed, E is the number of experiments, $RUL_{k_p}(j)$ and $\hat{RUL}_{k_p}(j)$ are the true and estimated RUL at instant k_p, respectively, for each jth experiment. When this metric is equal to 1 indicates the best performance and when it is equal to 0, the worst one.

3.3.1 Data set

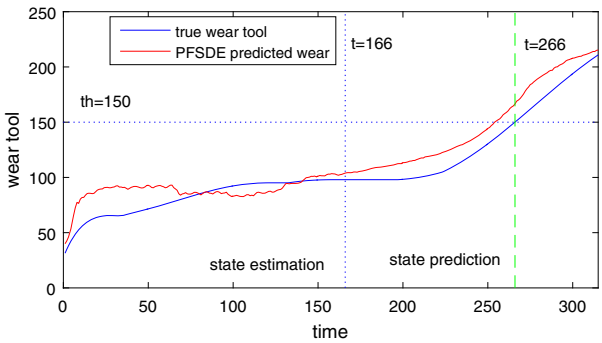
The database contains several histories of high-speed CNC milling machine 3-flute cutters (Röders Tech RFM760)



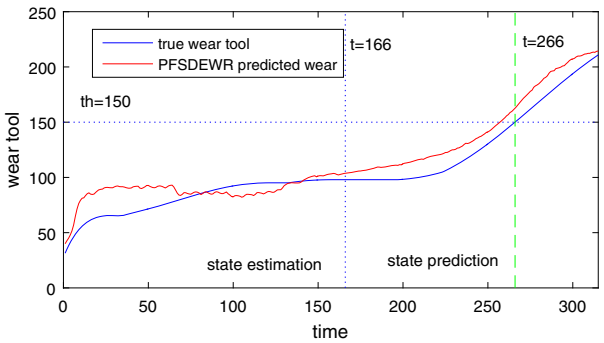
(a) Evolution particle path of c_4 database for PF.



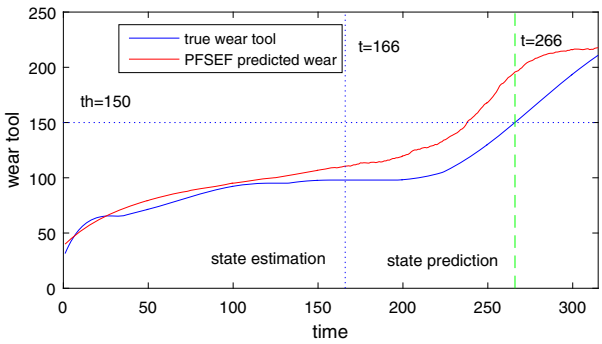
(b) Evolution particle path of c_4 database for IPF.



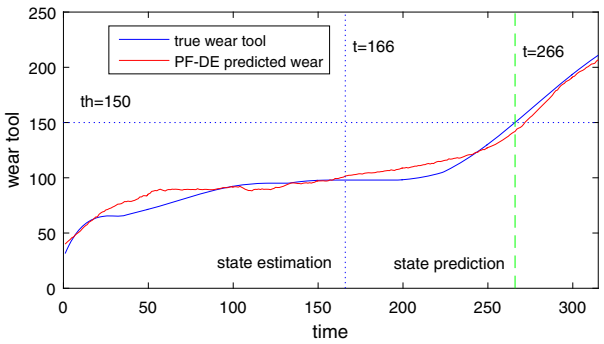
(c) Evolution particle path of c_4 database for PFSDE.



(d) Evolution particle path of c_4 database for PFSDEWR.



(e) Evolution particle path of c_4 database for SEF.



(f) Evolution particle path of c_4 database for PF-DE.

