The scenario approach for data-driven prognostics

D. Cesani * **M. Mazzoleni** * **F.Previdi** *

* *Department of Management, Information and Production engineering University of Bergamo, via Marconi 5, 24044 Dalmine (BG), Italy.*

Abstract: Prognostics is the process of forecasting the time-to-failure or the time-to-alarm of an industrial item using degradation models. Data-driven approaches to prognostics employ regression models fit on condition indicators computed from raw run-to-failure data to extrapolate the degradation behaviour of the item. The development of a reliable data-driven degradation model typically requires many run-to-failure acquisitions to understand the degrading behavior. Such experimental tests are destructive and expensive for items manufacturers. Thus, decreasing the number of run-to-failure experiments is key in reducing predictive maintenance costs. In this work, focusing on time-to-alarm prediction to anticipate items breakdown, we propose a data-driven method based on the scenario approach to characterise the degradation behaviour of an industrial item in certain operative conditions using only one run-to-failure experiment, updating the time-to-alarm prediction only when needed. The scenario approach gives probabilistic guarantees on the time-to-alarm predictions.

Keywords: Prognostics; Scenario approach.

1. INTRODUCTION

Predictive maintenance is a tool to maximize industrial machinery dependability, reduce damages caused by failed items and decrease intervention costs (Zio, 2022). In this context, Prognostics and Health Management (PHM) is a research field that links the study of failure mechanisms and life cycle management of mechanical items of industrial machinery (Javed et al., 2017). PHM approaches employ raw sensor data from industrial machines to compute condition indicators (CIs), preferably monotonic and proportional with the degradation level, used to continuously evaluate the health state of the items (*condition monitoring*) (Mazzoleni et al., 2021). Data-driven methods to *prognostics* fit a degradation model on the computed CIs using regression techniques. These models can be used at the current time to predict the CIs values at future time instants. The model and its predictions are then updated each time a new CI value is available (Javed et al., 2017; Maurelli et al., 2024), see (Zhai et al., 2019) and (Ramezani et al., 2023) for thorough reviews. Often, the uncertainty in the CIs predicted values is considered to provide an interval of possible future CI values.

Relying on the design specifications of the items, *health state thresholds* on CIs trend are derived to estimate the items Remaining Useful Life (RUL). Typically, prognostics methods work with a set of different thresholds, denoting an (i) initial degradation, (ii) an alarm condition where the degradation is critical, and (iii) a complete failure situation (Javed et al., 2015). RUL estimation is however a complex task since the prediction of the CIs future trend is affected by different sources on uncertainty (Javed et al., 2017)(Celaya et al., 2012)(Sankararaman, 2015). For this reason, many run-to-failure data are typically required to develop a reliable degradation model that is able to also quantify the uncertainty in the predictions. Still, run-to-failure experiments are costly and difficult to implement in industry (Mazzoleni et al., 2022).

Following the previous statements, the development of a data-driven degradation model can be thought as an optimization problem under uncertainty. In this context, the *scenario approach* theory allows to learn models from data providing finite-sample uncertainty quantification that are independent from the data probability distribution (Campi et al., 2021). In particular, Interval Predictor Methods (IPMs) apply the scenario theory on estimating a prediction interval, rather than a point value (Campi et al., 2009)(Garatti et al., 2019).

In this paper, we propose a data-driven prognostics method that relies on the scenario approach and employs an IPM for fitting a degradation model for a single CI. The use of the IPM allows to (i) consider *only a single* run-to-failure experiment for data acquisition, regardless of the fault on the item, so that only a single item needs to be characterized with respect to its degradation history; (ii) manage the uncertainty in the measured data, providing a forecast of the *time-to-alarm*[1] *interval* with *finite-sample and distribution-free* probabilistic guarantees; (iii) update the degradation model *only when needed*, in case the online behavior of a new item is different from the characterised one. The proposed approach to prognostics is evaluated on the public PRONOSTIA fempto-bearings dataset (Nectoux et al., 2012). Results are compared with the Summation-Wavelet Extreme Learning Machine (`sw-elm`) structure presented in (Javed et al., 2015).

The remainder of the paper is organised as follows. Section 2 presents the main results of the scenario approach theory.

[1] The time-to-alarm is defined as the time instant when the CI forecast intersects the alarm threshold denoting an alarm condition.

Section 3 and Section 4 define the proposed prognostics method and show validation and comparison results, respectively. Section 5 is devoted to concluding remarks.

