Real-Time Prognosis of Crack Growth Evolution Using Sequential Monte Carlo Methods and Statistical Model Parameters

Matteo Corbetta, Claudio Sbarufatti, Andrea Manes, and Marco Giglio

ACRONYMS AND ABBREVIATIONS:

ANN	Artificial Neural Network
DSS	Dynamic State-Space
EOP	End of Prediction
FCG	Fatigue Crack Growth
FEM	Finite Element Model
MCMC	Markov chain Monte Carlo
MH	Metropolis-Hastings
MLP	Multi-Layer Perceptron
PDF	Probability Density Function
PHM	Prognostic Health Management
RL	Residual Lifetime
SDSS	Stochastic Dynamic State-Space
SHM	Structural Health Monitoring
SIF	Stress Intensity Factor
SIR	Sequential Importance Resampling
SIS	Sequential Importance Sampling
SMC	Sequential Monte Carlo
TTF	Time To Failure

NOTATION:

$\mathrm{p}(\cdot)$	probability density function
$\mathrm{p}(\cdot\vert\cdot)$	conditioned probability density function
$\mathrm{q}(\cdot\vert\cdot)$	conditioned proposal density function
$\mathrm{N}(\cdot,\cdot)$	*s*-normal probability density function
$\mathrm{L}(\cdot\vert\cdot)$	likelihood
$\alpha(\cdot,\cdot)$	acceptance probability
$\Pr\{\cdot\}$	probability
$\Pr\{\cdot\vert\cdot\}$	conditioned probability
$\mathrm{E}(\cdot)$	expected value
$\mu_\cdot$	*s*-mean
$\sigma^2_\cdot$	variance
$\delta(\cdot)$	Dirac delta function
$\mathrm{x_k}$	system state at the k-th time step
$\mathrm{x^i}$	i-th sample of the system state (“particle”)
$\mathrm{z_k}$	measure (or observation) at the k-th time step
$\mathrm{z_{0:k}}$	measure sequence up to the k-th time step
$\mathrm{w_k^i}$	weight of the i-th sample at the k-th time step
$\mathrm{v_k}$	artificial noise affecting the evolution equation
σ^2_{v}	variance of the random process v
$\mathrm{n_k}$	noise affecting the measurement system
σ^2_{n}	variance of the random noise n
ϑ	vector of model parameters
$\mathrm{f}(\cdot)$	evolution function of the system state
$\mathrm{g}(\cdot)$	observation function
U	update frequency parameter
H	memory parameter
Θ_{H}	Matrix of the last H samples of the Markov chain
$\mathrm{c_d}$	scaling parameter of the adaptive proposal algorithm
B	average bias index
RL	actual residual lifetime of the system
$\widehat{\mathrm{RL}}$	estimated residual lifetime of the system

Manuscript received February 20, 2014; revised May 19, 2014, July 29, 2014, September 09, 2014, and October 03, 2014; accepted October 28, 2014. Date of publication November 11, 2014; date of current version June 01, 2015. Associate Editor: E. Zio.

The authors are with Politecnico di Milano, Dipartimento di Meccanica, Milan 20156, Italy (e-mail: matteo.corbetta@polimi.it; claudio.sbarufatti@polimi.it; andrea.manes@polimi.it; marco.giglio@polimi.it).

I. Introduction

Nowadays, the necessity of advanced maintenance strategies for critical systems is a concept generally accepted by the scientific and industrial community. The safety requirements are becoming higher as are the maintenance costs of large structures (bridges, high-rise buildings), and flight structures (airframes and aeronautical components). Obviously, maintenance optimization requires an in-depth knowledge of the system health, possibly with information related to the Residual Lifetime (RL) of the system. This concept goes well with the current developments on Structural Health Monitoring (SHM) and Prognostic Health Management (PHM) systems. SHM focuses on identification, localization, and quantification of damages [1], [2], while PHM focuses on the estimation of the Time To Failure (TTF), based on the current health condition, reliability prediction [3], and probabilistic state evolution [4]. Focusing on PHM, Fatigue Crack Growth (FCG) is one of the most widespread degradation problems affecting metallic structures, and much published work is dedicated to FCG prediction ([5], [6], and [7] are just a few examples). In particular, structural engineers are familiar with the phenomenological aspects of FCG. Recent advances on Sequential Monte Carlo (SMC) methods, specifically particle filtering algorithms, allowed coupling the mathematical FCG model (Paris-Erdogan model [8], NASGRO model [9], etc.) with the multiple uncertainties coming from measurement systems, and the intrinsic uncertainty of the phenomenon ([10], [11], and [12]).

Particle filter can be applied to nonlinear system evolutions described by non-Gaussian probability density functions (PDFs) [13]. Moreover, it is particularly suitable for real-time applications because of its sequential updating of the system state estimation. From a practical point of view, the main drawback of the method is the uncertainty related to the model parameters.[1] As recently presented by Kantas *et al.* [14], the knowledge of the actual model parameters can potentially be improved during the particle filtering operation (sometimes producing biased estimation depending on the adopted technique). The identification of model parameters is a crucial step for the lifetime prediction of structures subjected to FCG. As a matter of fact, the variability of the phenomenon markedly affects the TTF estimation. The deterministic definition of model parameters within particle filters can provide good results in terms of state estimation (if the artificial noise is able to filter the measure uncertainty), but it can make the life prediction useless because the estimation of the future evolution will be wrong if the expected parameters are different from the actual ones [4], [15], [16]. For this reason, a Sequential Monte Carlo sampling framework considering the statistical definition of FCG parameters is presented in this paper. The Dynamic State-Space (DSS) model, which is at the basis of the algorithm, becomes a Stochastic Dynamic State-Space (SDSS) model, and it is recursively updated using a Markov chain Monte Carlo (MCMC) subroutine. The methodology is first tested on simulated crack propagations, and then on real-portions of a helicopter fuselage, introducing an automatic estimation of the Stress Intensity Factor (SIF) acting on the crack tips by means of dedicated Artificial Neural Networks (ANN) trained with Finite Element (FE) data. The proposed algorithm is compared with a standard particle filter formulation for FCG, highlighting the differences in terms of residual life prediction. The application of a modified particle filter for real-time prognosis on several portions of helicopter fuselage is the main novelty of the work. As a matter of fact, the majority of particle filtering literature for lifetime prediction makes use of FCG simulations or small specimens in laboratory environments to prove the prognostic capabilities. If more complex structures are of interest, the amount of uncertainties related to FCG prediction explodes, and the damage evolution can be extremely different from the one that is actually predicted based on material properties available in literature. In this context, the proposed algorithm is applied several times, proving the robustness of the method and its validity for the prognosis of realistic structures. A performance index is used to evaluate the stability of the particle filter with adaptation of model parameters, according to the literature of prognostic performance. This performance index is applied on both simulated and experimental data.

The paper is structured as follows. Section II provides an overview of SMC sampling (i.e., particle filter), and MCMC algorithms. In Section III, the stochastic definition of the DSS model and the updating of the model parameters are shown. In Section IV, the results of the simulated crack growth case are presented. Section V is dedicated to the validation of the method with experimental FCG test data, showing the structure under discussion, the test-rig, the ANN for SIF range estimation, and the global prognostic performance. In the last section, the results are discussed.

II. Theoretical Background

The literature on Monte Carlo Sampling, Bayesian filters, and related techniques is very extensive; therefore, this section only provides a brief summary of the mathematical tools implemented in the prognostic unit. These algorithms are particle filtering methods (named also Sequential Monte Carlo Sampling, Sequential Importance Sampling (SIS), or Sequential Importance Resampling (SIR) algorithms), and Metropolis-Hastings (MH) algorithms with adaptive proposal distribution. The interested reader can refer to [13], [17], [18], and references therein for in-depth information on each subject.

A. Summary of Particle Filter

Particle filter is a recursive Bayesian filter method based on Monte Carlo simulation. It can be considered as a generalization of the Kalman filter using Monte Carlo sampling to estimate the system state in the presence of highly nonlinear dynamics and non-Gaussian PDFs. The two equations needed to implement a particle filter are the evolution (1), and the observation equation linking the measures with the system state (2):

$$\mathrm{x_k} = \mathrm{f}(\mathrm{x_{k-1}}, \vartheta, \mathrm{v_{k-1}}), \quad (1)$$

$$\mathrm{z_k} = \mathrm{g}\left(\mathrm{x_k}, \mathrm{n_k}\right). \quad (2)$$

A series of possible trajectories or particles defined by $\mathrm{x^i}$ represent the stochastic system evolution, indicating with the su-

[1] i.e., overlooking the mathematical problems related to the sampling impoverishment etc. [13].

perscript i the general i-th particle. The probabilities associated to the trajectories depend on the model describing the phenomenon, and the available observations z on the system itself. Supposing to be at time step $\mathrm{k} > 1$, the particle $\mathrm{x}_\mathrm{k}^\mathrm{i}$ (properly weighted using the likelihood of the measures z_k and the probabilities related to the model of the system) is representative of the conditioned probability of the system state given the series of available measures $\Pr\left\{\mathrm{x}_\mathrm{k}^\mathrm{i}|\mathrm{z}_{0:\mathrm{k}}\right\}$. The weight of each trajectory $\mathrm{w}_\mathrm{k}^\mathrm{i}$ is defined in (3), supposing the simplest weight definition often referred to as bootstrap approximation.

$$\mathrm{w}_\mathrm{k}^\mathrm{i} = \mathrm{w}_{\mathrm{k}-1}^\mathrm{i} \Pr\left\{\mathrm{z}_\mathrm{k}|\mathrm{x}_\mathrm{k}^\mathrm{i}\right\}. \tag{3}$$

This weight definition implies an importance density function equal to the transition density describing the system evolution. The term $\Pr\left\{\mathrm{z}_\mathrm{k}|\mathrm{x}_\mathrm{k}^\mathrm{i}\right\}$ represents the likelihood of the measure z given the value of the particle x^i at the general k-th step. The approximation of the posterior distribution is now available, based on the couples $\left\{\mathrm{x}_\mathrm{k}^\mathrm{i}, \mathrm{w}_\mathrm{k}^\mathrm{i}\right\}_{\mathrm{i}=1}^{\mathrm{N}_\mathrm{s}}$, indicating with N_s the number of particles (4).

$$\mathrm{p}\left(\mathrm{x}_\mathrm{k}|\mathrm{z}_{0:\mathrm{k}}\right) \approx \sum_{\mathrm{i}=1}^{\mathrm{N}_\mathrm{s}} \widetilde{\mathrm{w}}_\mathrm{k}^\mathrm{i} \delta\left(\mathrm{x}_\mathrm{k} - \mathrm{x}_\mathrm{k}^\mathrm{i}\right). \tag{4}$$

The term $\widetilde{\mathrm{w}}_\mathrm{k}^\mathrm{i}$ indicates a normalized weight such that $\sum_\mathrm{i} \widetilde{\mathrm{w}}_\mathrm{k}^\mathrm{i} = 1$. The transition density (5) that is directly related to (1) drives the particle evolution in time from k to k + 1.

$$\mathrm{x}_{\mathrm{k}+1}^\mathrm{i} \sim \mathrm{p}\left(\mathrm{x}_{\mathrm{k}+1}^\mathrm{i}|\mathrm{x}_\mathrm{k}^\mathrm{i}\right) = \mathrm{p}\left(\mathrm{f}\left(\mathrm{x}_\mathrm{k}^\mathrm{i}, \vartheta\right), \sigma_{\mathrm{v}_\mathrm{k}}^2\right). \tag{5}$$

Once the samples $\mathrm{x}_\mathrm{k}^\mathrm{i}$ and the measure z_k are available, a subsequent step of the algorithm can be computed by means of (3), (4), and (5). The estimation of the RL starts from the posterior distribution of the state $\mathrm{p}\left(\mathrm{x}_\mathrm{k}|\mathrm{z}_{0:\mathrm{k}}\right)$. Starting from (4), the evolution of the system up to a pre-defined or critical state can be computed considering all the possible evolutions, as highlighted in [10].

B. Metropolis-Hastings Algorithm

MCMC methods are widespread algorithms able to approximate complex, non-Gaussian, and multivariate posterior distributions [19]. Among various MCMC methods, the MH algorithm is one of the most popular algorithms because of its simplicity. The application of the MH algorithm in the field of parameter estimation delivers, as a result, a series of samples generated by the posterior PDF of the parameters, given a series of observations of quantities directly related to the parameters themselves. Moreover, the algorithm is able to estimate the noise associated to the observations. The algorithm finds general acceptance in this field, proven by the extensive literature on the topic. The MH algorithm employed hereafter has recently been further applied to crack growth parameter identification in the presence of variable-amplitude loading conditions [20]. Roughly speaking, it is a random walk based on a transition kernel linking two subsequent samples. This transition kernel is a PDF properly created to satisfy the reversibility and the ergodicity of the chain to guarantee an unbiased estimation of the posterior PDF of the parameters defined $\mathrm{p}\left(\vartheta|\mathrm{z}_{0:\mathrm{k}}\right)$. This unbiased estimation of the posterior PDF is allowed, satisfying the detailed balance condition [20]. A correct transition kernel can be built using a proposal distribution called $\mathrm{q}(\vartheta_\mathrm{j}|\vartheta_{\mathrm{j}-1})$, and an acceptance probability $\alpha\left(\vartheta_\mathrm{j}, \vartheta_{\mathrm{j}-1}\right)$. The latter depends on the likelihood of the sample given the measures, on the proposal distribution, and on the prior distribution (if it exists). The term α governs the acceptance or the rejection of the sample ϑ_j generated by the random walk. The definition of the likelihood strongly depends on the random processes affecting the observations. A Gaussian likelihood is assumed here, considering a measurement system affected by Gaussian random noise. The likelihood of the j-th sample with respect to the observation vector can be computed according to Table I. Because the acceptance of the sample depends on the ratio of two likelihoods, the proportionality can be changed into an equality for the algorithm implementation. The term σ_n^2 in the likelihood function is the variance of the random noise associated to the observations. This value is calculated through the MH algorithm itself. A sample of the random noise variance is extracted from a proposal distribution with the same procedure adopted for the parameter vector ϑ. As a consequence, the acceptance of the parameter samples depends on the sample of σ_n^2, too. At the end of the algorithm, a series of samples coming from the distribution of the noise variance are available, and they are representative of the noise affecting the measurement system and the damage evolution process.