Fig. 11 Evolution particle path of c_4 database for each one of the six filter approaches. The *blue line* indicates the true state, the *red line* indicates the predicted value. The *dotted line thr* is a threshold and other *dotted ones* represent the initial and final time for prognostic step

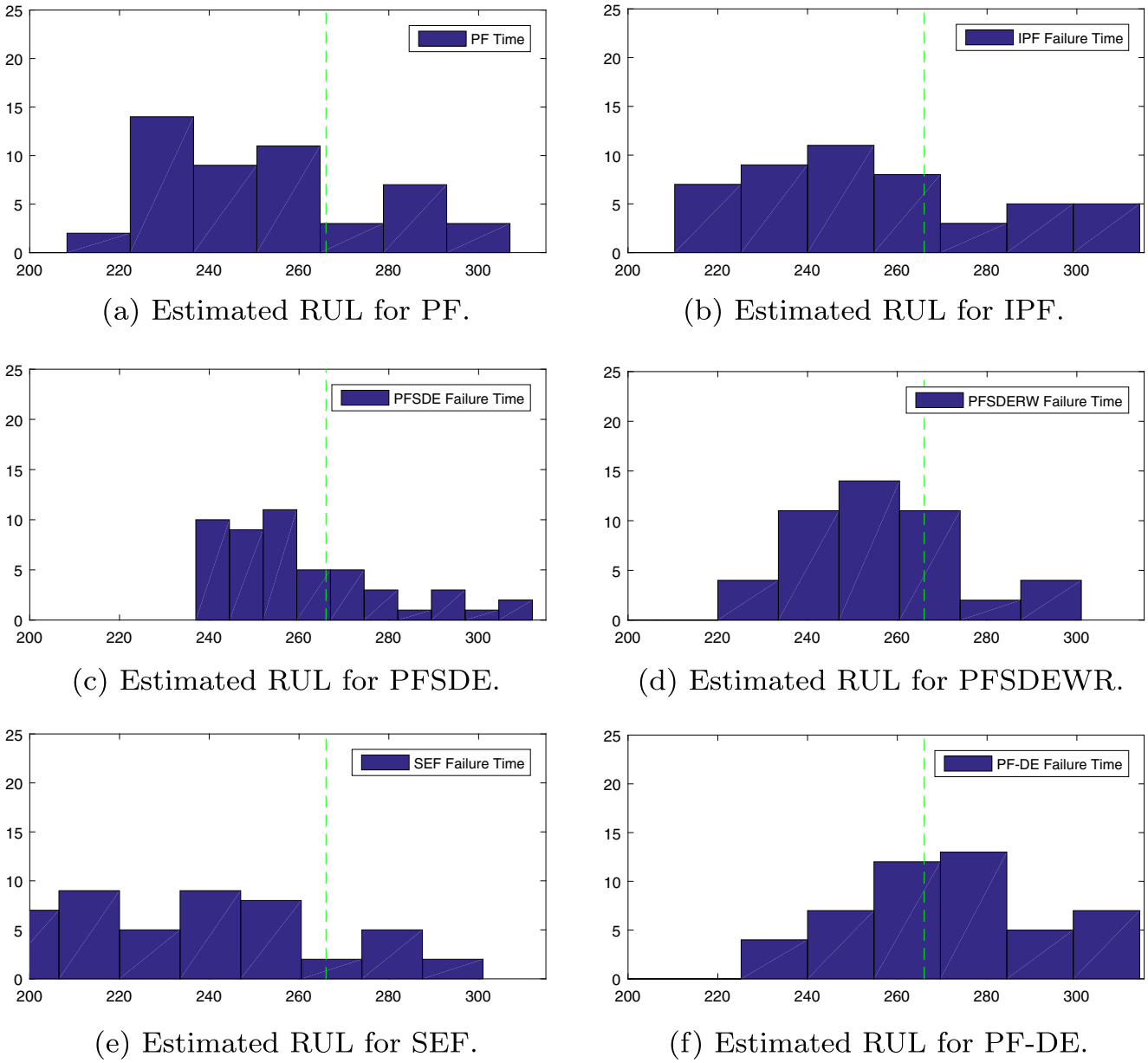
used until a significant wear stage. The challenge dataset contains six individual cutter records denoted by $c_1, ..., c_6$, but only three experiments have been used (c_1, c_4 and c_6) because they have been labeled by pairs.