2. SCENARIO APPROACH THEORY

2.1 Review of the scenario approach

Consider the optimization problem as defined in (Campi and Garatti, 2008, 2011; Garatti and Campi, 2013)

$$\min_{\boldsymbol{v}\in\mathbb{R}^{d-1}} \ell(\boldsymbol{v},\delta), \tag{1}$$

where $\boldsymbol{v} \in \mathbb{R}^{d-1}$ is the vector of design variables and $\delta \in \Delta$ is a generic *uncertain element*, with Δ the set of all possible uncertainty elements. The cost $\ell(\boldsymbol{v},\delta)$ is assumed convex in $\boldsymbol{v}$ for any given δ with no dependence on Δ. However, problem (1) is not well-defined as it lacks a description of how δ should be accounted for in the optimization process. To address this issue, the scenario approach considers N outcomes of the uncertainty variable $\delta_1, \delta_2, ..., \delta_N$, called *scenarios*, sampled independently from Δ according to the probability measure $\mathbb{P}$. In practical applications, a finite number of N scenarios are used to approximate Δ and (approximately) solve (1), following in its simplest formulation a worst-case philosophy. This leads to the following *scenario program*.

Definition 1. (Scenario Program).

$$\mathrm{SP}_N: \quad \min_{\boldsymbol{v}\in\mathbb{R}^{d-1}} \left[\max_{i=1,\dots,N} \ell(\boldsymbol{v},\delta_i)\right], \tag{2}$$

Problem (2) is solved in place of (1) to obtain a solution $\boldsymbol{v}^*$ with ℓ^* the related optimal cost value.

Consider now a new scenario $\delta_{N+1} \neq \delta_i, i = 1,\dots,N$. As δ_{N+1} has not be considered in solving (2), it may happen that $\ell(\boldsymbol{v}^*,\delta_{N+1}) > \ell^*$ (that is, the solution $\boldsymbol{v}^*$ is not the optimal one) with a certain probability, quantified as follows.

Theorem 1. (Campi and Garatti, 2008) For any $\epsilon \in (0,1)$ (*risk parameter*), and $\beta \in (0,1)$ (*confidence parameter*), if the number of scenarios N satisfies $N \geq \frac{2}{\epsilon}\left(\ln\frac{1}{\beta} + d - 1\right)$, then, with probability $\geq 1-\beta$, it holds that ℓ^* is ϵ-*risk guaranteed*, that is

$$\mathbb{P}\left\{\delta \in \Delta : \ell(\boldsymbol{v}^*,\delta) > \ell^*\right\} \leq \epsilon.$$

The risk parameter ϵ is defined by the user and bounds the quantity $\mathbb{P}\left\{\delta \in \Delta : \ell(\boldsymbol{v}^*,\delta) > \ell^*\right\}$, that must be interpreted as the probability of observing δ such that $\ell(\boldsymbol{v}^*,\delta) > \ell^*$ for fixed values of $\boldsymbol{v}^*$ and ℓ^*. The quantity

$$V(\boldsymbol{v}^*,\ell^*) \coloneqq \mathbb{P}\left\{\delta \in \Delta : \ell(\boldsymbol{v}^*,\delta) > \ell^*\right\} \tag{3}$$

is called the *violation probability* and quantifies the robustness level of $(\boldsymbol{v}^*,\ell^*)$. Furthermore, $\left\{\delta \in \Delta : \ell(\boldsymbol{v}^*,\delta) > \ell^*\right\}$ is called *violation set*, of which $V(\boldsymbol{v}^*,\ell^*)$ is the probabilistic measure. Lastly, note that N is independent of $\mathbb{P}$, so no knowledge of $\mathbb{P}$ is required.

The practical application of the scenario approach only requires to have at disposal enough scenarios, no matter how obtained. For instance, a scenario may represent a set of scalar observations $\{u_i, y_i\}_{i=1}^N$ that are related by a generic relationship $g(\cdot)$ so that

$$y_i = g(u_i) + \text{noise}. \tag{4}$$

This application concerns the interval predictor method.

2.2 Review of the interval predictor method

Interval predictors are regression models that return a *prediction interval* (also known as a *layer*), as opposed to just a single point prediction (Campi et al., 2009; Garatti et al., 2019). Consider a set of N independent observations (scenarios) from (4)

$$\delta_i \coloneqq (u_i, y_i), \quad i = 1,\dots,N. \tag{5}$$

Interval predictor models are composed by:

(1) a standard regression model $\hat{g}(\cdot)$ for $g(\cdot)$ in (4) estimated using (5);
(2) a point prediction $\hat{y}(u_{N+1}) = \hat{g}(u_{N+1})$ for a new unknown input u_{N+1} and a prediction interval $[\hat{y}_{\text{low}}(u_{N+1}), \hat{y}_{\text{up}}(u_{N+1})]$ where the true value $y(u_{N+1}) = g(u_{N+1})$ lies within a certain guaranteed probability.