TABLE I
MH ALGORITHM WITH ADAPTIVE PROPOSAL

```
Initialize the algorithm:
    $\vartheta_0$, $q(\vartheta_j|\vartheta_{j-1})=q(\mu_\vartheta=\vartheta_0,\sigma^2=\sigma_0^2)$, $\Pr\{\vartheta_0\}$
    $U=U_0$, $H=H_0$
for j = 1 : N
    Draw a candidate $\vartheta_j \sim q(\vartheta_j|\vartheta_{j-1})=q(\mu_\vartheta=\vartheta_{j-1},\sigma^2)$
    Evaluate the likelihood $L(\vartheta_j|z_{0:k}) = \frac{1}{\sigma_n}\exp\left\{-\frac{\left(z_{0:k}-x(\vartheta_j)\right)^2}{2\sigma_n^2}\right\}$
    Accept and store the sample $\vartheta_j$ with probability
        $\alpha(\vartheta_j,\vartheta_{j-1}) = \min\left(1, \frac{L(\vartheta_j|z_k)}{L(\vartheta_{j-1}|z_k)} \frac{\Pr\{\vartheta_j\}}{\Pr\{\vartheta_{j-1}\}} \frac{q(\vartheta_{j-1}|\vartheta_j)}{q(\vartheta_j|\vartheta_{j-1})}\right)$
    If the remainder of (j/U) is null
        Generate the residual of the chain: $R(\Theta_H)=\Theta_H-E(\Theta_H)$
        Update the covariance matrix of the proposal:
            $\sigma^2=\sigma_{\Theta_H}^2=\frac{c_d^2}{H-1}R(\Theta_H)^T R(\Theta_H)$
    end
end
Erasing the burn-in period and saving one sample every $n_t$ samples
(thinning).
```

One of the main challenges of the algorithm is the tuning of the variance associated to the proposal distribution q. Intuitively, the proposal distribution should be able to draw reasonable samples according to the initially known observations. If the proposal variance is too large, the number of accepted samples will be too low, and the convergence of the chain cannot be guaranteed. On the other hand, if this variance is too small, the random walk will produce jumps that are too small, and the chain will require too many samples to reach the convergence. The use of an adaptive proposal distribution resolves the problem, and a number of papers discuss the problem of adaptive proposal distributions to maximize the acceptance ratio and the convergence of the MCMC algorithms [21]–[26].

The method presented in [25] is used hereafter, due to its simplicity and effectiveness proven several times [4], [20]. It is able to recursively update the variance of the proposal distribution using the residuals of the chain, introducing two additional constant parameters U, and H called the *update frequency*, and the *memory* parameters, respectively. The first parameter drives the interval between two updates of the proposal variance, and the second parameter drives the number of samples used to evaluate the residuals and the covariance matrix of the proposal. In spite of the slight bias introduced with this method, the authors in [25] state that the error is negligible in the majority of the cases, and the chain remains ergodic [26]. Moreover, the thinning procedure at the end of the algorithm reduces the correlation of the samples [27].

A pseudo-code of the Metropolis-Hastings algorithm with an adaptive proposal is presented hereafter in Table I. The scaling parameter is defined as $\mathrm{c_d} = 2.4/\sqrt{\mathrm{d}}$, where d is the number of parameters to be estimated [25], [28], and $\Pr\{\vartheta_\mathrm{j}\}$ is the prior probability associated to the sample ϑ_j. It should be noted that the matrix Θ contains also the samples of the measurement noise in addition to the model parameters. The MH algorithm is used together with the particle filter method in Section III to build the prognostic system.

III. Stochastic Definition, and Updating of a Dynamic State-Space Model

Consider a system state $\mathrm{x_k}$; it evolves in time, observing the mathematical law f. This law depends on the parameters $\vartheta = [\vartheta_1, \vartheta_2, \ldots, \vartheta_\mathrm{m}]$. Usually, the model f links the state $\mathrm{x_{k-1}}$ with the subsequent $\mathrm{x_k}$ using deterministic model parameters and an artificial random noise needed to produce a stochastic process, thus generating the Dynamic State-Space model. However, the Stochastic DSS (SDSS) model proposed in [15] is used hereafter to design the prognostic system.

A. Statistical Definition of Model Parameters

The definition of empirical constants is commonly affected by errors from the measurement systems, shortage of available data, uncertainties related to the regression procedure, or the intrinsic uncertainty of the described phenomenon. Therefore, many statistical descriptions of model parameters are available in the literature for a wide range of engineering problems.

It is possible to use the parameter PDFs to produce a swarm of possible state evolutions, in which every state evolves according to a particular sample of the parameters. In this context, each parameter $\vartheta_\mathrm{j} \subset \vartheta, \quad \forall j = 1 \div \mathrm{m}$ is described by a PDF $\mathrm{p}\left(\mu_{\vartheta_\mathrm{j}}, \sigma^2_{\vartheta_\mathrm{j}}\right)$, and a covariance term $\mathrm{Cov}\left(\vartheta_\mathrm{i}, \vartheta_\mathrm{j}\right)$ defines the correlation between the i-th and j-th parameters. The Monte Carlo Sampling allows the extraction of random values of ϑ_j from $\mathrm{p}\left(\mu_{\vartheta_\mathrm{j}}, \sigma^2_{\vartheta_\mathrm{j}}\right)$, and assigns these samples to the particles $\mathrm{x^i}$ to generate the SDSS model. The initial weights are set equal to $\Pr\{\vartheta^\mathrm{i}\}$ supposing that a prior distribution of the parameters is available. This choice is compatible with the pdf of $(\log\mathrm{C}, \mathrm{m})$ of the crack growth model used later. In practice, the crack propagation problem is well-studied, and a distribution for the two parameters $(\log\mathrm{C}, \mathrm{m})$ is available in literature for some materials (e.g., the Aluminum alloy Al2024 used hereafter [29]). Thus, it is reasonable to set the initial weight proportional to the probability of the parameter samples assigned to each particle. This initial assignment is done at each resampling stage, according with the parameter sample ϑ^i.

The MH algorithm presented in Section II.B updates the PDF of the parameters during the system operation, producing the posterior distributions of the parameters, given the measures. The updating is carried out according to the resampling strategy adopted for the particle filter algorithm (i.e., every time a resampling technique regenerates the particles). If a correlation among the parameters $\vartheta_\mathrm{j} \subset \vartheta$ is present, the procedure remains the same, provided that a multidimensional proposal PDF with correlation parameters $\rho_{\vartheta_\mathrm{i},\vartheta_\mathrm{j}}$ is employed in the algorithm.

B. Updating via Metropolis-Hastings Algorithm

The updating of model parameters is based on the measures available from the measurement system. In this way, the knowledge of the model parameters increases in time, and improves the DSS model of the phenomenon from the prediction capability viewpoint. First, the parameter distributions become thinner, focusing on the correct parameter for that particular evolution of the system (reducing the inter-specimen variability as defined in [30]). Second, the estimation of the noise of the measures allows tuning the random noise inserted on the DSS model, optimizing the replicated intra-specimen variability.

The MH algorithm works together with the particle filter method in the same prognostic unit, and is activated when the resampling step of the particle filter algorithm is required. As a matter of fact, two resampling strategies can be adopted: the Sequential Importance Sampling (SIS) algorithm, and the Sequential Importance Resampling (SIR) algorithm. The SIS algorithm requires resampling only when the number of effective particles becomes unable to reproduce an effective PDF of the system state. An estimation of the number of effective particles $\mathrm{N_{eff}}$ is shown in (6) according to [13], indicating with $\widetilde{\mathrm{w}}^\mathrm{i}_\mathrm{k}$ the normalized weight of the i-th particle at the general k-th time step. If $\mathrm{N_{eff}}$ drops below a predetermined threshold, the resampling procedure is performed, and the MH produces new parameter PDFs and a new variance of the noise for the SDSS model. The SIR algorithm works as the SIS; nonetheless, resampling is performed every time a new value $\mathrm{z_k}$ is received from the measurement system. In this case, the MH algorithm estimates new PDFs for the model parameters $\mathrm{p_{\Theta|Z}}$ and a new variance of the noise σ^2_v at each discrete time-step of the system operation. Table II shows the pseudo-code of the particle filter combined with the MH algorithm. This method allows obtaining an unbiased estimation of $\mathrm{p}\left(\vartheta|\mathrm{z_k}\right)$ to the detriment of high computational costs.

$$\mathrm{N_{eff}} \approx \frac{1}{\sum\limits_\mathrm{i} \left(\widetilde{\mathrm{w}}^\mathrm{i}_\mathrm{k}\right)^2}. \tag{6}$$

IV. Performance Analysis on Simulated Case

The prognostic unit shown in Section III is applied to a simulated fatigue crack propagation. The simple mechanical structure is a simulated thin aluminum plate large enough to avoid any influence of the boundaries over the stress state at the crack tip. The phenomenon of crack propagation is described by the NASGRO (7) from [9] that provides the FCG rate, together with the assumption of linear damage accumulation (8). NASGRO

TABLE II
PARTICLE FILTER WITH STOCHASTIC DSS MODEL

```
Initialize the algorithm
            $x_0^i \sim p(\mu_{x_0}, \sigma_{x_0}^2)$,      $\forall\, i=1 \div N_s$
  $\vartheta_j^i \sim p\left(\mu_{\vartheta_j}, \sigma_\vartheta^2\right)$      $\forall\, j=1 \div n$, $\forall\, i=1 \div N_s$
  $x_0^i \leftarrow \vartheta^i$, $w_0^i = \Pr\{\vartheta^i\}\ \forall i=1{:}N_s$
for k = 1 to N
  Sequential estimation of PDF of the state (basic particle filter)
            $p(x_k|z_{0:k}) = \sum_{i=1}^{N_s} \widetilde{w}_k^i \delta(x_k^i - x_k)$
  If resampling is required at general k-th step:
    $x_k^i \sim p(x_k|z_{0:k})$ (according to systematic resampling method)
  $[\vartheta] = MH(z_{0:k})$ new samples $\vartheta$ from the posterior PMF $p(\vartheta|z_{0:k})$
            $x_0^i \leftarrow \vartheta^i$, $w_0^i = \Pr\{\vartheta^i\}\ \forall i=1{:}N_s$
  end
end
```

law describes the whole FCG rate from the threshold regime up to the unstable crack propagation region, while the linear damage accumulation function provides the crack length at different discrete time steps. This model constitutes the core of the Dynamic State-Space model of the particle filter.

$$\frac{dx}{dN} = C\left(\frac{1-f}{1-R}\Delta K\right)^m \frac{\left(1 - \frac{\Delta K}{\Delta K_{TH}}\right)^l}{\left(1 - \frac{K_{MAX}}{K_C}\right)^s} \tag{7}$$

$$x_k = x_{k-1} + \Delta N \frac{dx}{dN} v_k. \tag{8}$$

Some variables in (7) are easily recognizable as the Stress Intensity Factor ΔK affecting the crack tip, the two empirical parameters C and m appearing also in the well-known Paris-Erdogan model [8], and the load ratio R defined as the minimum over maximum applied load. An explanation of all the entities within the NASGRO model is beyond the aim of the paper, and the interested readers can refer to [9] for further information about the NASGRO equation. Equation (8) defines the crack length at the general k-th time step assuming that the load cycle increment ΔN is relatively small ($\Delta N \rightarrow 1$). The noise v_k associated to the DSS model is defined in Section IV.D according to different kinds of simulations. As explained above, the DSS model in (7) becomes a Stochastic DSS model considering a statistical definition of the model parameters. ϑ is the vector of the model parameters containing the empirical constants C, m (governing the crack growth). A series of samples is extracted from the probability density functions of the parameters in ϑ (9), and they are used to propagate the particles describing the process (10).

$$\vartheta^i = [C, m]^i \sim p\left(\mu_\vartheta, \Sigma_\vartheta\right) \tag{9}$$

$$x_k^i = x_k^i\left(\vartheta^i\right) \sim p\left(f\left(x_{k-1}^i, \vartheta^i\right), v_k\right) \tag{10}$$

$\vartheta^i = [C, m]^i$ represents a random sample from $p\left(\mu_\vartheta, \Sigma_\vartheta\right)$, and it is randomly associated to a particle representing the crack length x_k^i. The parameter sample must remain the same for the whole particle life. Once resampling is required, the MH algorithm estimates the new posterior distributions, and the procedure can be repeated, associating a new parameter sample to the new particle. The random noise associated to the process is defined according to a lognormal distribution (because the crack can only increase over time).