The sensor data (dynamometer, accelerometers and acoustic emission) needs to be pre-processed to verify whether they can represent the wear level. Unfortunately, the wear level cannot be obtained directly on a regular operation because it cannot be acquired from a working machine. We extracted the root mean square (RMS), the peak and the standard deviation from dynamometer, the RMS and the kurtosis from accelerometers and finally, the mean and the standard deviation from acoustic emission.

Pearson's correlation coefficient has been calculated between the extracted features and the tool wear width. The maximum wear value among the three flutes has been used for a conservative estimate because the objective is to stop the machine for predictive maintenance when a threshold is achieved. The highest value correlation between features and tool wear has been obtained by RMS vibration in the X dimension.

3.3.2 Results

This section shows the experimental results achieved in the database described in Section 3.3.1 with ANFIS models.



(a) Estimated RUL for PF.

(b) Estimated RUL for IPF.

(c) Estimated RUL for PFSDE.

(d) Estimated RUL for PFSDEWR.

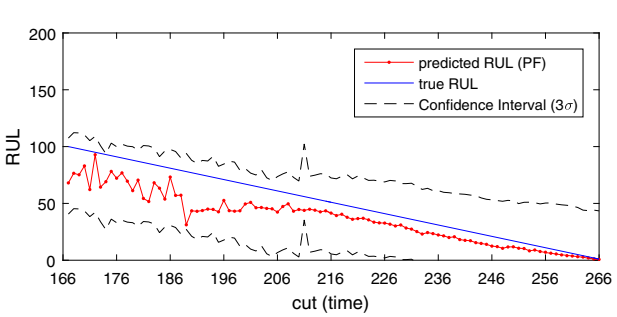
(e) Estimated RUL for SEF.

(f) Estimated RUL for PF-DE.

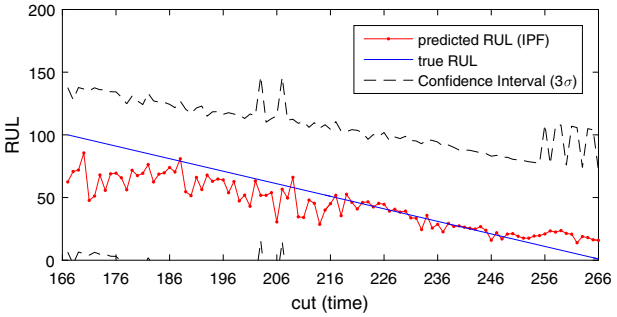
Fig. 12 Estimated RUL at $k = 166$ using of Np particles

Also thirty independent experiments are performed using multinomial resampling and initial conditions are $x \sim N(40, \sigma_x)$. The DE parameters are $F \sim U(0, 2)$ and $C = 0.5$. The IPF has been executed using $\alpha = 0.1$ and $p_m = 0.5$ and SEF has used $p_m = 0.5$ and $\lambda = 1$. The filter simulation parameters are 50 particles, time step number $N = 315$, $\sigma_x = 0.7$ and $\sigma_v = 0.05$, to process and measurement noises, respectively. Two datasets (c_1 and c_6) have been used to train the ANFIS models while c_4 has been used only for validation considering the unknown tool wear. Using the ANFIS models, the proposed filter PF-DE has been executed to estimate the wear tool progression. The fault occurs when the hidden state of the system, x_k, reaches a threshold x_{thr} defined by the user.

