Optimal tuning of particle filtering random noise for monotonic degradation processes

Matteo Corbetta[1], Claudio Sbarufatti[2], and Marco Giglio[3]

[1,2,3] *Politecnico di Milano, Dipartimento di Meccanica, via La Masa 1, Milan, 20156, Italy*
matteo.corbetta@polimi.it
claudio.sbarufatti@polimi.it
marco.giglio@polimi.it

ABSTRACT

Particle filtering is a model-based, Bayesian filtering algorithm widely employed in many scientific and engineering fields. It has been recently applied many times for the diagnosis and prognosis of engineering systems. Within the prognostics and health management scenario, particle filtering stood out as an effective and robust algorithm to predict the system's remaining useful life because of its capability in tracking nonlinear/non-Gaussian systems. One of the fundamental equations underneath particle filtering is the evolution equation describing the system's dynamics. This equation composes of a deterministic model and an artificially-added random process or random noise, thus making the evolution equation a stochastic equation. The selection of random noise is up to the algorithm's designer discretion and may vary on a case-by-case basis. Concentrating on the field of structural degradation processes, many studies on particle filtering-based prognostics have shown encouraging results. Though, they did not provide detailed discussions in supporting the appropriateness of the selected random noises altering the degradation model. An improper choice may cause the algorithm to be inefficient, moving the projected trajectories outside the state-space domain of the system, sometimes also introducing a bias in the stochastic evolution model. Therefore, this work examines the evolution equation with the aim of creating an optimal prognostic framework for monotonic degradation phenomena. The paper gives special emphasis to structural degradation caused by fatigue, which is a widespread monotonic damage progression process. Some of the existing works are reviewed, discussing the effectiveness of the random noises embedded in the algorithm. Eventually, the paper presents an unbiased, optimal random noise for monotonic damage progression, pointing out the strengths of the proposed solution against formulations suggested in literature. The presented particle filtering-based prognostic algorithm is applied to experimental crack growth observations on an aeronautical stiffened structure and the prediction performance is validated using dedicated prognostic metrics.

1. INTRODUCTION

A key task of prognostics and health management (PHM) is the estimation of the remaining useful life (RUL) of engineering systems. The RUL estimation deals with predicting future degradations that include, inherently, a certain degree of uncertainty (Sankararaman & Goebel, 2015). Different sources of uncertainty characterize the RUL prediction. These include, but they are not limited to, the uncertainty of the knowledge of the current system's condition, the modeling uncertainty and the uncertainty of future loading conditions (Sankararaman, 2015). In this context, a sequential Monte Carlo method known as *particle filtering* has proven its ability to monitor and predict the evolution of nonlinear, non-Gaussian processes by filtering the uncertainties affecting the systems' dynamics (Gordon, Salmond, & Smith, 1993). It is a model-based, Bayesian filter that enables the estimation of the conditional probability density function (pdf) of the system's state given a set of observations, called posterior pdf (according to the Bayes' nomenclature). Starting from the posterior pdf, the estimated system's state can be projected in the future using a series of Monte Carlo samples to calculate the RUL.

Particle filtering grounds on two equations called *evolution* (or *process*) equation and *observation* equation. The first models the system's state dynamics, while the second links the observations (obtained with a measurement system) with the system's state. Usually, deterministic models are the bases of the evolution and observation equations, and error terms add to such models to generate stochastic equations. These error terms describe: (i) approximations caused by the modeling approach, and (ii) uncertainties of the measurement system. They are typically represented by random processes

called (i) *process noise*, which adds to the evolution equation, and (ii) *measurement noise*, which adds to the observation equation.

The strengths of particle filtering are extensively tested in diagnostics and prognostics, insomuch as it now is considered a state-of-the-art technique (Jouin, Gouriveau, Hissel, Péra, & Zerhouni, 2015). The PHM domain provides many successful examples of particle filtering-based algorithms for prediction of the systems' degradation caused by fatigue damage accumulation (Cadini, Zio, & Avram, 2009b, 2009a; M. E. Orchard & Vachtsevanos, 2009; Baraldi, Compare, Sauco, & Zio, 2013; J. Chiachıo, Chiachıo, Saxena, Rus, & Goebel, 2013). The use of particle filtering is supported by its ability to deal with highly nonlinear systems and non-Gaussian pdfs, so non-Gaussian noises. The modeling of the measurement noise depends on the type of measurement system and measurement procedure, so it can be quantified and modeled consistently with the precision of the instrumentation. Instead, the process noise is mostly a choice of the algorithm's designer. Existing works use additive Gaussian noises to add disturbances to the evolution equation. This choice is often justified by the absence of quantitative, precise information on the type and the amount of uncertainty affecting the degradation process. Though, particle filtering-based predictions using a log-Normal process noise were presented in some papers. The use of log-Normal random processes was supported by probabilistic fracture mechanics literature. Despite these explanations, the effect of such process noises on the evolution equation was not investigated.

With this in mind, this work discusses the selection of particle filtering process noise to monitor and predict monotonic degradation processes. The paper focuses on fatigue damage accumulation, which is monotonic in nature. Though, the discussion can be further extended to other degradation processes that undergo the hypothesis of monotonic behavior. In this scenario, the effect of the process noise is analytically examined using the algebra of random variables and the conditional expected value of the system's state. The analysis underlines that the process noise strongly affects the exploration of the state-space. Its effect is notable in the prognostic stage, when the system's state is projected several steps ahead in the future. The review of the existing works emphasizes that:

- the use of an additive Gaussian noise may compromise the efficiency of the algorithm, and
- the log-Normal process already utilized in several papers produces a *biased* projection of the samples in the state-space.

Then, the paper proposes a log-Normal random process with specific relation between mean and variance. The suggested formulation solves the issues provoked by process noises adopted in literature so far. Eventually, the selected process noise is used in a particle filtering algorithm for combined state-parameter estimation. This algorithm efficiently predicts the RUL of a cracked aeronautical stiffened panel subject to tension-tension fatigue in a laboratory environment. The appropriateness of the selected process noise is assessed against previous, existing formulations using dedicated prognostic performance metrics.

2. SUMMARY OF PARTICLE FILTERING

The objective of the filtering problem is to recursively estimate the state of a system governed by a dynamic state-space (DSS) model (Haug, 2005). The DSS model composes of the evolution equation, $f(\cdot)$, describing the system's dynamics, and the observation equation, $g(\cdot)$, which links the measurements with the true (hidden) system's state. Equation (1) shows the discrete form of the DSS model, which satisfies the first order Markovian assumption (Arulampalam, Maskell, Gordon, & Clapp, 2002).

$$\begin{aligned} \boldsymbol{x}_k &= f\left(\boldsymbol{x}_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}, \boldsymbol{\omega}_{k-1}\right) \\ \boldsymbol{z}_k &= g(\boldsymbol{x}_k, \boldsymbol{\eta}_k) \end{aligned} \tag{1}$$

The vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T \in \mathcal{D} \subseteq \mathbb{R}^{n\times 1}$ collects the system's state variables, $\boldsymbol{z} = [z_1, z_2, \ldots, z_m]^T \in \mathcal{D}_{\boldsymbol{z}} \subseteq \mathbb{R}^{m\times 1}$ is the observation vector and the subscript k indicates the discrete *k-th* time step. The state-space domain $\mathcal{D}$ is the physical domain of the system's state variables, often represented by a partition of the set $\mathbb{R}^{n\times 1}$. The evolution function depends on the input $\boldsymbol{u}$, the model parameters $\boldsymbol{\theta}$ and the process noise, $\boldsymbol{\omega}$. The measurement system is governed by $g(\cdot)$ and affected by the measurement noise, $\boldsymbol{\eta}$. As already stated in the introduction, the noises are random processes transforming the deterministic equations into stochastic equations.