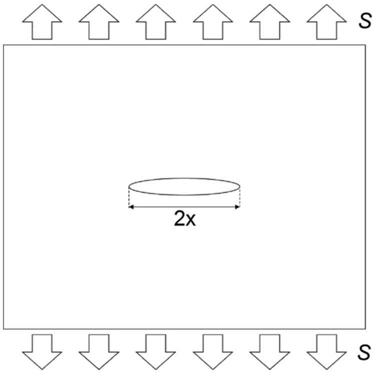


Fig. 1. Simulated thin aluminum plate with a central crack.

A. Specimen

The simulated specimen is shown in Fig. 1. The sinusoidal fatigue load S is responsible for crack propagation, thus the length of the crack 2x increases over time. The load has a peak value of 50 MPa, and a minimum value of 5 MPa (load ratio $R = S_{min}/S_{max} = 0.1$). The starting crack has a 12 mm length, and failure is considered upon reaching a 120 mm long crack.

B. Simulated Measurement System

The observations on the simulated plate are the measures of the crack length itself. These measures are normally distributed with a standard deviation of 1.5 mm heuristically selected. Moreover, the value remains constant during the entire crack evolution to satisfy the hypothesis of the MH algorithm. z being the variable describing the measures, and σ_n their standard deviation, (2) assumes the form in (11),[2] where the symbol $N(\mu, \sigma^2)$ states a normal distribution with a mean μ, and a variance σ^2.

$$z \sim N\left(x, \sigma_n^2\right). \tag{11}$$

C. Analytical Formulation of the Stress Intensity Factor

The stress field affecting the crack tip depends on the crack length itself. Then, the DSS model in (7) requires the relationship between the Stress Intensity Factor range ΔK and the crack length defined by x. The SIF assumes a closed form solution for simple mechanical systems like the thin plate used in this case (12). The crack shape function F is negligible for a large plate with a central crack subjected to fatigue load normal to the crack plane [31]. The SIF range is easily evaluable as the difference between the SIFs at maximum and minimum applied stress $\Delta K = K\left(S_{max}\right) - K\left(S_{min}\right)$ within the same load cycle.

$$K = FS\sqrt{\pi x}. \tag{12}$$

D. Algorithm Performance

To evaluate the capabilities of the prognostic unit defined above, two different simulations are performed: without considering a random noise affecting the simulated crack growth (first simulation), and with considering a random noise (second

[2] If the observations are the direct measures of the crack length, where the only source of uncertainty of the measurement system is represented by the random noise n_k, then $z_k = x + n_k = x_k + N\left(0, \sigma_n^2\right) = N\left(x_k, \sigma_n^2\right)$.

simulation). The first simulation is required to understand the algorithm performance. As a matter of fact, if the random noise is added to the simulated crack propagation $\mathrm{a_k^{sim}}$, the MH algorithm is unable to distinguish two sources of uncertainties: the process noise, and the observation noise. Therefore, the MH algorithm estimates parameters able to fit the data correctly, but being unable to associate the correct variance of the measurement noise, it induces a biased estimation of the FCG parameters. According to this concept, the performances of the system are assessed as follows. The first simulation is performed without the noise affecting the simulated crack growth. The capability of the algorithm in terms of parameter and Residual Lifetime estimation are evaluated. After that, the algorithm is tested considering a random noise affecting the process different from zero. This condition is more realistic in view of applications in real environments, where the real crack evolutions obviously do not follow the exact mathematical model because of the different sources of uncertainties. In this type of simulation, the MH algorithm produces a biased estimation of the noise affecting the measures as well as of the parameters describing the crack growth process due to its inability to account for the random noise of the process. However, the developed prognostic unit remains capable of predicting the Residual Lifetime if the data from the (simulated) measurement systems are properly fit. Both simulation results are compared with the results of the particle filter built with the deterministic DSS model applied to the same crack simulations. The differences between the two algorithms are discussed at the end of each section in terms of RL prediction capability.

1) Simulation With Null Process Noise: As explained above, the linear damage accumulation model without random noise (13) drives the simulated crack growth. Obviously, the process noise $\mathrm{v_k}$ is always used in the particle filter formulation (14) to correctly approximate the posterior probability of the state given the observations.[3] The process noise adopted in particle filtering, and the measurement uncertainty are reported in (16), and (17) respectively. Fig. 2 shows the simulated crack with respect to the measures obtained through (15). A crack growth model (13) produces a theoretic crack growth, to appreciate the parameter estimation capabilities of the MH algorithm. The NASGRO parameters used to simulate the crack growth are different from the expected values for the aluminum alloy Al 2024-T6. In particular, $\mathrm{C} = 1.39\mathrm{e}-11$ ($\log\mathrm{C} = -25$), $\mathrm{m} = 3$ instead of the averages of the alloy Al 2024-T6; that is, $\mathrm{C} = 2.383\mathrm{e}{-}12$ ($\log\mathrm{C} = -26.763$), and $\mathrm{m} = 3.2$. The distributions associated to these parameters come from Virkler's data [29], and have been regularly used for FCG analyses of the Al 2024 alloys (see for instance [4], [15], and references therein). Because Virkler's data are related to the alloy Al 2024-T3, while the alloy under discussion is Al 2024-T6, the variance of the parameters is modified, keeping the Coefficient of Variation (CoV), defined as $\mathrm{CoV} = \sigma/\left|\mu\right|$, constant. The distributions of the two parameters become $\log\mathrm{C} \sim \mathrm{N}(-26.763, 0.9966)$, and $\mathrm{m} \sim \mathrm{N}(3.2, 0.0346)$, for the logarithm of C, and m, respectively. Moreover, the two parameters are highly correlated, as clearly demonstrated in statistical analyses of FCG data. The correlation coefficient is close to -1 ($\rho_{\log\mathrm{C,m}} \approx -0.9979$) [30], thus the covariance becomes $\sigma_{\log\mathrm{C,m}} = -0.1853$. The noise used to create the SDSS implemented in the particle filter is $\sigma_\mathrm{v} = 0.89$, while the noise affecting the measures is set to $\sigma_\mathrm{n} = 1.5$.

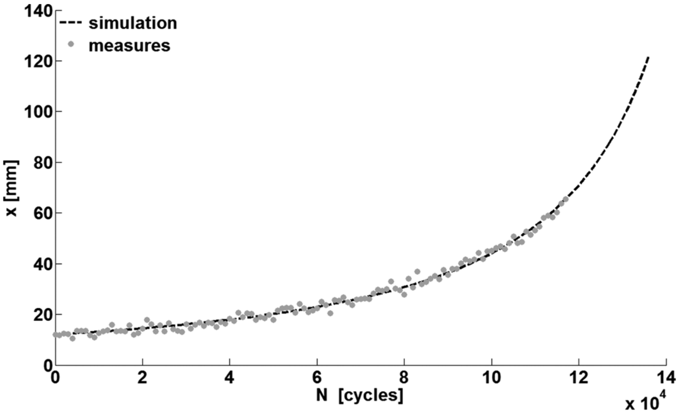


Fig. 2. Crack growth simulation according to null process noise and measures with non-zero random noise.

$$\mathrm{a_k^{sim}} = \mathrm{a_{k-1}^{sim}} + \Delta\mathrm{N}\left.\frac{\mathrm{da}}{\mathrm{dN}}\right|_{\mathrm{a}=\mathrm{a_{k-1}^{sim}}} \tag{13}$$

$$\mathrm{x_k^i} = \mathrm{x_{k-1}^i} + \Delta\mathrm{N}\left.\frac{\mathrm{dx}}{\mathrm{dN}}\right|_{\mathrm{x}=\mathrm{x_{k-1}^i}} \mathrm{v_{k-1}} \tag{14}$$

$$\mathrm{z_k} = \mathrm{a_k^{sim}} + \mathrm{n_k} \tag{15}$$

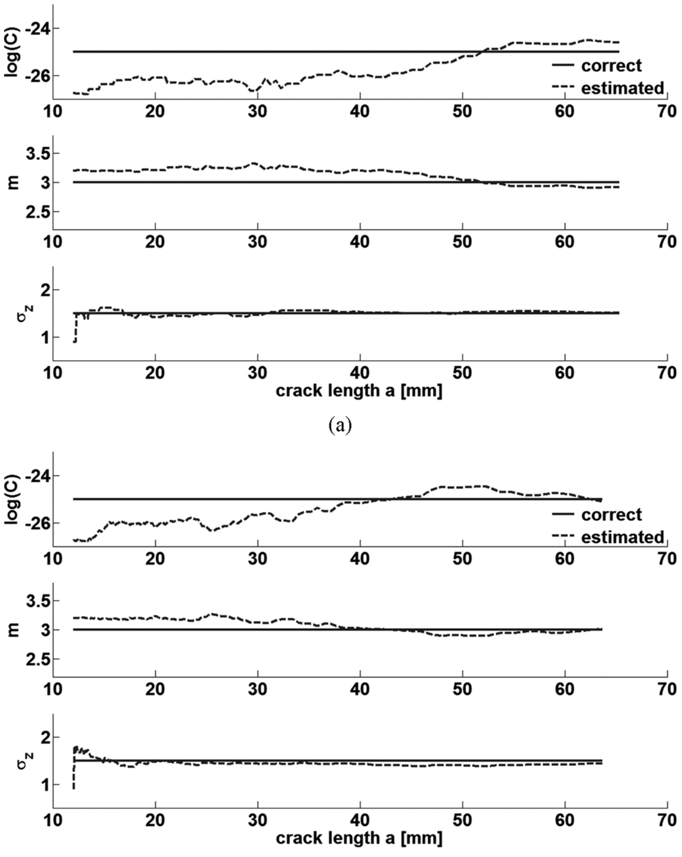
$$\mathrm{v_{k-1}} \sim \log\mathrm{N}\left(\mu_\mathrm{v}, \sigma_\mathrm{v}^2\right) \tag{16}$$

$$\mathrm{n_k} \sim \mathrm{N}\left(0, \sigma_\mathrm{n}^2\right). \tag{17}$$

The incongruity among the expected parameters $\log\mathrm{C}$, m, and σ_V, with respect to the parameters used to simulate the crack, initially produces incorrect Residual Lifetime predictions that are subsequently corrected during the crack propagation due to the estimation performed by the MCMC algorithm. Because the particle filter with a deterministic DSS model is unable to take into account these discrepancies, better performance is expected from the proposed prognostic unit with respect to the standard formulation of the algorithm. To produce an unbiased swarm of crack evolutions via the particle filter method, the lognormal distribution in (16) must have a unitary expected value μ_v. The noise parameters are more easily defined by means of the relationship between the normal and the lognormal distribution. Let us consider a random variable $\omega \sim \mathrm{N}\left(\mu_\omega, \sigma_\omega^2\right)$ such that $\mathrm{e}^\omega = \mathrm{v}$. Because the noise v multiplies the FCG rate (13), (14), the expected value of v must be one to produce an unbiased estimation of the crack growth evolution. This unitary mean of the lognormal process v is guaranteed if (18) is satisfied. Consequently, once the variance of the Gaussian random process has been selected, the mean of the Gaussian random process is driven by the formulation in (18).

The estimation of the parameters performed by the MH algorithm is presented afterwards (Fig. 3). Fig. 3(a) shows the MH output when included in a SIS algorithm, with resampling activated when the number of particles with non-zero weights drops below 70% of the total (equal to 1000). Every time resampling

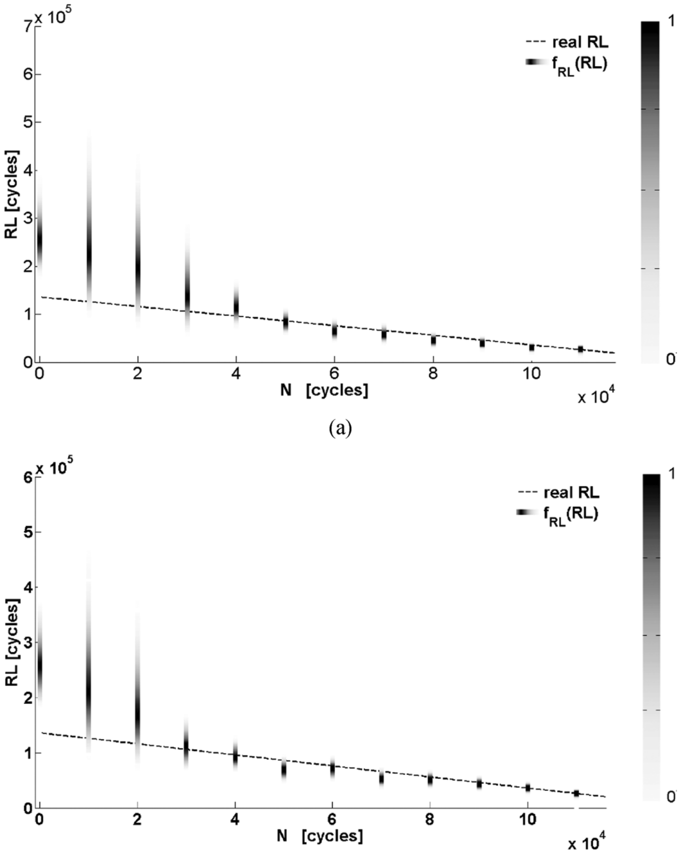
[3] For the sake of clarity, we stress the fact that no process noise is introduced in the simulated crack propagation that has to be prognosticated, while the noise in (14) and (16) is used to generate the SDSS.



(a)

(b)

Fig. 3. Parameter estimation using (a) SIS, and (b) SIR.