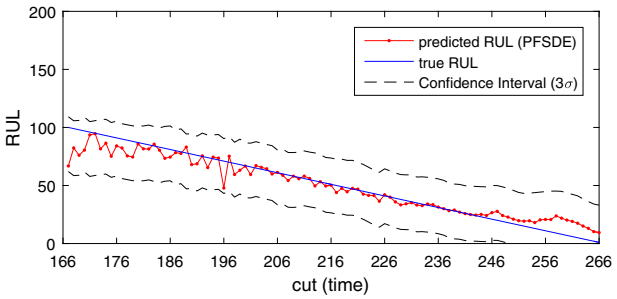
Two experiments are performed. Firstly, the filters monitored the CNC machine operation conditions up to the final instant. Secondly, the CNC machine is monitored from the instant when a prognostic estimation is performed, which happens at $k = 166$, until a threshold given by $x_{th} = 150$ is achieved, which occurs at instant $k = 266$. To evaluate the performance of the methods, the absolute error (AE) is computed for the first experiment and the accuracy metric of the prediction is calculated to the second experiment. Both results are presented in Table 7. Notice that accuracy is equal to 1 when the predicted and the true values are the same. Further, the accuracy function has a high decreasing rate when the predicted RUL is around the true RUL which gives a high sensitivity to the metric.



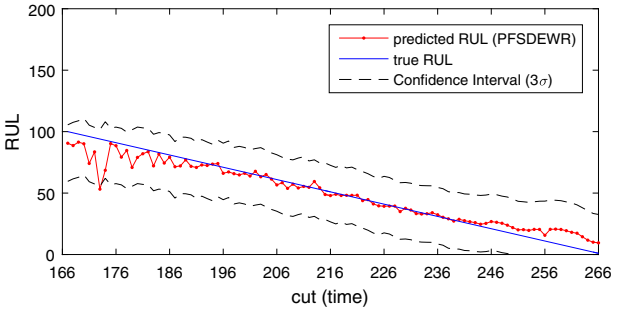
(a) Estimated and true RUL for PF.



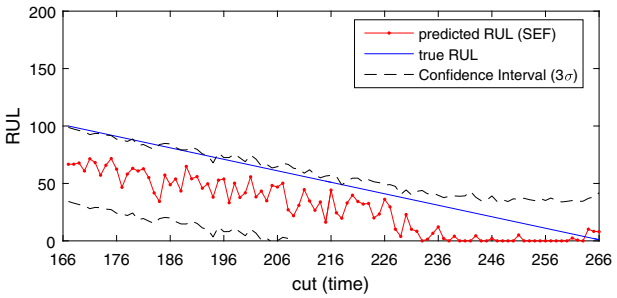
(b) Estimated and true RUL for IPF.



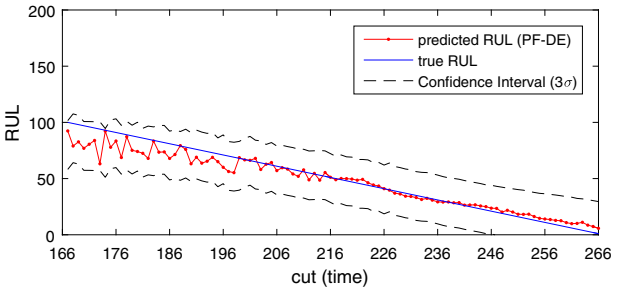
(c) Estimated and true RUL for PFSDE.



(d) Estimated and true RUL for PFSDEWR.



(e) Estimated and true RUL for SEF.



(f) Estimated and RUL for PF-DE.

Fig. 13 Estimated and true RUL for the particle filters

Table 8 presents the p-values obtained for all five comparisons concerning PF-DE: PF versus PF-DE (PF/PF-DE), IPF versus PF-DE (IPF/PF-DE), PFSDE versus PF-DE (SDE/PF-DE), PFSDEWR versus PF-DE (RW/PF-DE) and SEF versus PF-DE (SEF/PF-DE). As shown in Table 8, at level of significance $\alpha = 0.05$ the differences between PF-DE and the other methods for AE and AC are statistically significant.

Considering the case study, clearly the proposed method PF-DE presents a better estimation and prediction performance than the other methods considering the metrics in Table 7. The forecasting based on the propagation state transition has been performed without measurements and the uncertainty of RUL has been measured as in [31], which approximates the probability distribution by particles set. Figure 11 depicts the state estimation until $k = 166$ and predicted values after that until the end. Figure 12 presents the histogram for $Np = 50$ particles and the predicted RUL at $k = 166$ for each one of the six approaches.