Particle filtering aims at estimating the posterior pdf of $\boldsymbol{x}_k$ given the sequence of noisy observations $\boldsymbol{z}_{0:k}$, $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$, in case of nonlinear and non-Gaussian systems. The posterior pdf can be approximated by N_s weighted samples (also called *particles*) of the system's state, as expressed in Eqs. (2)-(3) (Doucet, Godsill, & Andrieu, 2000; Arulampalam et al., 2002; Haug, 2005).

$$\hat{p}(\boldsymbol{x}_k|\boldsymbol{z}_{0:k}) = \sum_{i=1}^{N_s} w_k^{(i)} \delta_{\boldsymbol{x}_k^{(i)}, \boldsymbol{x}_k} \tag{2}$$

$$\begin{aligned} \tilde{w}_k^{(i)} &= w_{k-1}^{(i)} p(\boldsymbol{z}_k|\boldsymbol{x}_k^{(i)}) \\ w_k^{(i)} &= \frac{\tilde{w}_k^{(i)}}{\sum_{j=1}^{N_s} \tilde{w}_k^{(j)}} \end{aligned} \tag{3}$$

Where: $\hat{p}(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$ is the approximation of $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$, $\boldsymbol{x}_k^{(i)}$ is the *i-th* sample of the system's state vector, $w_k^{(i)}$ is the normalized weight of $\boldsymbol{x}_k^{(i)}$, $p(\boldsymbol{z}_k|\boldsymbol{x}_k^{(i)})$ is the likelihood of the observation given $\boldsymbol{x}_k^{(i)}$ and $\delta_{i,j}$ is the Kronecker delta. The number of samples N_s is supposed to be large enough to describe the (unknown) true shape of $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$. It should be noted that Eqs. (2)-(3) refer to the *bootstrap* particle filter (Arulampalam et al., 2002).

The particle filtering-based prognosis is carried out by projecting the samples $\boldsymbol{x}_k^{(i)}; i = 1, \ldots, N_s$ many steps ahead in the future using Eq. (4), which is the *p-step ahead* prediction equation (Doucet et al., 2000).

$$\hat{p}(\boldsymbol{x}_{k+p}|\boldsymbol{z}_{0:k}) = \sum_{i=1}^{N_s} w_k^{(i)} \int_{\mathcal{D}} p\left(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k^{(i)}\right) \prod_{j=k+2}^{k+p} p(\boldsymbol{x}_j|\boldsymbol{x}_{j-1}) \mathrm{d}\boldsymbol{x}_{k+1:k+p-1} \tag{4}$$

The practical implementation of the algorithm requires the definition of two fundamental functions: the *transition density function* (tdf), $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$, and the likelihood function, $p(\boldsymbol{z}_k|\boldsymbol{x}_k)$. Both of them come from the probabilistic form of the DSS model, Eq. (5), (Jouin et al., 2015).

$$\begin{aligned} \boldsymbol{x}_k &= f\left(\boldsymbol{x}_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}, \boldsymbol{\omega}_{k-1}\right) \rightarrow p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) \\ \boldsymbol{z}_k &= g(\boldsymbol{x}_k, \boldsymbol{\eta}_k) \rightarrow p(\boldsymbol{z}_k|\boldsymbol{x}_k) \end{aligned} \tag{5}$$

The tdf drives the generation of the samples of $\boldsymbol{x}_k^{(i)}$ and the p-step ahead prediction in Eq. (4). Therefore, an improper selection of the process noise (that transforms the deterministic model in $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$) may compromise the efficiency of particle filtering. The generation of the tdf through $\boldsymbol{\omega}$ is discussed in Section 3.

3. ON THE SELECTION OF THE PROCESS NOISE FOR FATIGUE DAMAGE ACCUMULATION

Without loss of generality, let us consider a uni-dimensional system's state $\boldsymbol{x}_k \rightarrow x_k \in \mathcal{D} \subseteq \mathbb{R}^{1\times 1}$, a deterministic model parameter vector $\boldsymbol{\theta}$ and a deterministic input vector $\boldsymbol{u}$. The damage extent x_k increases with time, and the randomization of the evolution equation relies only on ω. The versatility of Monte Carlo sampling permits the use of any kind of process noise. However, it is desirable to build a tdf that is consistent with the true damage progression: the particles generated with the tdf should be representative of a potential damage growth. Consequently, the development of the tdf should satisfy the following requirements:

- The particles should remain inside the physical domain or *support* of the system's state variable $\mathcal{D}$. Otherwise, the algorithm would waist computational effort propagating samples that cannot describe the real system.
- Each particle should represent a possible damage progression path. It is well-known that fatigue damage accumulation is a monotonic phenomenon. Once the damage is nucleated in the material, it can only increase with time. Therefore, each particle $x_k^{(i)}$ must behave consistently with the physical process.
- The process noise must not change the trend of the evolution equation. It should introduce only a random disturbance, unless the algorithm's designer is aware that is altering, on purpose, the deterministic trend.

The three conditions above can be mathematically expressed through Eqs. (6)-(8).

$$x_k^{(i)} \in \mathcal{D} \subseteq \mathbb{R} \quad \forall\ k \in \mathbb{N}, i = 1, \ldots, N_s \tag{6}$$

$$x_k^{(i)} \geq x_{k-1}^{(i)} \quad \forall\ k \in \mathbb{N}, i = 1, \ldots, N_s \tag{7}$$

$$\begin{aligned} \mathrm{E}[x_k|x_{k-1}] &= \mathrm{E}[f(x_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}, \omega_{k-1})] \\ &= f'(x_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}), \quad \forall\ k \in \mathbb{N} \end{aligned} \tag{8}$$

Where $\mathrm{E}[x_k|x_{k-1}]$ is the conditional expected value of x_k and $f'(\cdot)$ is the deterministic evolution equation, i.e., the evolution equation without any process noise. It should be noted that the monotonicity is represented by the constraint in Eq. (7), while Eqs. (6) and (8) should be satisfied for any kind of dynamic process.