Let $\hat{g}(\cdot)$ be a polynomial model of degree $n = d-1$. The IPM is estimated by solving the convex problem

$$\min_{\boldsymbol{v}\in\mathbb{R}^n} \left[\max_{i=1,\dots,N} \left|y_i - \left[v_1 + v_2 u_i + ... + v_n u_i^{n-1}\right]\right|\right], \tag{6}$$

where $\boldsymbol{v} = \begin{bmatrix} v_1 & \dots & v_n \end{bmatrix}^\top \in \mathbb{R}^n$ are the polynomial coefficients to be estimated and $\boldsymbol{v}^* \in \mathbb{R}^n$ is the problem solution. Note that (6) follows from (2) by imposing

$$\ell(\boldsymbol{v},\delta_i) = \left|y_i - \left[v_1 + v_2 u_i + ... + v_n u_i^{n-1}\right]\right|,$$

with the uncertain elements δ_i defined by (5).

Introducing an auxiliary variable $\ell \in \mathbb{R}$, problem (6) can be reformulated as a problem in dimension $d = n+1$ (Carè et al., 2015):

$$\begin{aligned} &\min_{\ell,\boldsymbol{v}\in\mathbb{R}^n} \ \ell \\ &\text{s.t.}: \left|y_i - \left[v_1 + v_2 u_i + ... + v_n u_i^{n-1}\right]\right| \leq \ell. \end{aligned} \tag{7}$$

where $\{\ell^* \in \mathbb{R}, \boldsymbol{v}^* \in \mathbb{R}^n\}$ is the solution of (7). Using $\boldsymbol{v}^* = \begin{bmatrix} v_1^* & \dots & v_n^* \end{bmatrix}^\top \in \mathbb{R}^n$, the point prediction $\hat{y}(u_i)$ at input u_i is

$$\hat{y}(u_i) = v_1^* + v_2^* u_i + ... + v_n^* u_i^{n-1}, \tag{8}$$

while the prediction interval in u_i is defined by the limits

$$\hat{y}_{\text{low}}(u_i) = \hat{y}(u_i) - \ell^*, \tag{9a}$$
$$\hat{y}_{\text{up}}(u_i) = \hat{y}(u_i) + \ell^*. \tag{9b}$$

The term *layer* refers to the region of the u-y plane composed by values in (9) evaluated for any input u.

Given a new observation $\delta_{N+1} = (u_{N+1}, y_{N+1})$, the probability that δ_{N+1} lies outside (9) is the *violation* of the solution $(\boldsymbol{v}^*,\ell^*)$, defined as in (3)

$$V(\boldsymbol{v}^*,\ell^*) \coloneqq \mathbb{P}\left\{\left|y_i - \left[v_1^* + v_2^* u_i + ... + v_n^* u_i^{n-1}\right]\right| \geq \ell^*\right\}. \tag{10}$$

Theorem 1 can be specialised in this context as follows.

Theorem 2. (Campi et al., 2009) For any $\epsilon \in (0,1)$ (*risk parameter*), and $\beta \in (0,1)$ (*confidence parameter*), if the number of scenarios N in (4) satisfies $N \geq \frac{2}{\epsilon}\left(\ln\frac{1}{\beta} + n\right)$, then, with probability $\geq 1-\beta$ the layer fails to correctly predict (u_{N+1}, y_{N+1}) with probability no more than ϵ.

Theorem 2 is particularly powerful as it holds for every noise distribution and complexity of $g(\cdot)$ in (4). In many real situations where the model family $\hat{g}(\cdot)$ does not include $g(\cdot)$, Theorem 2 still holds, with a possibly wider layer (9).

3. DATA-DRIVEN PROGNOSTICS USING IPM

Consider two exemplars A and B of the same item operating under defined operative conditions. Let f_i, $i = 1, \dots, N$, be the i-th value of a (possibly) monotonic condition indicator (CI), computed from raw measurements acquired during a *single run-to-failure* experiment of item A, i.e. starting from healthy to a failed condition of the item. The observations (scenarios) follows from (4) as

$$\delta_i = (i, f_i), \quad i = 1, \dots, N \tag{11}$$

by setting $u_i = i$ and $y_i = f_i$, with i denoting the time index when f_i is computed. The time instants i do not need to be evenly spread, nor any knowledge about the fault mechanism is required. Then, an interval predictor model as (7) can be estimated using (11) to characterise the degradation process of item A as in (8)-(9)

$$\hat{f}(i) = v_1^* + v_2^* i + ... + v_n^* i^{n-1}, \tag{12a}$$

$$\hat{f}_{\text{low}}(i) = \hat{f}(i) - \ell^*, \tag{12b}$$

$$\hat{f}_{\text{up}}(i) = \hat{f}(i) + \ell^*, \tag{12c}$$

We denote this characterisation of the degradation model as the *primary* interval predictor.