(a)

(b)

Fig. 4. RL prediction via (a) SIS, and (b) SIR algorithms with zero process noise on the target crack growth.

is performed, the MH algorithm estimates a new PDF of the quantities of interest. Fig. 3(b) shows the case of SIR in which the MH algorithm is run every time a new measure becomes available.

$$\mu_\omega = -\frac{\sigma_\omega^2}{2}. \tag{18}$$

The algorithm requires a relatively large number of measures to assess the discrepancies between the expected values of the parameters and the real one.

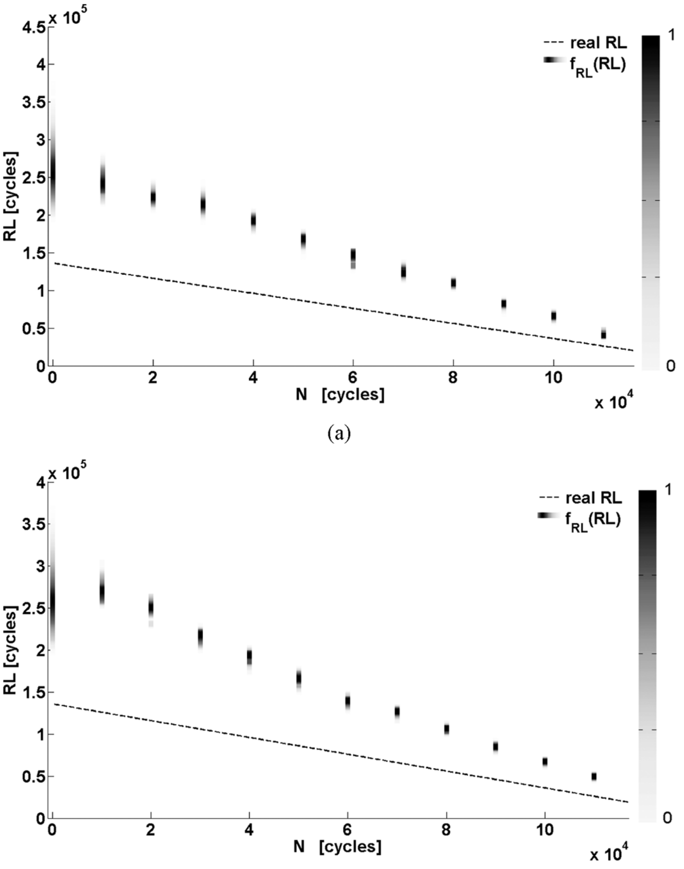
The RL prediction of the prognostic algorithm is shown in Fig. 4, in which the prediction of the RL converges to the correct number of residual load cycles during the crack evolution using both the SIS and the SIR algorithms. The bar on the right indicates a normalized PDF with its unitary peak. On first approximation, the high computational time required from the SIR algorithm seems excessive. However, the results are comparable in these simulations, where a high number of measures are available (simulating, for instance, an automatic measurement system). The SIR algorithm is considerably more accurate if only a few observations of the crack are available, as is the case for manual measurement systems.

Fig. 5 highlights the poor performances of the standard particle filter. As expected, the inability to account for the different model parameters produces incorrect RL predictions, both for the SIS and the SIR algorithms. The better accuracy of the results of the suggested algorithm are proven by the absolute error between the expected and the target Residual Lifetimes in Fig. 6. The algorithms start from the same error due to the discrepancy between the actual parameters of the simulation and the expected parameters for the Al 2024-T6; however, the prognostic system composed of the particle filter method and the MH algorithm recognizes the bias affecting the model parameters, and converges to the correct prediction. The standard particle filter remains unable to predict the correct RL for the entire life of the plate.

The robustness of the algorithm should be proved, introducing additional performance indices rather than the simple error between the actual RL and the estimation, according to the literature on prognostic performance [32]. Here, the *average bias* B is selected as a performance index for the proposed algorithm. According to [32], the average bias is calculated through (19).

$$\mathrm{B} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{EOP}} \left(\widehat{\mathrm{RL}} - \mathrm{RL}\right)_{\mathrm{i}}}{(\mathrm{EOP} - \mathrm{P} + 1)} \tag{19}$$

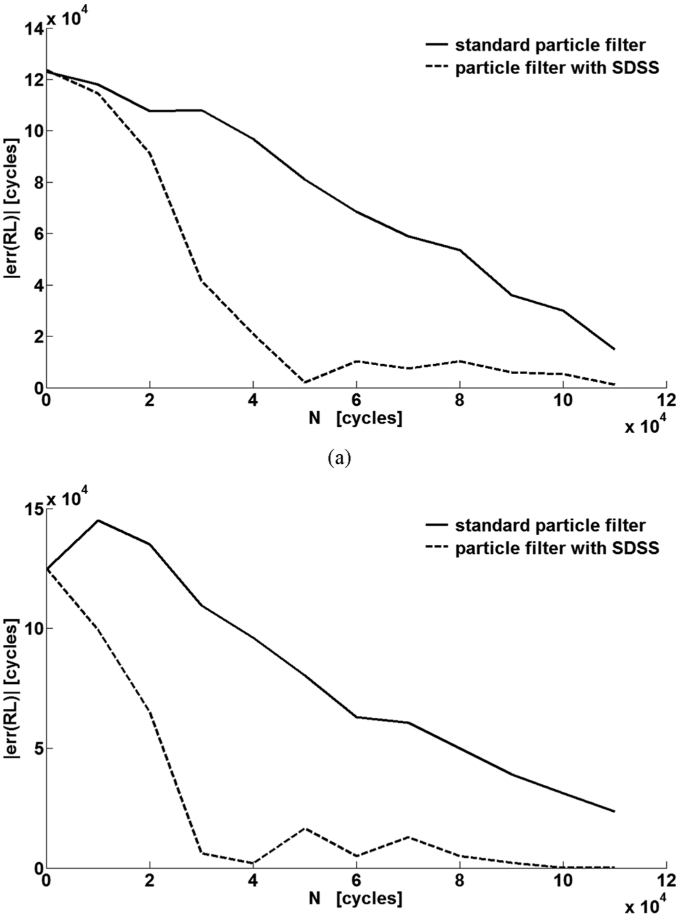
where P is the time index at which the algorithm starts, and i represents the i-th estimation of the RL. The average bias of a perfect prediction approaches 0; the higher the error of the prediction, the higher the value of B (towards $+\infty$ or $-\infty$). According to (19), a RL prediction with equal alternate errors with respect to the target is considered good, as the system is able to describe the target evolution without biases (for this reason, (19) does not consider the absolute value of the error at each step).



(a)

(b)

Fig. 5. RL prediction of a standard particle filter using (a) SIS, and (b) SIR algorithms with zero process noise.



(a)

(b)

Fig. 6. Error between the target and the expected RLs; comparison between standard particle filter and proposed algorithm in case of null process noise for (a) the SIS, and (b) the SIR algorithms.

TABLE III
AVERAGE BIAS B.

Algorithm	SIS	SIR
Standard particle filter	6.9E+04	7.4E+04
Particle filter & MH	2.7E+04	1.9E+04

The average biases related to the crack growth simulations are presented in Table III. As visible, the proposed algorithm with adaptation of the model parameters has a considerable reduction of the B index with respect to the standard particle filter.

Of course, this preliminary analysis of the algorithm performances is not exhaustive, and further investigation on other performance indices is mandatory. However, it could be considered a good starting point to highlight the importance of parameter adaptation in the framework of FCG, although the model parameters are well studied, and a large number of propagation models are available in literature.

2) Simulation With Non-Zero Process Noise: This paragraph highlights the performance of the algorithm in the presence of a non-zero process noise in the simulated crack evolution. According to the formulation of the previous paragraph, the crack simulation accounts for a process noise $\mathrm{u} \sim \mathrm{N}\left(\mu_{\mathrm{u}}, \sigma_{\mathrm{u}}^2\right)$ which alters the crack growth simulation in (13) that has to be prognosticated. Two different random noises affect the process: the process noise u modifying the crack simulation that should be included in the random noise v defined in the particle filtering algorithm, and the noise affecting the measures that should be estimated by the MH algorithm during the resampling step.

Equation (20), together with (14) through (17) governs the crack growth simulation and the RL prediction. Fig. 7 shows the simulation of the crack according to (20), and the measures obtained through (15).

$$\mathrm{a}_{\mathrm{k}}^{\mathrm{sim}} = \mathrm{a}_{\mathrm{k-1}}^{\mathrm{sim}} + \Delta \mathrm{N} \left.\frac{\mathrm{da}}{\mathrm{dN}}\right|_{\mathrm{a}=\mathrm{a}_{\mathrm{k-1}}^{\mathrm{sim}}} \mathrm{u}_{\mathrm{k}}. \tag{20}$$

As visible, the simplified crack evolution obtained by means of the linear damage accumulation model is modified by random values driven by the process noise characteristics. This procedure approaches the real case in which the crack growth evolution is clearly altered by several sources of uncertainties. Fig. 8 shows the parameter estimation of this case using a SIS algorithm (a), and a SIR algorithm (b). As clearly visible, the MH produces biased estimations of the parameters, particularly in the case of the SIS technique. The bias slightly decreases using the SIR algorithm; however, the standard deviation associated to the measure clearly differs from the correct

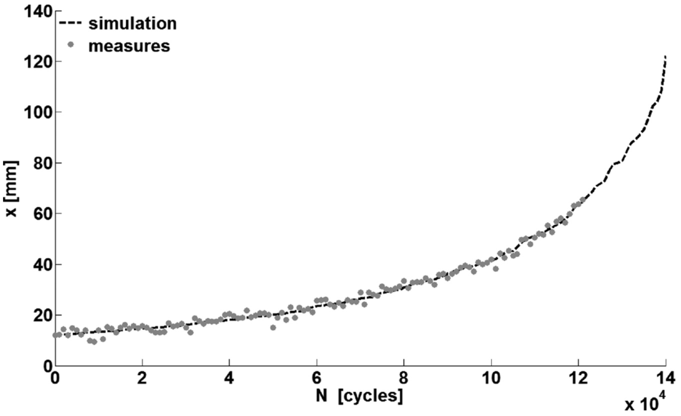
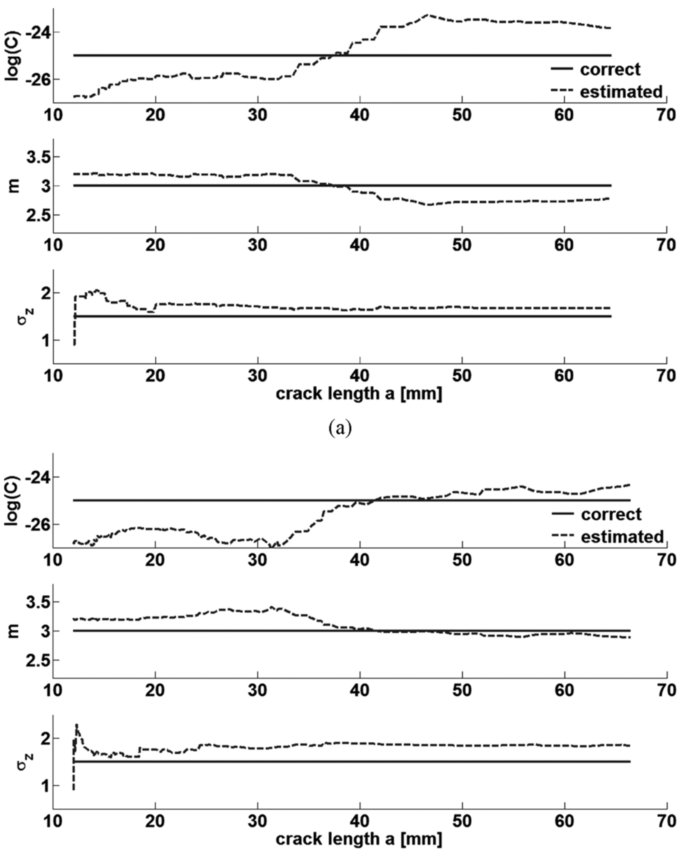


Fig. 7. Crack propagation simulation with non-zero process noise.



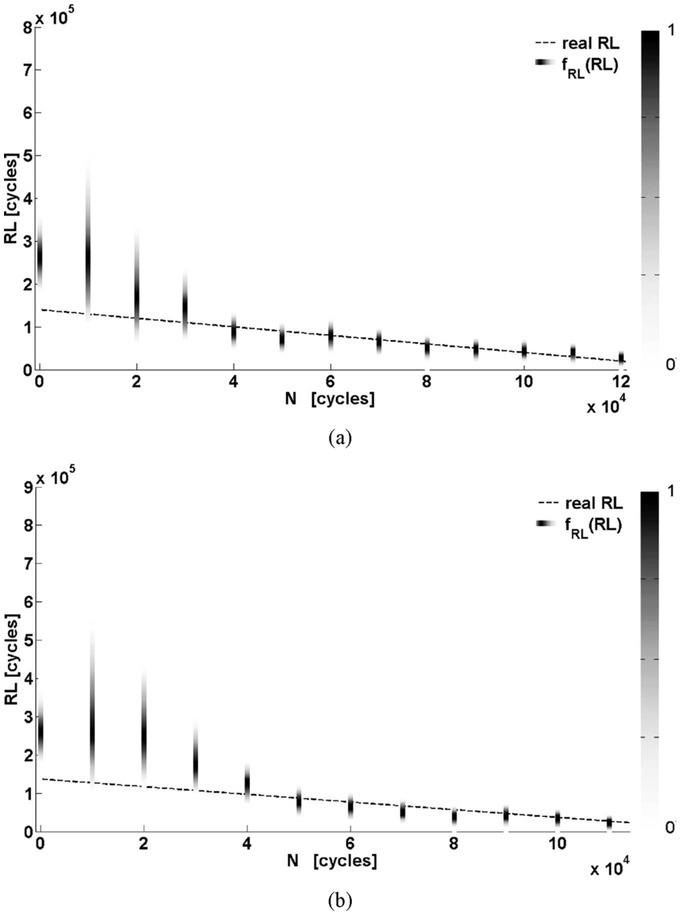
(a)

(b)

Fig. 8. Parameter estimation using (a) SIS, and (b) SIR technique.

deviation, as confirmed by several simulations each obtaining very similar results. The reason behind the incorrect parameter estimation is the inability of the MH algorithm to account for two sources of uncertainty: the process, and the measurement noises. The Markov chain associates to the crack measures $z_{0:k}$ a combination of noise from the actual noise of the measures and the process noise u.