Typical examples of damage accumulation processes that should be modeled like monotonic phenomena are: fatigue crack growth (FCG) in metallic alloys (Corbetta, Sbarufatti, Manes, & Giglio, 2014), matrix crack density and delamination in fiber-reinforced laminates (Corbetta, Saxena, Giglio, & Goebel, 2015) and creep-induced plastic strains (Baraldi, Mangili, & Zio, 2012). These damages cannot fall below zero, so the support of the system's state is the positive subset of real numbers, $\mathcal{D} \subseteq \mathbb{R}_{[0,+\infty]}$. Moreover, these damages are monotonic; they increase with the number of load cycles (or time, in case of creep-induced degradation). Commonly, these types of fatigue degradation processes are modeled using linear damage accumulation models that ground on the damage growth rate, Eq. (9).

$$x_k = x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{k-1} \Delta N \tag{9}$$

Where $\mathrm{d}x/\mathrm{d}N$ is the damage growth rate per load cycle, and ΔN is the number of load cycles between *k-1* and *k* (however, creep degradation is expressed against time). The modeling

of the damage growth rate usually resorts on power laws like *Paris' laws* or *modified Paris' laws*, Eq. (10).

$$\frac{\mathrm{d}x}{\mathrm{d}N} = \theta_1 h(x)^{\theta_2} \tag{10}$$

Where $\theta_j,\ j = 1, 2$ are empirical parameters depending on the material and $h(\cdot)$ is a function of the damage extent. The damage growth rates of cracks, micro-cracks, delamination, creep, etc. are always positive: $\frac{\mathrm{d}x}{\mathrm{d}N} \geq x,\ \forall\, x \in \mathcal{D}$. Therefore, they force the linear damage accumulation model in Eq. (9) to be monotonic, $x_k \geq x_{k-1},\ \forall\, k \in \mathbb{N}$.

Equations (9) and (10) form the deterministic evolution equation $f'(\cdot)$ for fatigue damage accumulation, and the process noise ω transforms $f'(\cdot)$ in $f(\cdot)$. Thereupon, the derivation of the tdf from $f(\cdot)$ is straightforward. The approaches to add the process noise ω presented in literature are examined below, and the consequent tdf is derived. All of them ground on damage growth rates similar to the Paris' law in Eq. (10), and the discussion shows that they may fail one or more requirements expressed through eqs. (6)-(8): the capability to keep the particles in the domain $\mathcal{D}$, the monotonicity of the particles, or the process noise-induced bias. Then, a process noise that satisfies all the requirements is proposed.

3.1. Additive Gaussian process noise

The simplest approach to generate a tdf is to add a Gaussian process noise to the deterministic evolution equation to generate $f(\cdot)$, Eq. (11).

$$x_k = x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N + \omega \tag{11}$$

Where $\omega \sim \mathcal{N}(\mu_\omega, \sigma_\omega^2)$. The process noise has been assumed stationary, so the dependence of ω on the time step has been neglected. Thus, the tdf becomes a Normal distribution (12), and the propagation of the particles follows Eq. (13).

$$p(x_k|x_{k-1}) = \frac{1}{\sqrt{2\pi}\sigma_\omega} \exp\left\{ -\frac{[x_k - (f'(x_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}) + \mu_\omega)]^2}{2\sigma_\omega^2} \right\} \tag{12}$$

$$x_k^{(i)} = x_{k-1}^{(i)} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}^{(i)}} \Delta N + \omega^{(i)} \tag{13}$$

Where $\omega^{(i)}$ is the *i-th* sample from $\mathcal{N}(\mu_\omega, \sigma_\omega^2)$. Additive Gaussian noises were used in many papers: (M. E. Orchard & Vachtsevanos, 2007), (M. Orchard, Kacprzynski, Goebel, Saha, & Vachtsevanos, 2008), (M. E. Orchard & Vachtsevanos, 2009), (Tang, DeCastro, Kacprzynski, Goebel, & Vachtsevanos, 2010), (J. Chiachıo et al., 2013) and (M. Chiachıo, Chiachıo, Saxena, Rus, & Goebel, 2014). They were mostly designed as zero-mean noises, $\omega \sim \mathcal{N}(0, \sigma_\omega^2)$, thus generating an unbiased evolution equation. So, the expected value of the samples $x_k^{(i)},\ i = 1, \ldots, N_s$ is equal to the deterministic damage evolution, Eq. (14), satisfying the condition in Eq. (8).

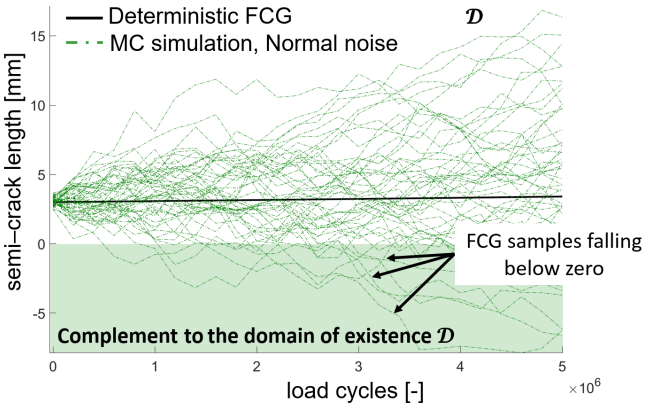

Figure 1. Prognostic stage of a 4 mm semi-crack length.

$$\begin{aligned} \mathrm{E}[x_k|x_{k-1}] &= x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} + \mathrm{E}[\omega] \\ &= x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} + 0 \\ &= f'(x_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}) \end{aligned} \tag{14}$$

Since $\omega^{(i)}$ adds to the deterministic equation, it may cause the sample $x_k^{(i)}$ to be smaller than $x_{k-1}^{(i)}$, thus generating particles with a decreasing trend. If this happens for a sufficient number of time steps, some particles will fall below zero. Then, the use of an additive Gaussian noise may fail two requirements: the monotonicity of the particles and the correct exploration of the state-space (since the particles fall outside $\mathcal{D}$).

Figure 1 shows an example of particles falling outside from the system's state support. A simulated FCG from the center of a metallic plate subject to fatigue load has been used as case study. The FCG has been simulated using Eqs. (9) and (10). The damage growth rate has been modeled using a Paris' law for Aluminum alloys and an analytical model for the stress intensity factor. The features to simulate the FCG are reported in table 1. This simulation is representative of the prognostic stage, when the posterior pdf $p(x_k|z_{0:k})$ has been already computed and the prediction of the RUL is carried out by propagating the samples $x_k^{(i)}$ using the tdf.

As visible from figure 1, the step-by-step simulation makes

Table 1. Initialization of Monte Carlo simulation of FCG.

number of samples	100
initial crack length	$x_0 \sim \mathcal{N}(2, 0.1), [mm]$
applied far-field stress	$\Delta S = 20\text{MPa}$
load ratio	$R = 0$
Paris' law parameters	$\theta_1 = C = 1.1994e-14$
	$\theta_2 = m = 3.79$
stress intensity factor	$h(x) = \Delta K(x) = F\Delta S\sqrt{\pi x}$
geometry factor	$F = 1$
process noise variance	$\sigma_\omega^2 = 1$

the particles fall outside $\mathcal{D}$. The longer the prognostic stage, the higher the number of samples may fall outside state-space domain. This behavior compromises the calculation of the RUL, since the samples that fall below zero will never reach the critical threshold (usually defined as a critical damage size). The algorithm's designer may reduce the variance of the process noise to overcome the problem. Nonetheless, two main issues arise:

- the procedure to select the variance would become strongly case-dependent: the initial distribution of the damage extent and the closeness of x_0 to 0 would affect the selection of the variance, diminishing the general validity of the algorithm, and
- if the variance became too small, the randomization effect would become negligible.