Let T_{al} be a threshold that if exceeded denotes an *alarm situation*, we define as $i_{T_{\text{al}}}$ the time index where the predicted CI-values interval (12b)-(12c) first reaches the value T_{al}, that is

$$\hat{f}_{\text{low}}(i_{T_{\text{al}}}) = T_{\text{al}}, \tag{13a}$$

$$\hat{f}_{\text{up}}(i_{T_{\text{al}}} - \rho) = T_{\text{al}}, \tag{13b}$$

with ρ the width (in time) of the estimated time-to-alarm interval. We use ρ to evaluate the performance of an estimated interval. In this context, a better performance means a lower ρ. Note that, if the model structure (12) is wrong (e.g. the data are generated by a polynomial model of higher grade or by a model of another family) the primary interval predictor will be wider (higher ρ) with respect to the one obtained using the right model structure. In this case, despite possible minor performance (i.e. a higher ρ) , all data points will still be inside the interval (12b) (12c) so the primary interval predictor is still valid. To improve the layer performance (i.e. to obtain a lower ρ) while maintaining a polynomial model, it might be necessary to increase the order of the polynomial model. Finally, we also define as $i^{\text{est}}_{T_{\text{al}}}$ the time index where the estimated polynomial model (12a) touches T_{al}, that is

$$\hat{f}(i^{\text{est}}_{T_{\text{al}}}) = T_{\text{al}} \tag{14}$$

Assume now to employ (12) in production, that is to perform prognostics using online data from another item B, which is a different exemplar but shares the same design of A (e.g. the two items can be different physical instantiations of the same part number). Two situations may arise:

(1) the item B presents a *similar* CI behavior as item A (i.e. with CI values f_t^{B} that lie inside the layer defined by (12c)-(12b)), so that model (12) is consistent with the new data. The quantity f_t^{B} denotes the CI computed on raw data from item B at a time $t \neq i$;
(2) the item B shows a *different* CI behavior than item A (i.e. with CI values f_t^{B} that lie outside the layer defined by (12c)-(12b)), so that model (12) is not consistent with the new data.

In the first case, the quantity (13) is especially important as allows, for another item B with *similar* CI behavior as item A, to reuse the same prognostics model (12), according to the following proposition.

Proposition 1. For any $\epsilon \in (0, 1)$ and $\beta \in (0, 1)$, if the number of observations N in (11) satisfies $N \geq \frac{2}{\epsilon}\left(\ln \frac{1}{\beta} + n\right)$, then, with probability $\geq 1 - \beta$ the time-to-alarm interval (13) does not contain the true time-to-alarm with probability no more than ϵ.

Proof 1. Since (11) contains time-to-alarm data, the proof follows from the application of Theorem 2 to (12).

Consider now the case where item B shows a *different* CI behavior than item A (i.e. with CI values f_t^{B} outside the layer defined by (12c)-(12b). In this case, the interval predictor model (12) needs to be updated to characterize the new degradation behavior f_t^{B}. We denote the updated models as the *secondary* interval predictor.

Note however that the analysis of the degradation process of item B is made online when the system is running, so run-to-failure data are not available from B (indeed, it is of interest to predict its alarm time). Thus, in this case, no a priori guarantees can be given on the time-to-alarm prediction interval as in Proposition 1 until $N \geq \frac{2}{\epsilon}\left(\ln \frac{1}{\beta} + n\right)$ samples of f_t^{B} are acquired. At that time, by updating the prognostics model for item B as needed considering its incoming CI values, the validity of Proposition 1 will be restored. So, the secondary layer has the same properties of the primary one. Nonetheless, the presented approach allows to fully leverage a single run-to-failure experiment, providing finite-sample and distribution-free guarantees of the time-to-alarm prediction if the online-computed CI values are compatible with the primary model (12).