Nonetheless, a quantitative analysis of the estimated noise should be made to prove the incapacity of the method to predict the correct noise. Despite the bias in the parameter estimation provided by the MH method, the RL predictions associated to this kind of simulations are accurate, as highlighted in Fig. 9.



(a)

(b)

Fig. 9. RL prediction using (a) SIS, and (b) SIR algorithms with a non-zero process noise on the target crack growth.

The RL predictions of this case are comparable with the predictions of the previous case made without the process noise in the crack growth simulation. The RL estimation capabilities are preserved due to the particle filter formulation, containing the intrinsic uncertainties of the process v, and the uncertainties of the measures n. According to these results, the prognostic system appears to be able to estimate the correct Residual Lifetime of a structure subjected to FCG, even when the estimation of the underlying model parameters and the noise associated to the measures are affected by some bias.

The standard particle filter, in which the model parameters remain constants, continues to produce incorrect RL estimations (Fig. 10). The comparison of the results points out the higher performance of the proposed system, confirmed by the absolute error affecting the lifetime prediction shown in Fig. 11. As visible, the suggested algorithm quickly converges to the correct RL.

The average bias B introduced in the previous section as a performance index of the prognostic algorithm is visible in Table IV. These results verify the outcome of the previous simulation without the process noise on the target crack propagation. Therefore, the proposed algorithm is tested hereafter on real portions of aeronautical structures to validate the method.

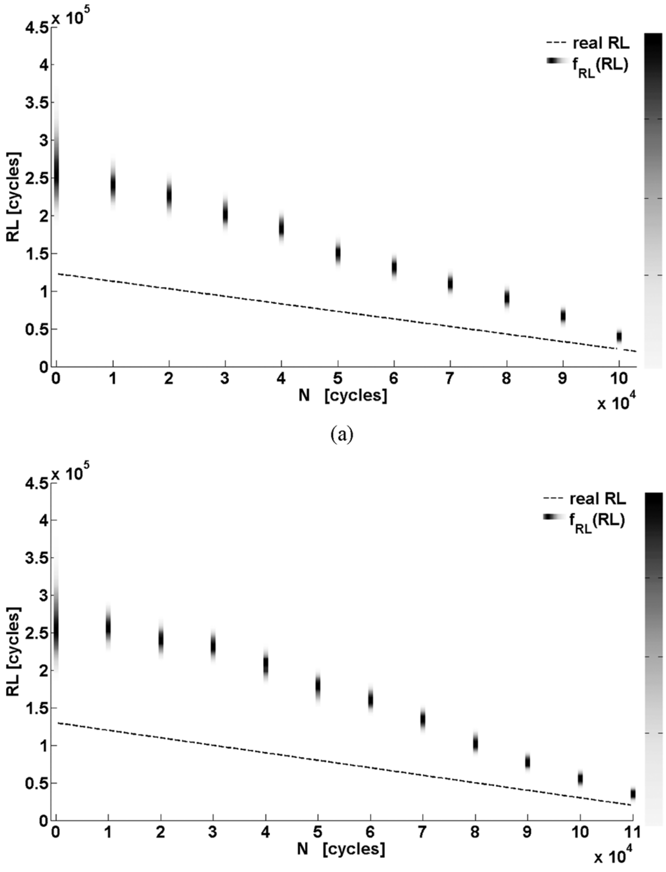


Fig. 10. RL prediction of a standard particle filter using (a) SIS, and (b) SIR algorithms with non-zero process noise on the target crack growth.

V. Extension to Real Applications

In this section, the prognostic unit developed via particle filtering and MCMC algorithms is used to monitor real Fatigue Crack Growth within a real helicopter panel structure subjected to sinusoidal fatigue load in a laboratory environment.

A. Specimen and Test Rig

A helicopter panel with a central artificially induced crack is used as a test structure. The panel skin is made of the same aluminum alloy as previously selected for the simulated plate (Al 2024-T6), while the four stringers used to stiffen the structure are made of Al 7075. Thus, the parameter PDFs assume the same shape and values of Section IV.D. The panel has been rigidly grounded to its lower end, and is connected to the actuator through its upper end (Fig. 12). The crack is artificially initiated in the center of the skin to guarantee the repeatability of the test. The initial length of the notch is 16 mm. A sinusoidal load is applied in the vertical direction with a maximum peak of 35 kN, and a load ratio $R = 0.1$. More details about the experimental set up are reported in [1], and [33]. The relevant features of the test are reported in Table V. Four different tests have been performed to evaluate the robustness, and the reliability of the prognostic system. The sensors applied on the panel visible in Fig. 8 are Fiber Bragg Gratings for strain field measurements. Though the data acquired from these sensors have not been directly adopted in the prognostic framework, strain measures have been used to validate the numerical model [33] introduced in Section V.C below, which is used to simulate SIFs as a function of crack length and position. The experimental crack propagation obtained during the four FCG tests are shown in Fig. 13. The crack length data have been used as an input to the prognostic unit to monitor the crack, and predict the RL of the panels. Also, in this case the critical crack length is assumed to be 120 mm, which is less than the distance between two subsequent stringers.

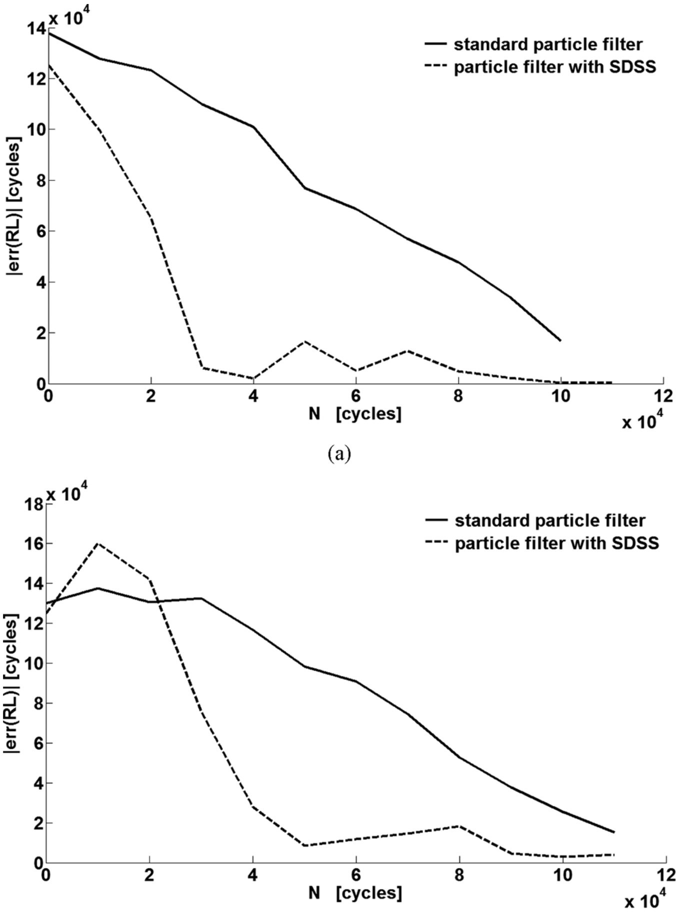


Fig. 11. Error between the target RL and the expected RL; comparison between standard particle filter and proposed algorithm in case of non-zero process noise for(a) the SIS, and (b) the SIR algorithms.

TABLE IV
Average Bias B

Algorithm	SIS	SIR
Standard particle filter	7.5E+04	8.02E+04
Particle filter & MH	2.3E+04	3.58E+04

B. Measurement System

The crack measures have been obtained by means of a manual caliper. As just explained above, these measures constitute the input of the prognostic unit, that is the values $z_{0:k}$. These measures contain the uncertainties represented by the intrinsic uncertainty of the caliper plus the uncertainty related to the human

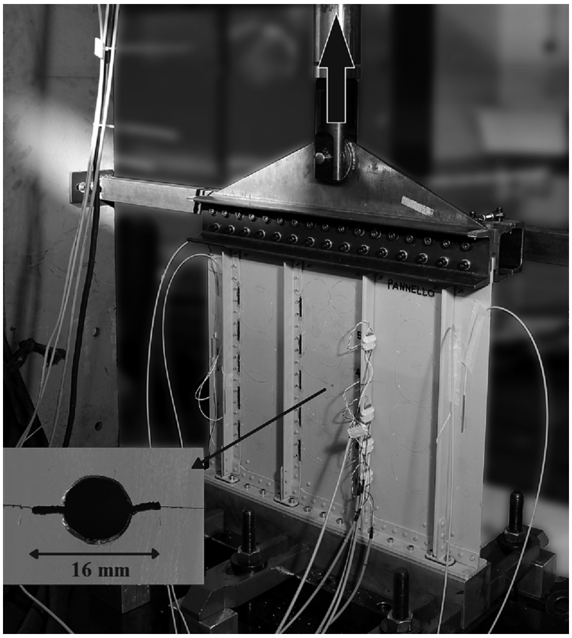


Fig. 12. Complete test rig. The vertical arrow indicates the applied load, while the zoom shows the notch used to artificially-induce crack propagation.

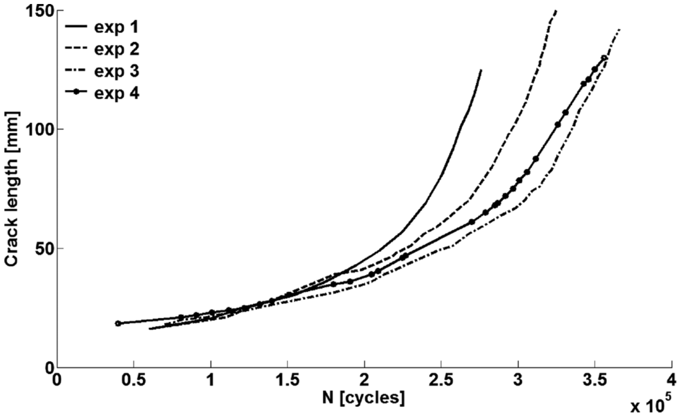


Fig. 13. Experimental crack propagations obtained from four different helicopter panels.

TABLE V
PARAMETERS OF CRACK PROPAGATION TESTS

Load shape	Sinusoidal
Load frequency	12 Hz
Maximum applied load	35 kN
Load ratio (R)	0.1
Damage type	Skin crack
Damage position	Central on central bay
Damage initiation	Artificial, 16 mm
Skin material	Aluminum 2024-T6

error during the measure reading. Every time a measure is provided to the algorithm, a RL prediction is performed. In this context, the measurement system is described by a s-normal PDF with a standard deviation of 3 mm within the particle filter unit. The value remains the same for the whole crack evolution up to the critical crack length. This simplifying hypothesis is required by the MH algorithm to correctly evaluate the variance associated to the measures; in fact, the common MCMC algorithms are unable to account for time-varying noise variance.

C. *Numerical* ΔK *Estimation*

This section focuses on the problem of estimating the SIFs at the crack tips. Most of the studies reported in the literature about FCG prognosis methodologies are applied to simplified (though sometimes realistic) structures. In realistic scenarios, the analytical closed form solution for the SIF calculation is not always available. Equation (11) was used before to provide an indication about the SIFs for an infinite thin plate subject to FCG. If complex structures are considered, as in the case under examination, numerical models can be adopted to fit the complex non-linear function that relates the damage parameters (crack length and position) and the load to the SIFs. A database of SIFs relative to various crack lengths over the considered structure is provided here as an input during the training of an ANN structure. The description of the Finite Element model used to calculate SIFs, the database of numerical SIFs, and the ANN structure used to estimate SIF parameters as a function of crack position and length are reported in detail below.

1) Finite Element Model: The commercial software ABAQUS 6.9 was used to model the real structure behavior, and to calculate the strain distribution as well as the SIFs at the crack tip. The FEM is shown in Fig. 14. Structural components have been modeled with quadratic shell elements. Three-axes springs have been used to model all the riveted connections among the skin, the stringers, and the reinforcing elements. To provide a realistic simulation of the experiments, also the connection elements to the ground and to the actuator have been modeled. Load is applied vertically, and the upper region of the panel is designed to transfer this load to both the stringers and the skin, reproducing the real structure behavior. The model has been experimentally verified in the area shown in Fig. 14(b), where cracks have been simulated by introducing a discontinuity between adjacent elements along the direction perpendicular to the stringers. Crack direction has been kept constant and perpendicular to the maximum principal direction, thus assuming *mode I* propagation. An array of singular elements has been used to more accurately estimate the stress field singularity that is present at the crack tips. Two SIFs have been calculated for each simulated damage case, producing a database dependent on the damage position and the length used in the prognosis evaluation. The interested reader can refer to [1] and [33] for a detailed description of the verification and validation procedure of the FEM, both in damaged and healthy conditions.