Another solution is the selection of a positive process noise mean $\mu_\omega > 0$ using historical data, as presented in (M. E. Orchard & Vachtsevanos, 2009) and (Baraldi et al., 2013), which adjusts the particles' trend. In this manner, a positive μ_ω may keep the particles within $\mathcal{D}$. Though, historical data may not be available for the component that has to be monitored, and this solution still has some drawbacks. First, the particles still do not fulfill the monotonicity requirement producing unlikely damage progression paths. In addition, a positive value of μ_ω would produce a faster damage growth with respect to the deterministic equation, supposing that the future damage growth will be faster than the expected one. This conclusion cannot be drawn during the tuning of the filter, before that the damage actually starts propagating.

3.2. Multiplicative non-Gaussian process noise

Another type of process noise from fracture mechanics theory was proposed. A log-Normal random process e^ω, $\omega \sim \mathcal{N}(0, \sigma_\omega^2)$ was multiplied to the FCG rate to satisfy the monotonicity requirement, Eq. (15).

$$x_k = x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N\, e^\omega \tag{15}$$

In this case, the tdf and the propagation of the particles follow Eqs. (16) and (17), respectively.

$$p(x_k|x_{k-1}) = \frac{1}{(x_k - x_{k-1})\sigma_\omega\sqrt{2\pi}} \exp\left\{-\frac{[\log(x_k - x_{k-1}) - \tilde{\mu}]^2}{2\sigma_\omega^2}\right\} \tag{16}$$

$$x_k^{(i)} = x_{k-1}^{(i)} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}^{(i)}} \Delta N e^{\omega^{(i)}} \tag{17}$$

Where Eq. (16) is a log-Normal distribution with shift parameter x_{k-1}, and $\tilde{\mu} = \mu_\omega + \log(\left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N)$. Since the log-Normal distribution is defined in the positive domain, it can be efficiently used to force each sample $x_k^{(i)}$ to stay within $\mathcal{D}$ and to increase with time.

Such a log-Normal process noise was used in (Cadini et al., 2009b), (Cadini et al., 2009a), (Zio & Peloni, 2011), (Yang, Yuan, Qiu, Zhang, & Ling, 2012) and was also used in (Zio & Di Maio, 2012), where a relevance vector machine was used to predict fatigue crack propagation. However, this randomization of the evolution equation introduces a bias in the tdf. This can be proved calculating the conditional expectation of x_k using the algebra of random variables, Eq. (18).

$$\begin{aligned}\mathrm{E}[x_k|x_{k-1}] &= x_{k-1} + \mathrm{E}\left[\left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N\, e^\omega\right] \\ &= x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N\, e^{\left(\mu_\omega + \frac{\sigma_\omega^2}{2}\right)}\end{aligned} \tag{18}$$

Equation (18) has been calculated exploiting the properties of the log-Normal distribution, which states that $c\,\omega \sim \log\mathcal{N}(\mu_\omega + \log(c), \sigma_\omega^2)$, where c is constant. As visible from Eq. (18), the mean and variance of the random process alter the expected value of the linear damage accumulation model. If $\omega \sim \mathcal{N}(0, \sigma_\omega^2)$ as made in previous, existing works, the bias between the deterministic evolution equation and $f(\cdot)$ can be quantified through Eq. (19).

$$\mathrm{E}[x_k|x_{k-1}] - f'(x_{k-1}, \boldsymbol{\theta}, \boldsymbol{u}_{k-1}) = \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N \left(e^{\left(\frac{\sigma_\omega^2}{2}\right)} - 1\right) \tag{19}$$

Then, the particles move away from the expected (deterministic) trend, and the distance increases as time passes by. This result is particularly important for the prognostic stage: the longer the prediction, the higher the difference between the deterministic equation and $f(\cdot)$. Also, the error is proportional to the noise variance σ_ω^2. The higher the variance, the higher the bias.

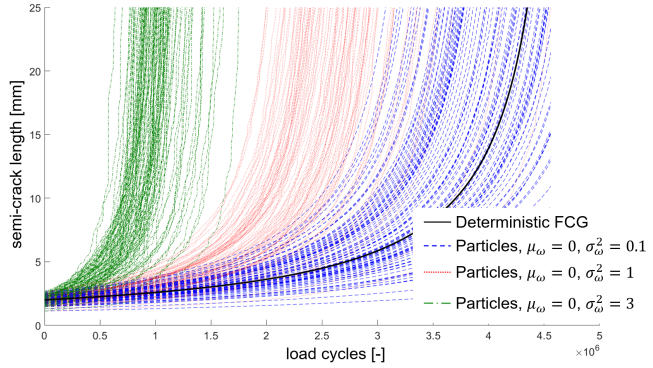

Figure 2. Particles swarm using $e^{\omega} \sim \mathcal{N}(0, \sigma_{\omega}^2)$ and different values of σ_{ω}^2.

Figure 2 presents the effect of the log-Normal process noise, showing the propagation of the samples using the tdf in (16). The case study refers to a FCG in a metallic plate, as made in Subsection 3.1, using $\Delta S = 45\,\mathrm{MPa}$. If the process noise variance is relatively small, the particles remain close to the deterministic equation. Yet, the difference between the deterministic equation and the trend of the particles increases with σ_{ω}^2 and time. It should be noted that the algorithm can provide satisfactory results even if the evolution equation is affected by a bias, provided that the biased swarm of particles will cover the state-space region where the future damage will growth. However, it is not possible to predict whether the future damage will growth slower or faster of the expected damage progression. Also, the algorithm's designer may need to increase (decrease) the process noise variance to increase (decrease) the particles' dispersion. If the evolution equation is biased, that procedure modifies also the trend of the particles, as clarified in Figure 2. Summarizing the discussion above, the process noise e^{ω}, $\omega \sim \mathcal{N}(0, \sigma_{\omega}^2)$ satisfies the monotonicity condition, produces a correct exploration of the state-space domain, but introduces a bias in the tdf, thus not fulfilling Eq. (8).

A modification of the log-Normal process noise able to meet the requirement in Eq. (8) is proposed in the next Subsection.

3.3. Optimal tuning of the process noise

The log-Normal random process presented in the previous section introduces a multiplicative term in the evolution equation, as already shown in Eq. (15), and the conditional expectation of x_k is affected by both μ_{ω} and σ_{ω}^2, Eq. (18). Since μ_{ω} and σ_{ω}^2 are choices of the algorithm's designer, the mean can be selected to ensure that $\mathrm{E}[e^{\omega}] = \exp(\mu_{\omega} + \sigma_{\omega}^2/2) = 1$, Eq. (20).

$$\mu_{\omega} = -\frac{\sigma_{\omega}^2}{2} \tag{20}$$

By so doing, the expected value of $f(\cdot)$ remains equal to the deterministic evolution equation regardless on the value of σ_{ω}^2, Eq. (21).

$$\begin{aligned} \mathrm{E}[x_k|x_{k-1}] &= x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N\, e^{\left(-\frac{\sigma_{\omega}^2}{2}+\frac{\sigma_{\omega}^2}{2}\right)} \\ &= x_{k-1} + \left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N\,;\ \forall\, \sigma_{\omega}^2 \in \mathbb{R}_{[0,+\infty)} \end{aligned} \tag{21}$$