Consider again the situation where item B shows CI values f_t^{B} that are not consistent with model (12) estimated using (11), so that (12) needs to be updated according to the incoming CI values f_t^{B}. As reported in Theorem 2, the scenario approach requires a minimum quantity of observations N in (5), that depend from the user choice of parameters ϵ, β and n, prior to employ the model with its probabilistic guarantees. The rationale for the updates of the interval predictor model is as follows. First, define the following *deployment thresholds* T_1, T_2, Q, in addition to the typical thresholds denoting (i) initial degradation T_{deg}, (ii) alarm T_{al}, and (iii) complete failure T_{fail}:

- T_1 is the degradation level where the comparison between the degradation behavior of A and B starts. It can be set as $T_1 = T_{\text{deg}}$ or $T_1 < T_{\text{deg}}$;
- T_2 is the degradation level where any new update of the interval predictor model (12) is performed, only when needed. To update the model in advance with respect to a supposedly true alarm, it is set as $T_2 < T_{\text{al}}$. When feature values from B exceed T_2, the model can be updated according to the threshold Q;
- Q is the number of CI samples that are allowed to lie outside the primary layer (12), before triggering the updated of the interval predictor.

The exact definition of the degradation/alarm/failure thresholds is application-dependent and mostly based on the physics of the degradation process. The proposed

deployment thresholds T_1, T_2, Q are less critical to tune as they may be defined as a percentage of the standard thresholds $T_{\text{deg}}, T_{\text{al}}, T_{\text{fail}}$ or (especially T_2) be set automatically when exactly a total of N feature samples from `B` are acquired. Algorithm 1 and Algorithm 2 summarize the method.

Algorithm 1: Primary interval predictor.

Input: ϵ, β, n
Output: primary interval predictor model (12)
Acquire raw run-to-failure data from item `A`
Compute N values of the CI as in (11)
Solve (7) using (11) to get (12)

Algorithm 2: Use and update of the primary model.

Input: primary model (12), a new CI value f_t^{B}, $T_1, T_2, Q, T_{\text{deg}}, T_{\text{al}}, T_{\text{fail}}$
Output: secondary interval predictor models
$q = 0$
if $f_t^B \geq T_1$ **and** $f_t^B \leq T_2$ **then**
  if f_t^B **is not** *in the primary layer* (12) **then**
    $q = q + 1$
  end
else
  if $q > Q$ **then**
    update the interval predictor model using the f_t^{B} CIs collected up to the current time t
    $q = 0$
  end
end

4. CASE STUDY: PRONOSTIA FEMPTO-BEARINGS

The PRONOSTIA experimental platform (Nectoux et al., 2012) is used to validate the proposed approach. We consider bearings data under rotation speed of 1800 rpm and a radial force of 4000 N, using only the horizontal accelerometer measurements. We use the `bearing1-1` (`b1-1`) experiment for offline estimation of the primary model (12), and the `bearing1-3` (`b1-3`), `bearing1-4` (`b1-4`) experiments to show how the fine tuning works and when it is performed, simulating an online prognostics. The performance of the proposed method will be assessed by the width ρ of the estimated time-to-alarm interval. The results considering the other available acquisitions are not reported to simplify the exposition. However, the method performs similarly also in these cases.

Figure 1-(left) shows the PRONOSTIA raw data. Note that the `b1-1` experiment has the longest run-to-failure acquisition while `b1-3` and `b1-4` reach a complete failure earlier. On `b1-1` we define the time instant where we consider that the bearing started to degrade (green dashed line), the time instant where the component is critically degraded (red dashed line), and the time instant where it is broken (black dashed line). It is clearly visible that `b1-4` reaches the complete failure status much earlier than `b1-1` (used to estimate the primary model), so in this case the degradation process is different and we expect that an updating of the primary model is needed.

As in (Javed et al., 2015), the monotonic *cumulative feature* $\tilde{\sigma}(\text{arctan})_i$ is defined as the CI f_i as follows: first, the arctan value is computed for each sample of the raw vibration signal; then, the data are organised in

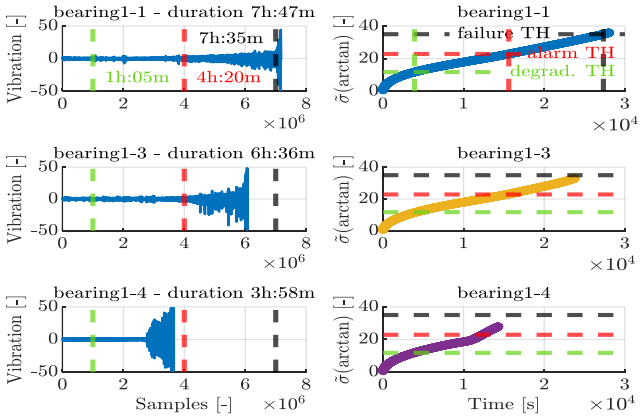

Fig. 1. (Left) raw vibration signals. The green dashed line is the time instant at which we consider that the bearing starts to degrade. The red one is the time instant at which we consider that it is critically degraded. The black one is the time instant at which the component breaks down. (Right): Condition indicator $\tilde{\sigma}(\text{arctan})_i$. Each point covers 10 s of raw data. The horizontal green dashed line represents the degradation threshold T_{deg}; the red one is the alarm threshold T_{al} and the black one is the complete failure threshold T_{fail}.