Figure 13 shows the RUL prediction results for different starting points which vary from $k = 166$ to $k = 266$. It means that the true RUL is 100 cuts at instant $k = 166$ and 90 cuts at instant $k = 176$. The filter continues to monitor the wear condition and projecting the RUL successively. The red line indicates the predicted RUL, the blue line indicates the true RUL and the dotted line is the confidence interval which has been computed by a three sigma rule. Notice that the prediction estimations are depicted in Fig. 13 for the six approaches. The proposed filter achieved better RUL and smaller confidence interval than others.

As indicated by the results obtained in the experiments, it is clear that the traditional DE without any specific strategy applied to the FP is not able to reach good estimates in all the cases shown. A possible solution for this problem is to increase the number of generations within the DE, as present in the literature, which implies extra computational effort. However, the present work can obtain better estimates than the other methods with only a single generation, as it is adopted by the IPF and SEF. Compared to the GA methods (IPF and SEF) the results also confirm the best ability of the DE to enlarge the search space. In the specific case of the PF, the DE vectors of different amplitudes and direction generate better solutions around the regions of high likelihood than GA. Finally, PFSDERW obtained the worst estimates in all cases, due to the withdrawal of the PF resampling step without the use of any additional strategy.

4 Conclusions

This paper presents an improved modified particle filter (PF) for health state monitoring and fault prognostic purposes. The proposed PF-DE can mitigate the impoverishment problem found in this class of approach by increasing the diversity of the particles. The results for unidimensional and multidimensional experiments indicate that the state estimation has been improved when compared with the others evolutionary filters and standard PF if one considers underestimated noises. The effectiveness of the proposed particle filter for health monitoring and fault prognosis is illustrated in two practical problems: the estimation of bearings degradation in PRONOSTIA data sets and the prediction of tool wear conditions of a CNC machine. The unknown degradation has been predicted from the available data sensors and it is shown that the method may be able to perform health condition and fault prognostics in real time.

Acknowledgements This work was supported by the Brazilian National Research Council (CNPq) and the Research Foundation of the State of Minas Gerais (FAPEMIG), Brazil.

References

1. Ali M, Ahn CW, Siarry P (2014) Differential evolution algorithm for the selection of optimal scaling factors in image watermarking. Eng Appl Artif Intell 31:15–26. doi:10.1016/j.engappai.2013.07.009. Special Issue: Advances in Evolutionary Optimization Based Image Processing
2. Arulampalam M, Maskell S, Gordon N, Clapp T (2002) A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans Signal Process 50(2):174–188
3. Chen C, Vachtsevanos G, Orchard ME (2012) Machine remaining useful life prediction: an integrated adaptive neuro-fuzzy and high-order particle filtering approach. Mech Syst Signal Process 28:597–607
4. Chen C, Zhang B, Vachtsevanos G, Orchard M (2011) Machine condition prediction based on adaptive neuro-fuzzy and high-order particle filtering. IEEE Trans Ind Electron 58(9):4353–4364. doi:10.1109/TIE.2010.2098369
5. Chen Z (2003) Bayesian filtering: from kalman filters to particle filters, and beyond. Tech. rep., McMaster University. http://www.dsi.unifi.it/users/chisci/idfric/Nonlinear_filtering_Chen.pdf
6. Das S, Suganthan PN (2011) Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31
7. Douc R, Cappé O (2005) Comparison of resampling schemes for particle filtering. In: Proceedings of the 4th international symposium on image and signal processing and analysis, 2005. ISPA 2005. IEEE, pp 64–69
8. Doucet A, Johansen AM (2009) A tutorial on particle filtering and smoothing: fifteen years later. Handb Nonlinear Filter 12(656–704):3
9. Guo H, Li Y, Liu X, Li Y, Sun H (2016) An enhanced self-adaptive differential evolution based on simulated annealing for rule extraction and its application in recognizing oil reservoir. Appl Intell 44(2):414–436
10. Gustafsson F (2010) Particle filter theory and practice with positioning applications. IEEE Aerosp Electron Syst Mag 25(7):53–82
11. Higuchi T (1997) Monte carlo filter using the genetic algorithm operators. J Stat Comput Simul 59(1):1–23
12. Jang JSR, Sun CT (1997) Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Prentice-Hall, Inc., Upper Saddle River