The tdf is still a log-Normal distribution with shift parameter x_{k-1}, Eq. (22), and the generation of the samples still follows Eq. (17).

$$\begin{aligned} &p(x_k|x_{k-1}) = \\ &= \frac{1}{(x_k - x_{k-1})\sigma_{\omega}\sqrt{2\pi}} \exp\left\{-\frac{[\log(x_k - x_{k-1}) - \tilde{\mu}]^2}{2\sigma_{\omega}^2}\right\} \end{aligned} \tag{22}$$

Where $\tilde{\mu} = -\frac{\sigma_{\omega}^2}{2} + \log(\left.\frac{\mathrm{d}x}{\mathrm{d}N}\right|_{x_{k-1}} \Delta N)$. The proposed log-Normal process noise, which does not introduce any bias in the tdf, is defined *balanced log-Normal* process noise henceforth. Figure 3 shows the particles' propagation referring to the case study in Subsection 3.2. The particles have been propagated using the unbiased tdf and different values of σ_{ω}^2. They remain centered on the deterministic evolution equation regardless on the amount of perturbation introduced by e^{ω}.

This property is particularly useful during the development of the prognostic unit. In fact, the algorithm's designer may need to increase the particles' dispersion to explore the state-space correctly, or decrease the particles' dispersion if many samples fall in unlikely regions of the state-space. The use of the log-Normal random noise with $\mu_{\omega} = -\sigma_{\omega}^2/2$ permits to increase or decrease the randomization of the evolution equation always satisfying the requirements discussed in Section 2: (i) the particles remain in the state-space support $\mathcal{D}$, (ii) each particle increases over time, thus representing a potential fatigue damage accumulation path, and (iii) the swarm of samples remains centered on the deterministic trend regardless of the amount of noise. The latter allows the user to increase or decrease the process noise without introducing any bias in the swarm of particles. If the user needs to modify the particles' trend, he may resort on combined state-parameter estimation methods (Liu & West, 2001). In this way, the model parameters can be updated during the run-time to adjust the particles' trend. Thus, the trend of the particles and their dispersion can be adjusted independently to each other.

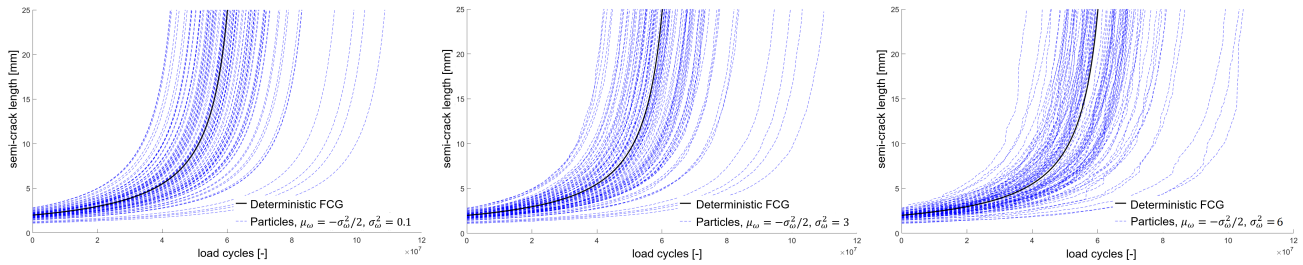

Figure 3. Particles swarm using e^{ω}, $\omega \sim \mathcal{N}(-\frac{\sigma_\omega^2}{2}, \sigma_\omega^2)$ and different values of σ_ω^2: 0.1 (left), 3 (center) and 6 (right).

4. APPLICATION TO REAL FATIGUE CRACK PROPAGATION DATA

This section shows the application of particle filtering to experimental observations of FCG in an Aluminum helicopter panel. The experimental activity was conducted in a representative, but rather simplified, laboratory environment. The aim of the experiment was the testing of SHM systems composed of sensor networks and machine learning algorithms, as well as the testing of prognostic methodologies for real-time SHM applications. However, the data from the sensor networks and the diagnostic algorithms have not been used in this work. The data on damage propagation caused by fatigue loading have been used here to evaluate the particle filtering capabilities in tracking the crack propagation and predicting the remaining life of the structure. The section compares the proposed balanced log-Normal process noise to the other existing formulations discussed in Section 3 using three dedicated performance metrics (Saxena et al., 2008).

4.1. Fatigue crack growth experiment

A stiffened aeronautical panel with dimensions 600 mm $\times$ 500 mm composed of 0.81 mm skin and four stringers was used as test structure. The panel was rigidly grounded on its lower end by a proper design of the lower edge, thereby simulating the skin-stringer-frame connection of real structures. The applied load was transferred from the actuator to the specimen through a dedicated steel structure composed of two C-shaped beams and a thin steel triangle, which were connected together using two series of bolts. Then, the steel triangle was connected to the actuator. By so doing, the structure has been clamped at its lower end and stretched along the stringer axis by the vertical load. A positive load ratio $R = S_{min}/S_{max} = 0.1$ was selected to avoid buckling instability. An artificial hole (with a diameter of 10 mm) and two deep narrow notches on the hole sides generated a high stress concentration factor that favored the crack initiation and propagation (the total length of the notch was around 16 mm). Figure 4 shows the panel and the test rig. The zoom on the central bay-notch shows the crack propagated from the tips of the notch. Table 2 summarizes the test features. The stress range acting on the skin was approximately 50 MPa, calculated using a finite element model (Sbarufatti, Manes, & Giglio, 2014). The FCG was observed by means of a simple caliper during the test, thus collecting the semi-crack length against load cycles, Figure 5. The semi-crack length measured by the caliper is provided to the algorithm at pre-determined load cycles, thus simulating a real-time application of the algorithm.

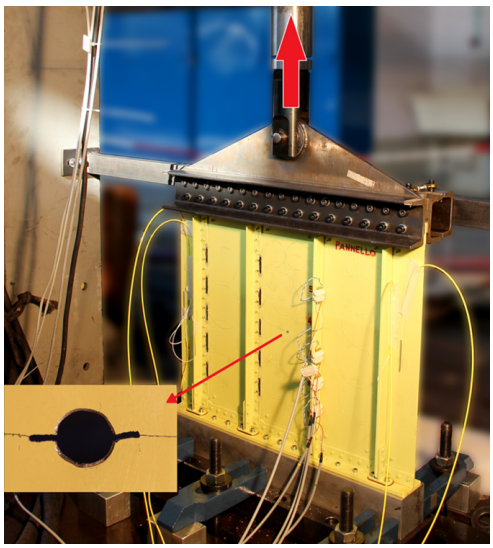

Figure 4. FCG experiment: the aeronautical panel is subject to tension-tension fatigue loads, and an artificial notch in the center of the bay induced the fatigue crack propagation. The thick, vertical arrow on the actuator rod represents the load direction.