N batches that cover 10 s of processed samples, with P data in each batch. For batch $i = 1, \ldots, N$, the standard deviation of the data is computed and a smoothing process is performed by applying a local regression (LOESS) filter with a span value of 0.3, obtaining the *simple feature* $\sigma(\text{arctan})_{i,j}$ for the j-th sample in the i-th batch. The cumulative feature is then computed as

$$\tilde{\sigma}(\text{arctan})_i = \frac{\sum_{j=1}^{M} \sigma(\text{arctan})_{i,j}}{\left|\sum_{j=1}^{M} \sigma(\text{arctan})_{i,j}\right|^{\frac{1}{2}}}, \quad i = 1, \ldots, N \quad (15)$$

This definition allows to use (15) in online situations, like the considered case. Figure 1-(right) shows the computed CIs (15). In the first graph above we derive the degradation/alarm/complete failure thresholds considering the CI values corresponding to the time instants defined on raw vibration data. For `b1-3` and `b1-4` we simulate an online acquisition process, using Algorithm 2.

4.1 Primary interval predictor estimation

The proposed method performs offline only the primary interval predictor estimation, using all the CI samples together calculated from the run-to-failure raw data of `b1-1`. We fix $\epsilon = 0.05$, $\beta = 1 \cdot 10^{-9}$ and $n = 6$. As a result, the minimum quantity of samples needed is $N = 1069$. The deployment thresholds are set as $T_1 = 0.9 \cdot T_{\text{deg}}$ and $T_2 = 0.9 \cdot T_{\text{al}}$. Lastly, we set $Q = 200$ which allows new features to lie outside the primary layer for 2000 seconds consecutively. Cyan lines in Figure 2 show how the estimated layer contains the true value of (15) (blue line). The width of this primary time-to-alarm interval is $\rho = 21$ minutes. This performance can be considered good in this context, since on more than 6-hours test, it does not lead to overly premature alarms.

4.2 Online usage: model updating not needed

Consider now the CI (15) from `b1-3`. The yellow line and the green zoomed box in Figure 2 show that the new

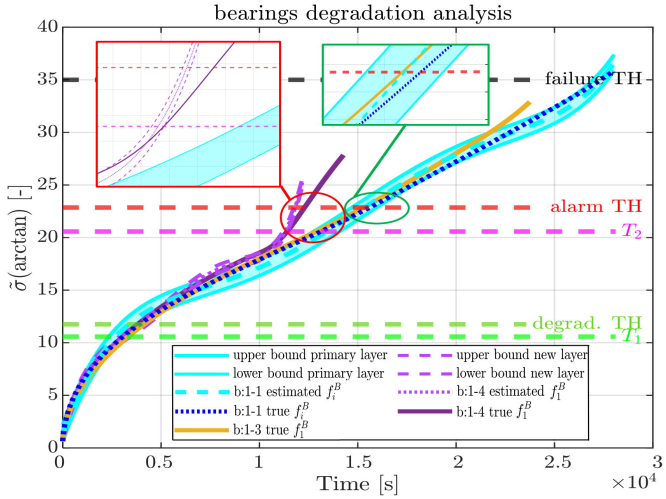

Fig. 2. (Cyan area): primary interval predictor estimation. (Yellow line): `b1-3` degradation process analysis. (Purple line) `b1-4` degradation process analysis.

considered item has a degradation process that stays in the primary layer (12c)-(12b). It has some points outside prior to threshold T_2, but the counting does not exceed Q.

In this case, the proposed method does not solve any other optimization problem, saving computational resources that can be used for other tasks. Human operators can check runtime how the degradation evolves having already the primary time-to-alarm interval. So, the preparation of the maintenance can be scheduled very early. If, as in the next case, the degradation process is outside the primary layer for more than Q points before the T_2 level, a pre-alarm can be generated to alert the operators.

4.3 Online usage: model updating needed

Lastly, we analyse the behavior of `b1-4`. The purple line in Figure 2 shows that the new item has a degradation process not consistent with the primary layer. So, the degradation model must be updated. The new interval predictor is computed only when the feature (15) reaches the threshold T_2, so reducing the computational load. In the red zoomed part of Figure 2, it is possible to observe that the projection of the interval estimation (purple dotted and dashed lines) from T_2 to the alarm threshold T_{al} no longer contains the true value of (15) (purple solid line). This is acceptable for two reasons: first, the estimated interval is conservative with respect to the real alarm instant; second, data that occur after T_2 have not yet been acquired. Here comes the possibility to update the model each time a new CI value, or a batch of new values, is acquired.