13. Jardine AK, Lin D, Banjevic D (2006) A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech Syst Signal Process 20(7):1483–1510
14. Jouin M, Gouriveau R, Hissel D, Péra MC, Zerhouni N (2016) Particle filter-based prognostics: review, discussion and perspectives. Mech Syst. Signal Process 72–73:2–31
15. Kordestani JK, Ahmadi A, Meybodi MR (2014) An improved differential evolution algorithm using learning automata and population topologies. Appl Intell 41(4):1150–1169
16. Kwok N, Fang G, Zhou W (2005) Evolutionary particle filter: re-sampling from the genetic algorithm perspective. In: Proceedings of the IEEE/RSJ international conference on intelligent and robotic systems, pp 2935–2940
17. Li D, Zhou Y (2015) Combining differential evolution with particle filtering for articulated hand tracking from single depth images. Int J Signal Process Image Process Pattern Recognit 8(4):237–248
18. Li HW, Wang J, Su HT (2011) Improved particle filter based on differential evolution. Electron Lett 47(19):1078–1079
19. Liao L (2014) Discovering prognostic features using genetic programming in remaining useful life prediction. IEEE Trans Ind Electron 61(5):2464–2472. doi:10.1109/TIE.2013.2270212
20. Liao L (2014) Discovering prognostic features using genetic programming in remaining useful life prediction. IEEE Trans Ind Electron 61(5):2464–2472
21. Liao L, Köttig F (2016) A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction. Appl Soft Comput 44:191–199. doi:10.1016/j.asoc.2016.03.013. http://www.sciencedirect.com/science/article/pii/S1568494616301223
22. Nectoux P, Gouriveau R, Medjaher K, Ramasso E, Chebel-Morello B, Zerhouni N, Varnier C (2012) Pronostia: an experimental platform for bearings accelerated degradation tests. In: IEEE international conference on prognostics and health management, PHM'12, pp 1–8. IEEE Catalog Number: CPF12PHM-CDR
23. Orchard ME, Hevia-Koch P, Zhang B, Tang L (2013) Risk measures for particle-filtering-based state-of-charge prognosis in lithium-ion batteries. IEEE Trans Ind Electron 60(11):5260–5269. doi:10.1109/TIE.2012.2224079
24. Orchard ME, Vachtsevanos GJ (2009) A particle-filtering approach for on-line fault diagnosis and failure prognosis. Trans Inst Meas Control 31:221–246
25. PHM Society (2010) Conference data challenge. https://www.phmsociety.org/competition/phm/10. Accessed 11 July 2016
26. PHM Society (2010) Data challenge. https://www.phmsociety.org/. Accessed 11 July 2016
27. Sikorska J, Hodkiewicz M, Ma L (2011) Prognostic modelling options for remaining useful life estimation by industry. Mech Syst Signal Process 25(5):1803–1836
28. Simon D (2006) Optimal state estimation, Kalman, H_∞, and nonlinear approaches. Wiley-Interscience
29. Stron R, Price K (1996) Minimizing the real functions of the icec'96 contest by differential evolution. In: Proceedings of the IEEE international conference on evolutionary computation, pp 842–844
30. Stron R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11:341–359
31. Sun J, Zuo H, Wang W, Pecht MG (2014) Prognostics uncertainty reduction by fusing on-line monitoring data based on a state-space-based degradation model. Mech Syst Signal Process 45(2):396–407
32. Vachtsevanos G, Lewis F, Roemer M, Hess A, Wu B (2006) Intelligent fault diagnosis and prognosis for engineering systems. Wiley, New York
33. Vasan ASS, Long B, Pecht M (2013) Diagnostics and prognostics method for analog electronic circuits. IEEE Trans Ind Electron 60(11):5277–5291. doi:10.1109/TIE.2012.2224074
34. Wang Y, Qi Y (2013) Memory-based cognitive modeling for robust object extraction and tracking. Appl Intell 39(3):614–629
35. Wilcoxon F (1945) Individual comparisons by ranking methods. Biom Bull 1(6):80–83
36. Yan W, Zhang B, Wang X, Dou W, Wang J (2016) Lebesgue-sampling-based diagnosis and prognosis for lithium-ion batteries. IEEE Trans Ind Electron 63(3):1804–1812. doi:10.1109/TIE.2015.2494529
37. Yin S, Zhu X (2015) Intelligent particle filter and its application to fault detection of nonlinear system. IEEE Trans Ind Electron 62(6):3852–3861
38. Yin S, Zhu X, Qiu J, Gao H (2016) State estimation in nonlinear system using sequential evolutionary filter. IEEE Trans Ind Electron 63(6):3786–3794
39. Yu M, Wang D, Luo M (2015) An integrated approach to prognosis of hybrid systems with unknown mode changes. IEEE Trans Ind Electron 62(1):503–515. doi:10.1109/TIE.2014.2327557
40. Zhao Z, Wang J, Tian Q, Cao M (2010) Particle swarm-differential evolution cooperative optimized particle filter. In: 2010 international conference on intelligent control and information processing, pp 485–490
41. Zou D, Liu H, Gao L, Li S (2011) An improved differential evolution algorithm for the task assignment problem. Eng Appl Artif Intell 24(4):616–624. doi:10.1016/j.engappai.2010.12.002