4.2. Application of particle filtering algorithm to FCG observations

Here, the objective of the particle filtering-based prognostic algorithm is the estimation of the remaining number of load cycles to reach the end of the test, which is the instant when the semi-crack length becomes x_f = 60.5 mm, after N_f = 346000 load cycles. The RUL is the difference between the current load cycle, N, and the end-of-life of the damaged panel, N_f. Then, $\text{RUL}_k = N_f - N_k$, where N_k is the number

Table 2. Test features.

load shape	sinusoidal
load frequency	12 Hz
maximum force	$F_{max} = 35$ kN
load ratio	$R = 0.1$
damage type	skin crack
damage location	central bay
damage initiation	artificial notch, 16 mm
skin material (driving FCG)	Al2024-T6

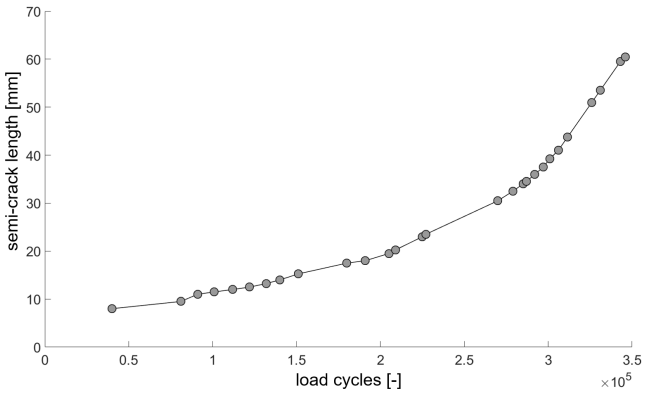

Figure 5. Semi-crack length as a function of the number of load cycles.

of applied load cycles at time step *k*.

The algorithm is a *sequential importance resampling*, described in (Arulampalam et al., 2002), with embedded *kernel smoothing* sub-algorithm for combined state-parameter estimation (Liu & West, 2001), named KS-PF, which states for *kernel smoothing-particle filtering*. The evolution equation $f(\cdot)$ grounds on the linear damage accumulation in Eq. (9) and the Paris' law (10) already used for the simulations in Section 3. The model parameter vector contains the two empirical parameters of the Paris' law: $\boldsymbol{\theta} = [\log(C), m]^T$, which are updated during run time, according to the kernel smoothing method. Since information on the uncertainty of the caliper was not available, the likelihood function has been modeled as an unbiased Gaussian pdf centered on the true semi-crack length and the variance of the measurement noise was empirically selected: $\sigma_\eta^2 = 2\,\text{mm}^2$. Equation (23) defines the main steps of the state estimation process from *k-1* to *k* using KS-PF.

$$
\begin{aligned}
\mu_{\boldsymbol{\theta},k}^{(i)} &= \sqrt{1-h^2}\,\boldsymbol{\theta}_{k-1}^{(i)} + (1-\sqrt{1-h^2})\,\mathrm{E}[\boldsymbol{\theta}]_{k-1} \\
x_k^{(i)} &= f(x_{k-1}^{(i)}, \mu_{\boldsymbol{\theta},k}^{(i)}, \boldsymbol{u}_{k-1}, \boldsymbol{\omega}_{k-1}^{(i)}) \\
\boldsymbol{\theta}_k^{(i)} &= \boldsymbol{\theta}_{k-1}^{(i)} + \mathcal{N}(0, h^2 \mathrm{V}[\boldsymbol{\theta}]_{k-1}) \\
\tilde{w}_k^{(i)} &= w_{k-1}^{(i)} p(z_k | x_k^{(i)}) \\
w_k^{(i)} &= \frac{\tilde{w}_k^{(i)}}{\sum_{j=1}^{N_s} \tilde{w}_k^{(j)}}
\end{aligned}
\tag{23}
$$

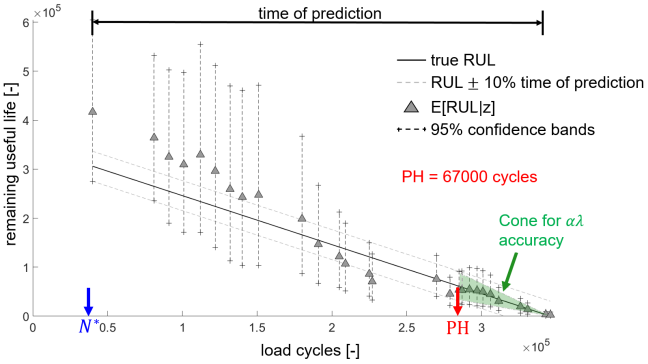

Figure 6. RUL prediction using the optimal log-Normal noise formulation.

Where $h \in \mathbb{R}_{[0,1]}$ is the smoothing parameter, which has been kept equal to 0.1 for all the simulations. The term $\mu_{\boldsymbol{\theta},k}^{(i)}$ is the kernel location of the *i-th* sample, which helps in concentrating the parameter samples in the high probability regions of the parameter-space. The input vector $\boldsymbol{u}_{k-1}$ contains the maximum and minimum stress within a single load cycle, $\boldsymbol{u}_{k-1} = [S_{max,k-1}, S_{min,k-1}]$, and forms the input of the damage growth rate model. Once a new observation becomes available, the algorithm propagates the samples up to the time step *k* referring to that observation, computes the likelihood of the observation given the samples $p(z_k|x_k^{(i)})$, and updates the weights to obtain an updated posterior distribution of the semi-crack length. Then, the samples are propagated in the future through (4) to provide the posterior estimation of the RUL. The resampling stage is performed using $p(x_k|z_{0:k})$: the posterior cumulative distribution function of the semi-crack length is approximated by the cumulative sum of the weights $w_k^{(i)}$, $\forall i = 1, \ldots, N_s$. Then, the *j-th* sample is extracted from the cumulative distribution function using the traditional Monte Carlo approach, i.e., $\Pr\{x_k^{(j)} = x_k^{(i)}\} = w_k^{(i)}$, $\forall j = 1, \ldots, N_s$: the sample *j* replaces the sample *i*. The procedure in (23), the RUL estimation (4) and the resampling stage are repeated until the true semi-crack length reaches x_f. Figure 6 shows the RUL predicted by the KS-PF algorithm based on the balanced log-Normal process noise discussed in this work and $\sigma_\omega^2 = 2$. The confidence bands always included the true RUL, and the expected value of the RUL prediction seemed to converge to the true RUL. Figure 6 shows also the time of prediction $N_f - N^*$, where N^* is the number of load cycles when the crack was detected and the algorithm started operating, and some indices that are discussed later to assess the prognostic performance: the prognostic horizon (PH) and the triangle used to calculated the $\alpha\lambda$ accuracy (AL).