2) Database Generation: The same database of damages previously used by the authors in [1] to perform damage diagnosis based on strain field is used here for the prognosis based on numerical SIFs. The numerical results are used in this study to approximate the function that relates the crack parameters (center position and length) and the load to the SIFs at its tips. In particular, the database contains a total of 1700 damage cases. Crack lengths from 20 mm to 100 mm, with a 5 mm step, have been simulated. 100 crack center positions have been considered at each crack length level, randomly locating the crack center within the validated region in Fig. 14(b). The two SIFs corresponding to the two crack tips used for the prognosis of the crack evolution have been numerically calculated for each damage case while a static 35 kN load has been applied vertically (corresponding to the peak of the sinusoidal fatigue load),

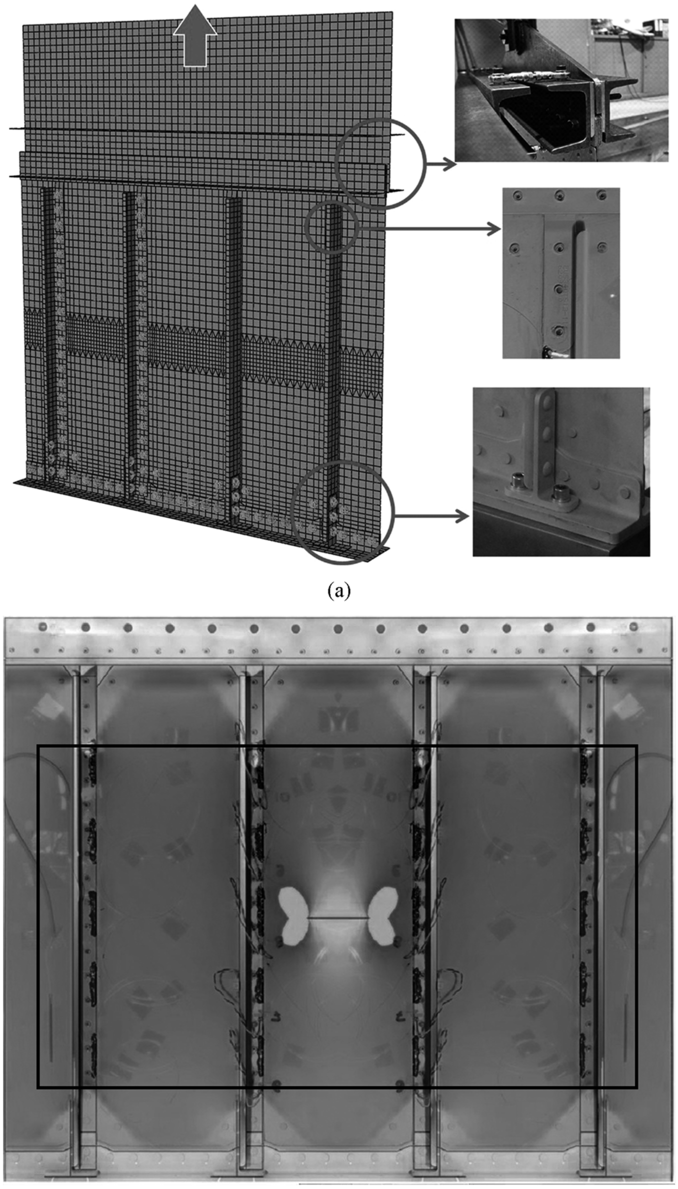


Fig. 14. FE model for SIF calculation: (a) FE model details to reproduce the entire test rig, and (b) simulated strain field distribution (vertical direction) corresponding to a crack in the center of the panel.

as shown in Fig. 14(a). Nevertheless, under the assumption of model linearity, the SIF magnitude for different loads can be simply obtained via multiplication by a factor dependent on the load level.

It is important to consider that, though the position of the crack center is constant in all four experimental FCG tests, the methodology under study remains valid and applicable for any crack center position within the validated region.

3) Artificial Neural Network for $\Delta\mathrm{K}$*:* A machine learning approach based on ANNs is used in this work to provide estimates of SIFs at the crack tips. In particular, the ANN used here is the Multi-Layer Perceptron (MLP), which uses a multi-layer feed-forward structure. It consists of a collection of connected nodes, namely the input layer nodes, hidden layer nodes, and finally the output nodes. In this present work, the input nodes correspond to the crack parameters (namely crack center position, and crack length), one hidden layer is used for computation, and the outputs are the prediction of the SIFs at the left and right tips of the crack. It is not our intent to enter into the details of the theory of ANNs and their mathematical formulation, as it is widely discussed in the relative literature [34]. It is however important to specify the shape of the activation functions that have been used. In the hidden layer nodes, it is restricted to $\mathrm{h}\left(\mathrm{x}\right) = \tanh\left(\mathrm{x}\right)$, while the linear activation function should be considered for the output nodes, as SIF estimation a mainly a regression problem. The first stage of using a network to model an input-output system is to establish the appropriate values for the connection weights. This stage is the *training* or *learning* phase. The type of training adopted here is a form of supervised learning, and makes use of a set of network inputs for which the desired network outputs are known from numerical simulations. At each training step, a set of inputs is passed forward through the network yielding trial outputs that can be compared with the desired outputs. The *Scaled Conjugate Gradient* algorithm [35] has been used here to minimize the comparison error, thus optimizing the network synapses weights. Once the comparison error is reduced to an acceptable level over the whole training set, the training phase ends, and the network is established. The networks used for this study have been designed and trained using NETLAB [35] functions, based on MATLAB code. Various regularization techniques are available to guarantee sufficient generalization capabilities for the ANN. *Early-stopping* based on a validation set [34] has been used here. In fact, the entire dataset has been randomly split into two subsets, namely training, and validation, containing 80%, and 20% of the data respectively. The use of a *committee* of ANNs has also been considered here as an aid to generalization. The simplest committee consists of averaging the output of a set of individual ANN models.

Different ANN models can be created by maintaining the ANN structure unchanged but training it with different datasets. However, in practice, only one single dataset is available (the numerical database), and a way to introduce variability between the different models is to use *bootstrap* datasets [34]. Each ANN belonging to the committee is trained and validated with a different dataset, randomly selecting those test cases that fall into the training and validation sets, however using all the data available inside the initial domain. This technique has been adopted here to train 50 ANNs for SIF estimation. Each ANN has the same structure, optimized as described in the following, but it is trained and validated with a different dataset. Defining the optimal structure of an ANN is complex because of the number of variables involved during training, often correlated with each other. Some early trials with different numbers of hidden layers were performed to test for an improved algorithm performance. However, no significant improvement was found, and one single hidden layer will be considered from now on. *ANalysis Of VAriance* (ANOVA) has been used to optimize the number of nodes inside one single hidden layer. The Root Mean Square Error (RMSE) based on the validation set has been calculated for 50 ANNs trained on different subsets. Increasing the number of hidden nodes has been considered, and a PDF of RMSE is available at different levels of hidden layer node number. No benefit has been obtained from increasing the number of the

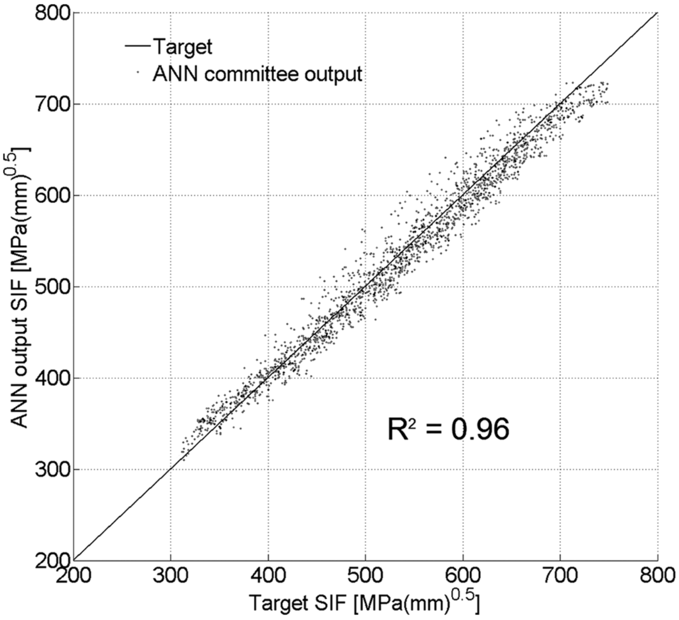


Fig. 15. ANN committee output comparison with target output.

TABLE VI
PARAMETERS FOR ANN STRUCTURE OPTIMIZATION AND TRAINING

ANN type	Function fitting MLP
Input layer	3 nodes (x-y coordinates of crack center and crack length)
Output layer	2 nodes (left and right SIFs)
Hidden layer nodes	15 nodes
ANN # in committee	50 ANNs
Training strategy	Scaled Conjugate Gradient
Regularization techniques	Early stopping and average ensemble committee

hidden nodes above 15, and thus the optimized ANN structure contains one hidden layer with 15 computational nodes. A detailed explanation of the procedure adopted for ANN optimization is available in [36]. A summary of the parameters associated to the ANN structure and the training procedure is reported in Table VI. The algorithm performance on the numerical database can be appreciated in Fig. 15. Each mark indicates the output of the ANN committee, which is the average of the 50 outputs from 50 ANNs with the same optimized structure when the same input is provided. The target reference line is also indicated. The RMSE of the committee output with respect to the target was found to be equal to 31 $\mathrm{MPa}\sqrt{\mathrm{mm}}$. Fig. 15 shows that a good fit was obtained for the largest portion of the database. However, some variance in the estimation of the SIFs associated to the cracks at the boundaries of the domain, and in some cases for the crack tips below the stringers, is visible. This result is due to the higher complexity of the relation between SIFs and crack parameters.

D. Algorithm Performance

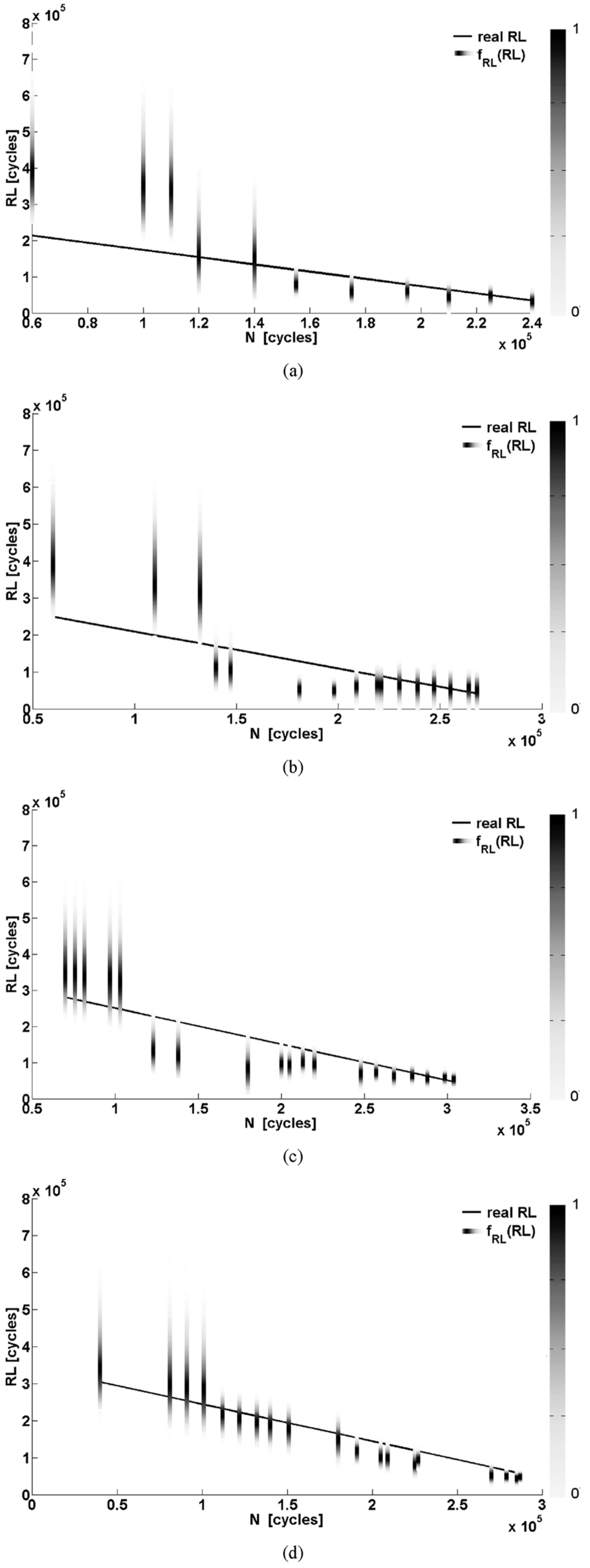
The Residual Lifetime predictions of the algorithm for all the tests are shown next. As for the simulated case, the predicted RL is shown in terms of normalized probability density functions, and the real RL.