Figure 3 shows this iterative update. The layer was set to be updated whenever a batch of 50 new CI values is available. It can be seen that the updated layer (in yellow) touches T_{al} closer to the true alarm instant. The time difference between the two instants is about 5 minutes. This small estimation error is not critical in the prognostics task as the alarm is not too premature. Note that this error can be imputed to the choice of a polynomial interval predictor. Selecting different model families may result in different predictions. Nevertheless, the prognostics task was completed. The model performance is still very good,

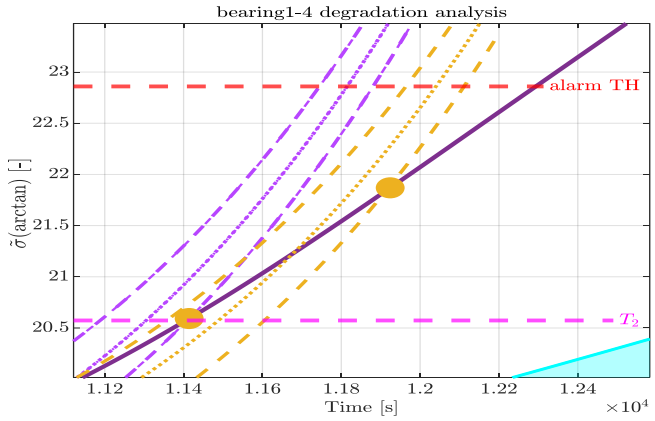

Fig. 3. Updating the new layer (purple dashed line) with a batch of 50 new feature samples, that lie between the 2 yellow dots on the purple solid line. In yellow, the updated layer.

as the width ρ of the estimated fine tuned interval is about 2 minutes and 30 seconds. In this case, a key role is played by T_1 and specially by T_2 thresholds. The former can be fixed to a lower level to make the comparison earlier and possibly to allow arising a pre-alarm if the degradation is different to the primary one. The latter determines when the new degradation model estimation is performed. A lower T_2 leads to an earlier model estimation, however it is possible that the time difference between the predicted alarm time and the true one increases, as enough observations N should be collected for the scenario guarantees to hold.

4.4 Comparison with `sw-elm` structure

The comparison between the proposed method and the `sw-elm` structure (Javed et al., 2015) consists of evaluating how much the RUL estimation $\widehat{RUL}(t_c)$ differs from the true one $RUL(t_c)$ for each *current time* t_c such that $0 < t_c < i_{T_{\text{al}}}^{\text{true}}$, where $i_{T_{\text{al}}}^{\text{true}}$ is the true time-to-alarm instant. In details, we define

$$RUL(t_c) = i_{T_{\text{al}}}^{\text{true}} - t_c; \qquad \widehat{RUL}(t_c) = i_{T_{\text{al}}}^{\text{est}} - t_c. \tag{16}$$

where $i_{T_{\text{al}}}^{\text{est}}$ is defined as (14). We consider the absolute value of the RUL estimation error, defined as follows:

$$|RUL_{\text{error}}(t_c)| = |RUL(t_c) - \widehat{RUL}(t_c)|. \tag{17}$$

Figure 4 shows how the methods perform, considering three different bearings: `b1-3`, `b1-4` and `bearing1-7` (`b1-7`). Figure 4 does not show the layers widths ρ. It is clearly visible that the first and the latter cases (blue and green lines) does not require IPM model updates; `b1-4` (red lines) instead implies two model updates. Considering the experiments `b1-3` and `b1-7` it is possible to say that the proposed method is beneficial with respect to the `sw-elm` algorithm: a constant error is more useful than one which continues to rise and fall in time, since it makes the maintenance schedule easier. Furthermore, especially in the `b1-3` case, $\widehat{RUL}(t_c)$ is only thirty seconds away from true $RUL(t_c), \forall t_c$. On experiment `b1-4` the proposed method has three different constant error levels, one relative to the primary layer and two for each model updates. These updates decrease the error significantly, as can be seen in the zoomed part of Figure 4. Instead, `sw-elm` performs worse at the beginning, but at some