4.3. Analysis of the particle filtering performance

The performance of the algorithm is examined using the process noises presented in Subsections 3.1, 3.2 and 3.3. The

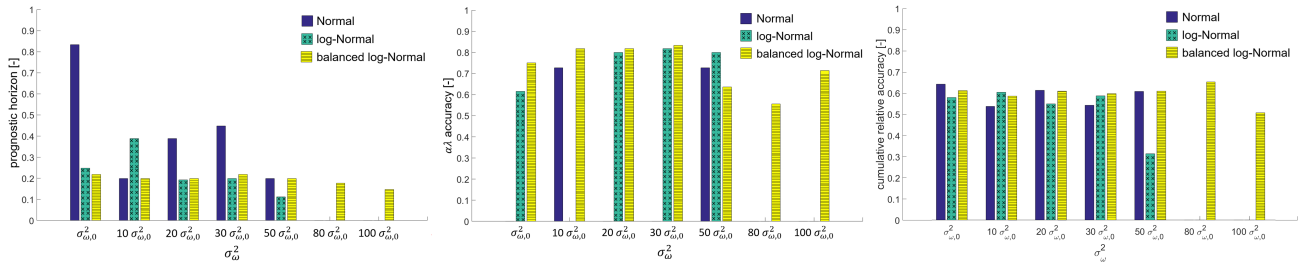

Figure 7. normalized PH (left), $\alpha\lambda$ accuracy (center) and CRA (right) using different process noises.

three algorithms have been run several times, and the process noise variance has been increased at every run in order to evaluate the prognostic capabilities with respect to σ^2_ω. The first process noise variances have been chosen empirically: $\sigma^2_{\omega,0} = 1e-4$ for the additive Gaussian noise and $\sigma^2_{\omega,0} = 0.1$ for the log-Normal process noises. Then, the variances have been incremented using the following sequence: 10, 20, 30, 50, 80 and 100 times $\sigma^2_{\omega,0}$.

Three performance metrics from (Saxena et al., 2008) have been calculated after each run: PH, AL and cumulative relative accuracy (CRA), which are described below.

- **Prognostic horizon** has been defined here as the difference between N_f and the time instant when the 60% of the RUL pdf area first falls between the range $[\text{RUL} \pm 10\%(N_f - N^*)]$. Also, it has been normalized over the time of prediction $(N_f - N^*)$. Then, different PHs coming from different damage propagations can be compared to one another.
- $\alpha\lambda$ **accuracy** is defined as the number of times that the 60% of the RUL pdf area falls within a region that shrinks as time passes by, once the PH criterion has been satisfied (Saxena et al., 2008). The region of interest has been defined here as a triangle starting when the PH criterion is satisfied, and the end of the region is represented by the vertex of the triangle in (N_f,0). AL has been normalized over all the RUL predictions made after the PH criterion is satisfied, then: $\text{AL} \in \mathbb{R}_{[0,1]}$. Figure 6 emphasizes the region to calculate the AL.
- **Cumulative relative accuracy** is an overall measure of the RUL prediction error over the entire run time. Here, CRA has been calculated as the sum of the relative accuracy (RA) values (23) weighted by linear, normalized weights (25).

$$\text{RA}_k = 1 - \frac{\text{RUL}_k - \text{E}[\text{RUL}]_k}{\text{RUL}_k} \tag{24}$$

$$\text{CRA} = \sum_{j=1}^{k'} \gamma_j RA_j \tag{25}$$

Where RUL_k is the true RUL of the panel at the *k-th* time step, while $\text{E}[\text{RUL}]_k$ is the expected RUL calculated with KS-PF at time step k. The term RA_k refers to the relative accuracy, which is the complement of the relative error (24) and k' is the total number of RUL predictions made during a single run of the algorithm. The weights γ_j are linearly increasing from 0 to 1, so RUL prediction errors close to the end-of-life of the panel are penalized with respect to RUL prediction errors made at the beginning of the operation, when the algorithm has collected a few data. The weights are then normalized such that $\sum_{j=1}^{k'} \gamma_j = 1$.

The prognostic results are collated to one another in Figure 7, where the prognostic metrics described above are analyzed against the increasing process noise variance. Analyzing the results of the PH (Fig. 7, left), the Gaussian process noise seems to outperform the other formulations. However, the results noticeably change using larger σ^2_ω. The PHs of the log-Normal and balanced log-Normal are comparable until $\sigma^2_\omega \leq 50 \cdot \sigma^2_{\omega,0}$. It is worth noting that the algorithm with additive Gaussian noise sometimes does not converge when $\sigma^2_\omega \geq 30 \cdot \sigma^2_{\omega,0}$ because some samples fell below zero. Also, the algorithms with Gaussian and log-Normal process noises never converge when the variance becomes too large ($\geq 80 \cdot \sigma^2_{\omega,0}$). The algorithm with additive Gaussian noise stopped because some particles never reached the critical semi-crack length x_f as they had fallen below zero. The algorithm with log-Normal noise failed because the bias introduced by the variance became too large and all the particles failed too early. Instead, the balanced log-Normal noise always converged and its normalized PH is fairly constant regardless the selected noise variance. This accentuates the reliability and robustness of the algorithm based on the balanced log-Normal noise. The analysis of the $\alpha\lambda$ accuracy (Fig. 7, center) stresses on the reliability of the proposed noise already emphasized by the PH. The additive Gaussian noise-based algorithm never met the $\alpha\lambda$ requirement but in two runs, and these runs are characterized by a small PH. The log-Normal noise-based algorithm met the $\alpha\lambda$ requirement in 5 cases over 7, and the accuracy is similar to the accuracy of the balanced process noise. However, the latter always met the requirement and provides an average $\alpha\lambda$ accuracy

of 0.7323. This means that, once the PH criterion is met, the RUL pdf calculated with the particle filtering based on the balanced log-Normal remains close to the true RUL. The three algorithms show comparable CRA until $\sigma_\omega^2 < 50 \cdot \sigma_{\omega,0}^2$. Then, the algorithms with additive Gaussian and log-Normal noises failed (Fig. 7, right).

The application of the three process noises emphasized the stability of the tdf built with $\omega \sim \mathcal{N}(-\frac{\sigma_\omega^2}{2}, \sigma_\omega^2)$: the algorithm always converges to the RUL and the performance is apparently independent of σ_ω^2. From these results, it is reasonable to assume that the tuning of particle filtering based on the balanced log-Normal process noise requires a limited effort if compared to the other existing formulations presented in literature. A wide range of σ_ω^2 would produce the same prognostic results, while the tuning of the variance of the Normal and biased log-Normal process noises appears less robust. In addition, the authors believe that the introduction of a bias caused by a random perturbation in the evolution equation is not appropriate for real-time applications, where the damage progression trend cannot be predicted in advance.

5. CONCLUSIONS

The analysis conducted in this work emphasized that the selection of the process noise is a primary issue for prognostics of monotonic degradation phenomena using particle filtering. The selection and design of a proper process noise is discussed analyzing three requirements: (i) the particles must remain in the state-space support, (ii) each particle should behave like the physical phenomenon, which is a monotonic damage growth, and (iii) the process noise must not introduce any bias in the evolution equation.

The review of existing formulations, extensively applied in literature, has shown that the additive Gaussian noise does not meet conditions (i) and (ii), while the log-Normal noise $e^\omega,\ \omega \sim \mathcal{N}(0, \sigma_\omega^2)$ produces a biased evolution of the samples, and the bias depends on σ_ω^2. So, it does not meet condition (iii). The process noise proposed in this paper grounds on a log-Normal distribution with specified mean and variance ($e^\omega,\ \omega \sim \mathcal{N}(\mu_\omega, \sigma_\omega^2),\ \mu_\omega = -\sigma_\omega^2/2$) and it is able to satisfy all the conditions expressed above: the particles remain in the state-space support, each of them represents a potential damage progression path and the amount of process noise (i.e., the selection of σ_ω^2) does not alter the evolution equation.