Figs. 16(a)–(d) and 17(a)–(d) show the results of the SIS, and SIR algorithms, respectively. The prognostic unit produces some inaccurate estimations of the RL, as shown in Figs. 16(b) and 17(b), close to 180,000 load cycles. The crack measures after 180,000 load cycles help the algorithm to approach the correct RL. The poor predictions are caused by the trend of the crack propagation visible in Fig. 13 (dashed black line, experiment 2). This trend is related to experiment 2, and is affected by a sudden slope change close to $N = 180,000$ cycles. Therefore, the fitted parameters up to that point produce a swarm of possible crack evolutions markedly different from the future crack evolution, when the crack seems to slow down. Considering test 4, in which harsh slope changes in the crack propagation are absent, the prediction is more robust. Despite some of the discrepancies between the expected and the real RL, the prognostic unit shows an overall good, robust behavior. The predictions of the algorithm with deterministic DSS are shown in Figs. 18 and 19 for the SIS, and SIR algorithms respectively. As expected, the results of the standard particle filter are distant from the target RL for the majority of the tests. The RL confidence boundaries do not include the target lifetime in most of the algorithm operations, sometimes producing unexpected shapes of the PDFs because of the high discrepancy between the expected trends of the crack growth (driven by the deterministic values of the model parameters) and the observed measures. Nevertheless, a sufficiently accurate result is provided by the particle filter with deterministic DSS for experiment 4 (Fig. 13, dotted black line). In this case, a good RL prediction is obtained at the first operation of the algorithm, and the PDF of the RL includes the actual RL for the entire crack propagation. This good behavior is driven by the proximity of the first prediction to the actual RL, meaning that the average parameters of the Al 2024-T6 produce a swarm of possible crack propagations close to that particular experimental propagation.

However, this coincidence does not happen in other tests. As a matter of fact, the inability of the standard DSS model to adjust the possible crack evolutions strongly reduces the robustness of the system, compared with the suggested prognostic unit composed of a particle filter with a Stochastic DSS, sequentially updated via the MH algorithm. The absolute error of the two algorithms (standard particle filter, and proposed particle filter with MH) is visible in Fig. 20. The comparison of the absolute errors shows the highest performance of the proposed method. The values of the average bias B for each panel, and of the global average bias (calculated as the mean for all the panels), highlight that the adaptation of model parameters is fundamental to produce a reliable, robust lifetime prediction for real structures (Table VII).

VI. CONCLUSIONS

The presented work analyzes a prognostic unit composed by a Sequential Monte Carlo algorithm combined with a Markov chain Monte Carlo technique. The Dynamic State-Space model usually implemented in a particle filtering framework becomes an adaptive Dynamic State-Space model; it is built on the PDFs of the FCG parameters C and m, producing the Stochastic DSS presented in [15], and it is updated through the MH algorithm. The adaptive DSS forms the particle filtering framework for



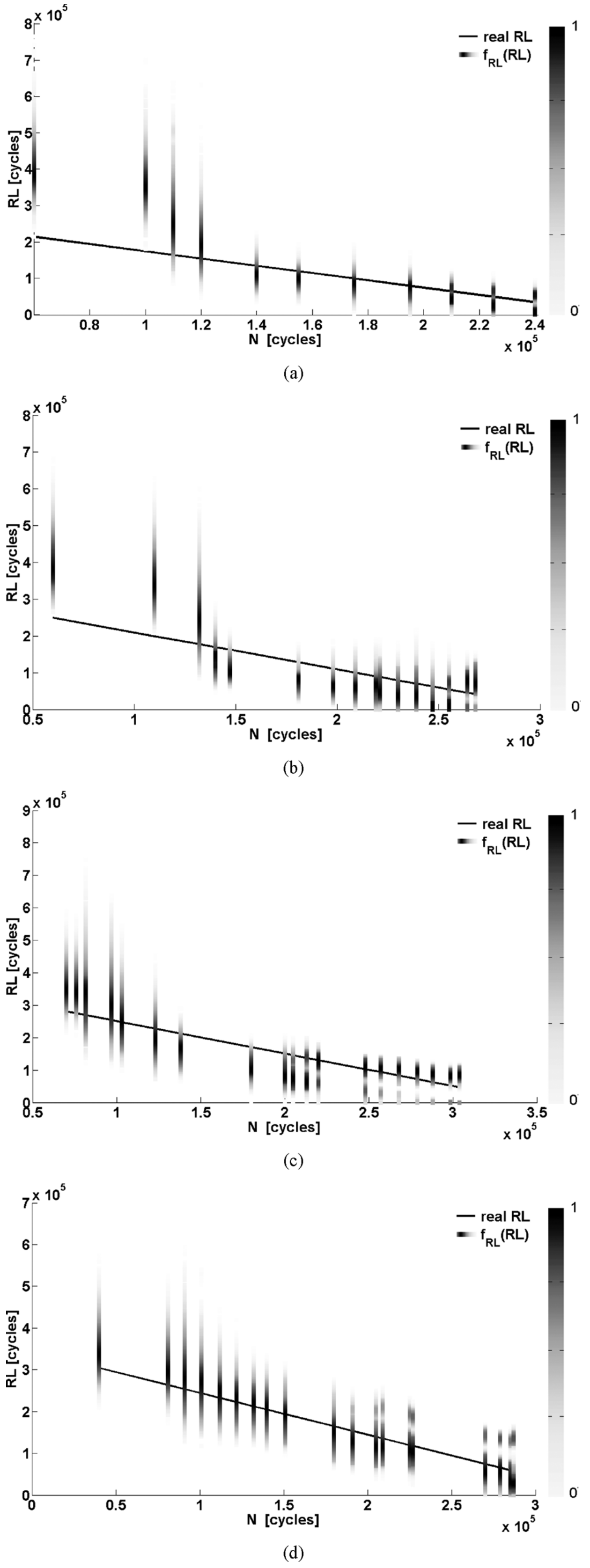
(a)

(b)

(c)

(d)

Fig. 16. RL prediction for the 4 tests using a SIS algorithm.



(a)

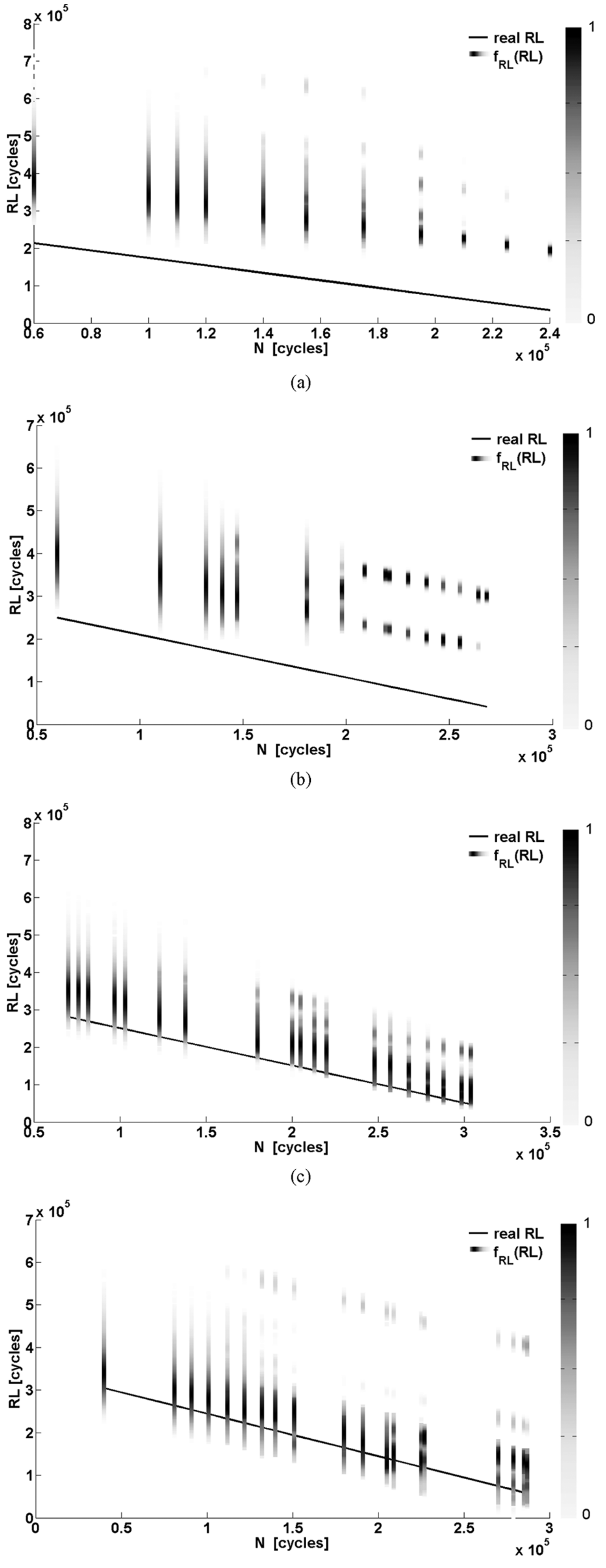
(b)

(c)

(d)

Fig. 17. RL prediction for the 4 tests using a SIR algorithm.

TTF and RL prediction. The measure and the environmental uncertainties have been taken into account inside the random noises commonly implemented in the particle filter, and divided into process noise and observation noise. The MH algorithm



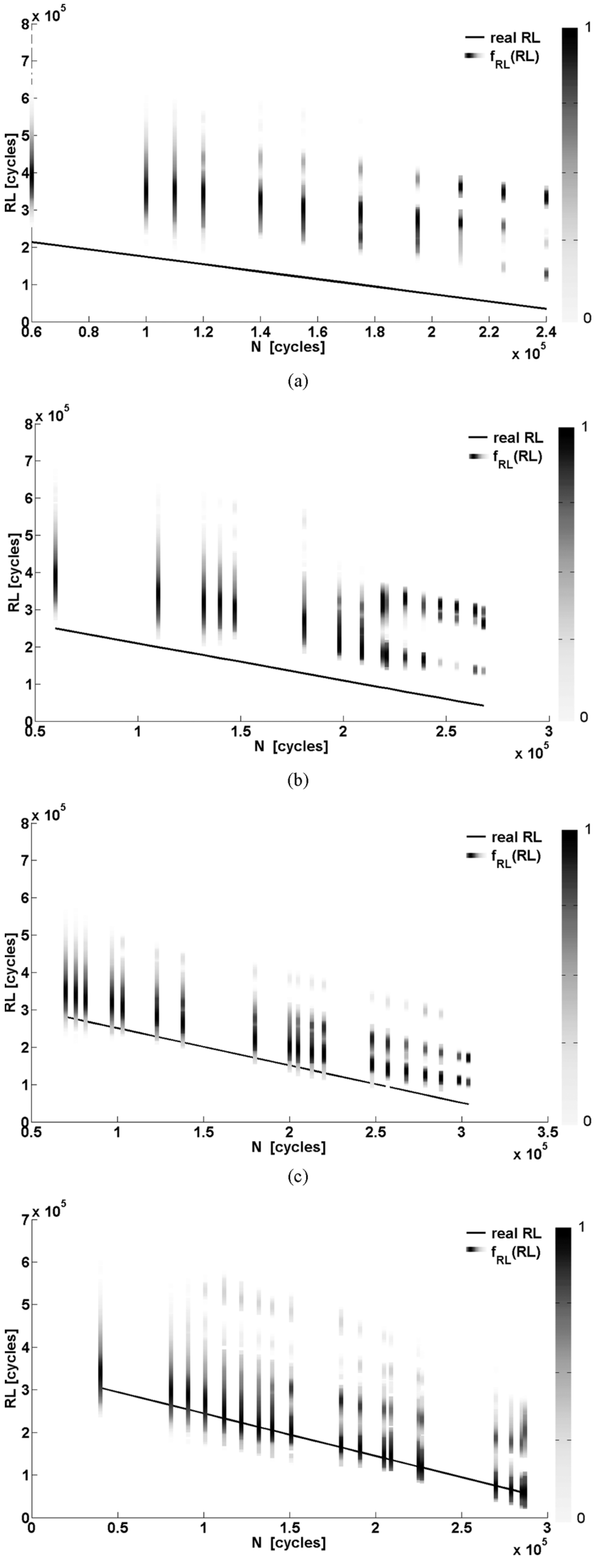
(a)

(b)

(c)

(d)

Fig. 18. RL predictions for the 4 tests using a SIS algorithm with deterministic DSS (standard particle filter formulation).



(a)

(b)

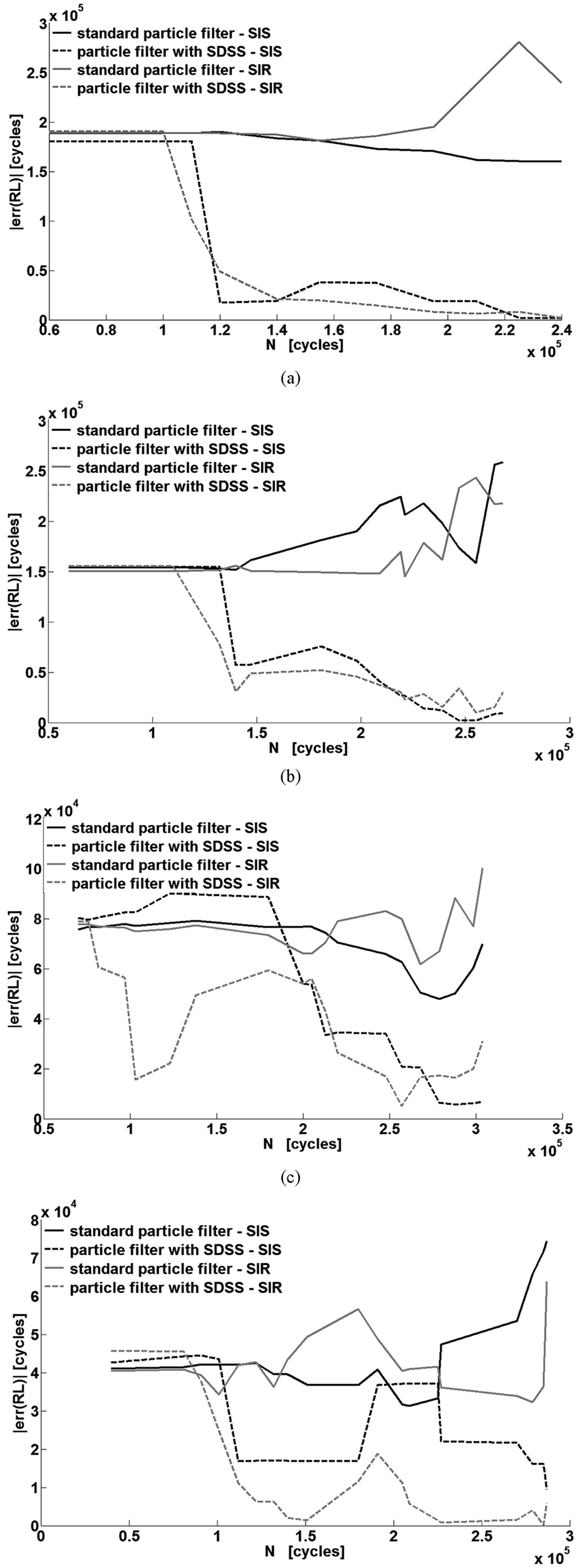
(c)

(d)

Fig. 19. RL predictions for the 4 tests using a SIR algorithm with deterministic DSS (standard particle filter formulation).

updates the parameter PDF during the resampling stage of the particle filter, in both the SIS algorithm (when the safe particles drops below a certain threshold), and in the SIR algorithm.