Luciana Balieiro Cosme received the B.S. degree in information systems from University of Montes Claros, Brazil, in 2006, and the M.S. degree in electrical engineering from the University of Minas Gerais, Brazil, in 2011. She is currently a Ph.D. student from the University of Minas Gerais, Brazil. She is also a professor at the Federal Institute of Education, Science and Technology of Norte de Minas Gerais (IFNMG) Campus Montes Claros, Brazil, since 2010. Her research interests include computational intelligence and fault prognostic.

Marcos Flávio S. V. D'Angelo joined the State University of Montes Claros in 2000 as an Associate Professor in Information Science. He received his Ph.D. degree in Electrical Engineering from the Federal University of Minas Gerais in 2010. In a broad sense, D'Angelo main research interests include dynamic systems and optimization theory/applications, and soft computing.

Walmir Caminhas received the B.Sc. degree in electrical engineering from the Federal University of Minas Gerais in 1987, the M.Sc. degree in electric engineering from the Federal University of Minas Gerais in 1989, and the PhD degree in electric engineering from the University of Campinas in 1997. He is professor of the Department of Electronics Engineering, Federal University of Minas Gerais, since 1989. His interest areas include computational intelligence, fault detection and diagnosis, and nonlinear systems modeling.

Shen Yin received the B.E. degree in automation from the Harbin Institute of Technology, Harbin, China, in 2004, and the M.Sc. degree in control and information systems and the Ph.D. Degree in electrical engineering and information technology from the University of Duisburg–Essen, Essen, Germany, in 2007 and 2012, respectively.

He is currently a Full Professor with the School of Astronautics, Harbin Institute of Technology. His current research interests include model-based and data-driven fault diagnosis, fault-tolerant control, and big data focused on industrial electronics applications.

Reinaldo Martinez Palhares is currently a professor at the Department of Electronics Engineering, University of Minas Gerais, Brazil. Palhares' main research interests include robust linear/nonlinear control theory; fault detection, diagnosis and prognostic; and soft computing. Palhares has been serving as an Associate Editor for IEEE Transactions on Industrial Electronics and Guest Editor of the Journal of The Franklin Institute Special Section on Recent Advances on Control and Diagnosis via Process measurements.