A particle filtering algorithm with embedded kernel smoothing sub-algorithm for combined state-parameter estimation has been applied to FCG data from a relevant aeronautical structure, and the use of the three different process noises discussed in this paper has been critically analyzed. A preliminary sensitivity analysis of the filter against the process noise variance σ_ω^2 has shown that, at this stage of the research, the proposed formulation outperforms other process noises already existing in literature. The prognostic algorithm based on the balanced log-Normal noise always converges regardless of the amount of perturbation introduced by ω.

The work should be further extended with: (i) a sensitivity analysis of particle filtering to the process noise variance using several runs, (ii) additional case studies concerning fatigue damage progression (e.g., damages growing in composite laminates or creep degradation), and (iii) the extension to other monotonic degradation processes.

REFERENCES

Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *Signal Processing, IEEE Transactions on*, *50*(2), 174–188.

Baraldi, P., Compare, M., Sauco, S., & Zio, E. (2013). Ensemble neural network-based particle filtering for prognostics. *Mechanical Systems and Signal Processing*, *41*(1), 288–300.

Baraldi, P., Mangili, F., & Zio, E. (2012). A kalman filter-based ensemble approach with application to turbine creep prognostics. *Reliability, IEEE Transactions on*, *61*(4), 966–977.

Cadini, F., Zio, E., & Avram, D. (2009a). Model-based monte carlo state estimation for condition-based component replacement. *Reliability Engineering & System Safety*, *94*(3), 752–758.

Cadini, F., Zio, E., & Avram, D. (2009b). Monte carlo-based filtering for fatigue crack growth estimation. *Probabilistic Engineering Mechanics*, *24*(3), 367–373.

Chiachıo, J., Chiachıo, M., Saxena, A., Rus, G., & Goebel, K. (2013). An energy-based prognostics framework to predict fatigue damage evolution in composites. In *Proceedings of the annual conference of the prognostics and health management society* (Vol. 1, pp. 363–371).

Chiachıo, M., Chiachıo, J., Saxena, A., Rus, G., & Goebel, K. (2014). An efficient simulation framework for prognostics of asymptotic processes-a case study in composite materials. In *Proceedings of the european conference of the prognostics and health management society, nantes, france* (pp. 202–214).

Corbetta, M., Saxena, A., Giglio, M., & Goebel, K. (2015). Evaluation of multiple damage-mode models for prognostics of carbon fiber-reinforced polymers. In *International workshop on structural health monitoring.*

Corbetta, M., Sbarufatti, C., Manes, A., & Giglio, M. (2014). On dynamic state-space models for fatigue-induced structural degradation. *International Journal of Fatigue*, *61*, 202–219. doi: http://dx.doi.org/10.1016/j.ijfatigue.2013.11.008

Doucet, A., Godsill, S., & Andrieu, C. (2000). On sequential monte carlo sampling methods for bayesian filtering.

Statistics and computing, *10*(3), 197–208.

Gordon, N. J., Salmond, D. J., & Smith, A. F. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. , *140*(2), 107–113.

Haug, A. (2005). A tutorial on bayesian estimation and tracking techniques applicable to nonlinear and non-gaussian processes. *MITRE Corporation, McLean*.

Jouin, M., Gouriveau, R., Hissel, D., Péra, M.-C., & Zerhouni, N. (2015). Particle filter-based prognostics: Review, discussion and perspectives. *Mechanical Systems and Signal Processing*.

Liu, J., & West, M. (2001). Combined parameter and state estimation in simulation-based filtering. In *Sequential monte carlo methods in practice* (pp. 197–223). Springer.

Orchard, M., Kacprzynski, G., Goebel, K., Saha, B., & Vachtsevanos, G. (2008). Advances in uncertainty representation and management for particle filtering applied to prognostics. In *Prognostics and health management, 2008. phm 2008. international conference on* (pp. 1–6).

Orchard, M. E., & Vachtsevanos, G. J. (2007). A particle filtering-based framework for real-time fault diagnosis and failure prognosis in a turbine engine. In *Control & automation, 2007. med'07. mediterranean conference on* (pp. 1–6).

Orchard, M. E., & Vachtsevanos, G. J. (2009). A particle-filtering approach for on-line fault diagnosis and failure prognosis. *Transactions of the Institute of Measurement and Control*.

Sankararaman, S. (2015). Significance, interpretation, and quantification of uncertainty in prognostics and remaining useful life prediction. *Mechanical Systems and Signal Processing*, *52*, 228–247.

Sankararaman, S., & Goebel, K. (2015). Uncertainty in prognostics and systems health management. *International Journal of Prognostics and Health Management*, *6*.

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for evaluating performance of prognostic techniques. In *Prognostics and health management, 2008. phm 2008. international conference on* (pp. 1–17).

Sbarufatti, C., Manes, A., & Giglio, M. (2014). Application of sensor technologies for local and distributed structural health monitoring. *Structural Control and Health Monitoring*, *21*(7), 1057–1083.

Tang, L., DeCastro, J., Kacprzynski, G., Goebel, K., & Vachtsevanos, G. (2010). Filtering and prediction techniques for model-based prognosis and uncertainty management. In *Prognostics and health management conference, 2010. phm'10.* (pp. 1–10).

Yang, W., Yuan, S., Qiu, L., Zhang, H., & Ling, B. (2012). A particle filter and lamb wave based on-line prognosis method of crack propagation in aluminum plates. In *4th international symposium on ndt in aerospace.*

Zio, E., & Di Maio, F. (2012). Fatigue crack growth estimation by relevance vector machine. *Expert Systems with Applications*, *39*(12), 10681–10692.

Zio, E., & Peloni, G. (2011). Particle filtering prognostic estimation of the remaining useful life of nonlinear components. *Reliability Engineering & System Safety*, *96*(3), 403–409.

BIOGRAPHIES

Matteo Corbetta received B.Sc., M.Sc. and Ph.D. degrees in mechanical engineering at Politecnico di Milano, Italy. He is currently a post-doc research scientist with the department of mechanical engineering at Politecnico di Milano. His research interests include probabilistic and stochastic methods for diagnostics and prognostics of large structures, machine learning and statistical signal processing, system failure modeling, Bayesian filtering and Monte Carlo approaches, distributed sensing and structural health monitoring. He is currently involved in international research projects and collaborations on real-time algorithms for structural health monitoring and fatigue damage prognosis of composite materials.

Claudio Sbarufatti is a research scientist and assistant professor with the department of mechanical engineering at Politecnico di Milano, Italy. His main research topics are sensor network design for damage and impact diagnosis, model-based structural health monitoring, prognosis of residual life under constant and variable loads, impact modeling on composite structures. He is currently involved in the management of international projects mainly focused on structural health monitoring and residual life prognosis.

Marco Giglio is Full Professor of Mechanical Design and Strength of Materials, and works in the Department of Mechanical Engineering at Politecnico di Milano, Italy. His research fields are novel methods for SHM application, ballistic damage and evaluation of the residual strength, methods of fatigue strength assessment in mechanical components subjects to multiaxial state of stress, design and analysis of helicopter components with defects, optimization of structures for energy application. He is the author of over 150 scientific papers in international journals and conferences and is a member of scientific associations (AIAS, Italian Association for the Stress Analysis, IGF, Italian Group Fracture).