The validation of the methodology is made first using simulated crack propagation with and without artificial process noise,



(a)

(b)

(c)

(d)

Fig. 20. Absolute error on the Residual Lifetime for each experimental test (a)–(d).

highlighting the parameter identification. The methodology is then applied to real crack propagations on helicopter fuselage panels, estimating the RL of the structure. The comparison of the results obtained with the performances of a standard particle filtering algorithm (both on simulated cracks and on real crack propagation data) proves the effectiveness of the method. Moreover, the test on several portions of a helicopter fuselage is a significant novelty in the field of real-time prognostics via Bayesian filtering methods. Although the system produces good results, the computational effort is still an open issue, especially in real-time applications. The two main causes for the high time-consumption are the MH algorithm required to update the parameter PDF and the numerical simulation of FCG based on the NASGRO law for RL updating. As a matter of fact, there is no closed form solution for the NASGRO integration. Thus, crack propagation must be simulated according to the linear damage accumulation model in (14), or through numerical methods, to calculate the integral of the NASGRO law. Moreover, the employment of special numerical methods for the SIF formulation also increases the time-consumption. As a matter of fact, high computational costs are associated in this work also to the SIF calculation, here performed by an ANN regressor. Another problem that has to be solved is the number of measures required by the algorithm to produce good results. The paper does not face this problem, and it should be a matter of future research. Of course, the proposed algorithm is suitable for advanced systems composed by diagnostic units able to provide continuous information about the health state of the structure, so that the number of measures would be sufficiently large to guarantee the algorithm convergence. If the number of observations becomes limited, the time steps required by the algorithm would become another fundamental variable for the evaluation of the performances. However, the results related to the experimental data come from a relatively small number of crack measures, manually collected during the tests. The numbers of measures are 11, 16, 19, and 19 for the four experiments presented here. According to the average bias B presented in Table VII, the higher the number of measures, the better the overall performance of the algorithm will be. As declared above, this intuitive outcome should be investigated further. On the other hand, the presented work highlights the importance of the model parameters to provide reliable RL predictions. Several papers related to the topic proved the suitability of particle filter for prognostic problems. However, if the model parameters used in prediction algorithms are wrong, then the prognosis on the system evolution will also be wrong. Because of the uncertainties affecting real structures, the employment of standard Sequential Monte Carlo sampling requires the adjustment of the model parameters and

TABLE VII
AVERAGE BIAS B – EXPERIMENTAL DATA

	Algorithm	SIS	SIR
Exp. 1	Standard particle filter	1.62E+05	1.9E+05
	Particle filter & MH	0.39E+05	0.38E+05
Exp. 2	Standard particle filter	1.8E+05	1.6E+05
	Particle filter & MH	0.06E+05	0.05E+05
Exp. 3	Standard particle filter	6.6E+04	7.3E+04
	Particle filter & MH	-0.57E+04	0.35E+04
Exp. 4	Standard particle filter	4.3E+04	4.0E+04
	Particle filter & MH	-0.8E+04	0.65E+04
Global Average B	Standard particle filter	1.13E+05	1.16E+05
	Particle filter & MH	7.78E+03	1.31E+04

the process noise on data that are not available yet. The problem appears even if the process noise artificially inserted in the algorithm is able to cover the variability between two subsequent time-steps of particle filtering (i.e., two subsequent measures). In fact, the prognosis for the very far future is mainly driven by the model parameters (governing the estimated system evolution) rather than by the artificial process noise adopted in the SIS and SIR formulation. This paper furthermore underlines the capability of the adaptive algorithms to cover the inter-specimen variability, to make the real-time prognostics feasible and reliable. Several suggestions for further developments towards the prognosis of real structures arise from this work. First of all, Bayesian filtering techniques accounting for the uncertainty of model parameters should be used in these types of problems. As an example, particle filters with an augmented state vector or with particular modifications to estimate the model parameters have been studied in [37], [38], and other papers. Moreover, an in-depth investigation of Bayesian filtering capabilities to deal with several time-varying variables should be assessed. As a matter of fact, aeronautical structures (like the one presented in this work), civil structures, and several mechanical structures are subjected to variable amplitude and random load histories. Thus, the Dynamic State-Space model of the damage evolution becomes a function of the future loading conditions, which are in most cases unknown. Another important aspect to be considered is the possibility to apply the methods described in this work to a case of real-time damage diagnosis, such as the one presented by the authors in [1], thus allowing us to filter the uncertainties inevitably present in a real-time diagnostic framework.

References

[1] C. Sbarufatti, A. Manes, and M. Giglio, “Performance optimization of a diagnostic system based upon a simulated strain field for fatigue damage characterization,” *Mech. Syst. Signal Process.*, vol. 40, no. 2, pp. 667–690, Nov. 2013.

[2] V. M. Karbhari and F. Ansari, *Structural Health Monitoring of Civil Infrastructure Systems*. Cambridge, U.K.: WP Limited, 2009.

[3] G. V. Arkadov, A. F. Getman, and A. N. Rodionov, *Probabilistic Safety Assessment for Optimum Nuclear Power Plant Life Management*. Cambridge, U.K.: WP Limited, 2012.

[4] M. Corbetta, C. Sbarufatti, A. Manes, and M. Giglio, “On dynamic state-space models for fatigue-induced structural degradation,” *Int. J. Fatigue*, vol. 61, pp. 202–219, Apr. 2014.

[5] J. N. Yang and S. D. Manning, “A simple second order approximation for stochastic crack growth analysis,” *Eng. Fracture Mech.*, vol. 53, no. 5, pp. 677–686, Mar. 1996.

[6] R. Patankar and A. Ray, “State space modelling of fatigue crack growth in ductile alloys,” *Eng. Fracture Mech.*, vol. 66, pp. 129–151, May 2000.

[7] J. Luo and P. Bowen, “A probabilistic methodology for fatigue life prediction,” *Acta Mater.*, vol. 51, no. 12, pp. 3537–3550, Jul. 2003.

[8] P. C. Paris and F. Erdogan, “A critical analysis of crack propagation laws,” *Trans ASME, J. Basic Eng.*, vol. 85, no. 4, pp. 528–34, Dec. 1963.

[9] NASGRO Reference Manual, Version 4.02, NASA JS. Centre and Southwest Research Institute, 2002.

[10] M. E. Orchard and G. J. Vachtsevanos, “A particle filtering approach for on-line failure prognosis in a planetary carrier plate,” *Int. J. Fuzzy Logic Intell. Syst.*, vol. 7, no. 4, pp. 221–227, Dec. 2007.

[11] F. Cadini, E. Zio, and D. Avram, “Monte Carlo-based filtering for fatigue crack growth estimation,” *Probab. Eng.. Mech.*, vol. 24, no. 3, pp. 367–373, Jul. 2009.

[12] J. Chiachio, M. Chiachio, A. Saxena, G. Rus, and K. Goebel, “An energy-based prognostic framework to predict fatigue damage evolution in composites,” in *Proc. Annu. Conf. Prognost. Health Manag. Soc.*, Oct. 2013.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[14] N. Kantas, A. Doucet, S. S. Singh, and J. M. Maciejowski, “An overview of sequential Monte Carlo methods for parameter estimation in general state-space models,” in *Proc. 15th IFAC Symp. System Identification*, 2009.

[15] M. Corbetta, C. Sbarufatti, A. Manes, and M. Giglio, “Stochastic definition of state-space equation for particle filtering algorithms,” *Chem. Eng. Trans.*, vol. 33, pp. 1075–1080, 2013.

[16] M. Corbetta, C. Sbarufatti, A. Manes, and M. Giglio, “On-line updating of dynamic state-space model for Bayesian filtering through Markov chain Monte Carlo techniques,” *Chem. Eng. Trans.*, vol. 33, pp. 133–138, 2013.

[17] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York, NY, USA: Springer Science Business Media, 2004, Springer Text in Statistics.

[18] S. P. Brooks, “Markov chain Monte Carlo method and its application,” *Statistician*, vol. 47, no. 1, pp. 69–100, 1998.

[19] S. Chib and E. Greenberg, “Understanding Metropolis-Hastings algorithm,” *Amer. Statistician*, vol. 49, no. 4, pp. 327–335, Nov. 1995.

[20] M. Corbetta, C. Sbarufatti, A. Manes, and M. Giglio, “Fatigue Crack Growth under random spectrum loading: Markov chain Monte Carlo methods for parameter identification,” in *Proc. 22nd Eur. Safety RELiability Conf.*, 2013.

[21] G. O. Roberts and J. S. Rosenthal, “Optimial scaling of discrete approximations to Langevin diffusions,” *J. R. Statist. Soc. B*, vol. 60, no. 1, pp. 255–268, 1998.

[22] G. O. Roberts and J. S. Rosenthal, “Optimal scaling for various Metropolis-Hastings algorithms,” *Statist. Sci.*, vol. 16, no. 4, pp. 351–367, 2001.

[23] W. R. Gilks, G. O. Roberts, and E. I. George, “Adaptive direction sampling,” *Statistician*, vol. 43, no. 1, pp. 179–189, 1994.

[24] W. R. Gilkts, G. O. Roberts, and S. K. Sahu, “Adaptive Markov Chain Monte Carlo through regeneration,” *J. Amer. Statist. Assoc.*, vol. 93, no. 443, pp. 1045–1054, Sep. 1998.

[25] H. Haario, E. Saksman, and J. Tamminen, “Adaptive proposal distribution for random walk Metropolis algorithm,” *Computat. Statist.*, vol. 14, no. 3, pp. 375–395, Sep. 1999.

[26] H. Haario, E. Saksman, and J. Tamminen, “An adaptive Metropolis Algorithm,” *Bernoulli*, vol. 7, no. 2, pp. 223–242, Apr. 2001.

[27] K. Worden and J. J. Hensman, “Parameter estimation and model selection for a class of hysteretic systems using Bayesian inference,” *Mech. Syst. Signal Process.*, vol. 32, pp. 153–169, Apr. 2012.

[28] A. Gelman, G. O. Roberts, and W. R. Gilks, “Efficient Metropolis jumpings rules,” in *Proc. 5th Valencia Int. Meeting*, 1996, pp. 599–607.

[29] D. A. Virkler, B. M. Hillberry, and P. K. Goel, The Statistical Nature of Fatigue Crack Propagation, 1978, Tech. rep. AFFDL-TR-78-43.

[30] J. M. Bourinet and M. Lemaire, “Form sensitivities to correlation: Application to fatigue crack propagation based on Virkler data,” in *Proc. 4th Int. ASRANet Colloq.*, 2008.

[31] N. E. Dowling, *Mechanical Behaviour of Materials*. Englewood Cliffs, NJ, USA: Prentice Hall, 2002.

[32] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, “Metrics for evaluating performance of prognostic techniques,” in *Proc. Int. Conf. Prognostics and Health Management*, 2008.

[33] C. Sbarufatti, A. Manes, and M. Giglio, “Application of sensor technologies for local and distributed structural health monitoring,” *Struct. Control Health Monit.*, vol. 21, no. 7, pp. 1057–1083, Jul. 2014.

[34] C. M. Bishop, *Neural Network for Pattern Recognition*. Oxford, U.K.: Oxford Univ. Press, 1995.

[35] I. T. Nabney, *NETLAB Algorithms for Pattern Recognition*. London, U.K.: Springer-Verlag, 2004.

[36] C. Sbarufatti, G. Manson, and K. Worden, “A Numerically enhanced machine learning approach to damage diagnosis using a Lamb Wave Sensing Network,” *J. Sound Vibrat.*, vol. 333, no. 19, pp. 4499–4525, Sep. 2014.

[37] G. Storvik, “Particle filters for state-space models with the presence of unknown static parameters,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 281–289, Feb. 2002.

[38] E. Chatzi and A. W. Smyth, “Particle filter scheme with mutation for the estimation of time-invariant parameters in structural health monitoring applications,” *Struct. Control Health Monit.*, vol. 20, pp. 1081–1095, 2013.