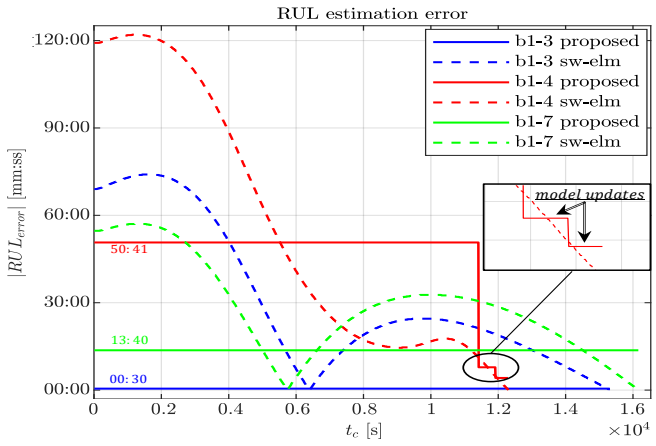

Fig. 4. Absulute value of RUL estimation errors. The solid lines are relative to the proposed method. The dashed ones instead, refer to the `sw-elm` method. `b1-4` requires IPM model updates to reduce the corresponding $|RUL_{\text{error}}(t_c)|$.

point its performance become better than the proposed method. However, when the IPM model is updated the two methods behaves similarly. Furthermore, `sw-elm` estimates the entire degradation process for each t_c, and thus it is computationally heavier with respect to the proposed IPM model that is updated only when needed.

5. CONCLUSIONS

This work presented a data-driven prognostics method based on the scenario approach theory. The strategy uses only one run-to-failure acquisition to characterise the degradation process of an item, leading to a primary time-to-alarm interval estimation. This interval has probabilistic guarantees of containing the true alarm instant. The obtained results evidence that the method performances are satisfactory, also in comparison with `sw-elm` algorithm. The strategy is thought to have a light computational load, updating the predictions only if the behavior of a new item is different from the the characterised one.

Future research will be devoted to provide an optimal strategy to define the deployment thresholds T_1 and T_2 as well as improving the comparison of degradation processes to detect differences more efficiently.

REFERENCES

Campi, M.C. and Garatti, S. (2008). The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3), 1211–1230. doi:10.1137/07069821X.

Campi, M. and Garatti, S. (2011). A sampling-and-discarding approach to chance-constrained optimization: Feasibility and optimality. *Journal of Optimization Theory and Applications*, 148, 257–280. doi:10.1007/s10957-010-9754-6.

Campi, M., Calafiore, G., and Garatti, S. (2009). Interval predictor models: Identification and reliability. *Automatica*, 45(2), 382–392. doi:10.1016/j.automatica.2008.09.004.

Campi, M., Carè, A., and Garatti, S. (2021). The scenario approach: A tool at the service of data-driven decision making. *Annual Reviews in Control*, 52, 1–17. doi:10.1016/j.arcontrol.2021.10.004.

Carè, A., Garatti, S., and Campi, M.C. (2015). Scenario min-max optimization and the risk of empirical costs. *SIAM Journal on Optimization*, 25(4), 2061–2080.

Celaya, J., Saxena, A., and Goebel, K. (2012). Uncertainty representation and interpretation in model-based prognostics algorithms based on kalman filter estimation.

Garatti, S., Campi, M., and CarÃ¨, A. (2019). On a class of interval predictor models with universal reliability. *Automatica*, 110, 108542. doi:10.1016/j.automatica.2019.108542.

Garatti, S. and Campi, M. (2013). Modulating robustness in control design: Principles and algorithms. *Control Systems, IEEE*, 33, 36–51. doi:10.1109/MCS.2012.2234964.

Javed, K., Gouriveau, R., and Zerhouni, N. (2017). State of the art and taxonomy of prognostics approaches, trends of prognostics applications and open issues towards maturity at different technology readiness levels. *Mechanical Systems and Signal Processing*, 94, 214–236. doi:10.1016/j.ymssp.2017.01.050.

Javed, K., Gouriveau, R., Zerhouni, N., and Nectoux, P. (2015). Enabling health monitoring approach based on vibration data for accurate prognostics. *IEEE Transactions on Industrial Electronics*, 62(1), 647–656. doi:10.1109/TIE.2014.2327917.

Maurelli, L., Mazzoleni, M., Previdi, F., and Camisani, A. (2024). A New Physics-Informed Condition Indicator for Cost-Effective Direct Current Solenoid Valves Using Significant Points of the Excitation Current. *Journal of Dynamic Systems, Measurement, and Control*, 146(3), 031007. doi:10.1115/1.4064602.

Mazzoleni, M., Rito, G.D., and Previdi, F. (2021). *Electro-Mechanical Actuators for the More Electric Aircraft*. Springer International Publishing. doi:10.1007/978-3-030-61799-8.

Mazzoleni, M., Sarda, K., Acernese, A., Russo, L., Manfredi, L., Glielmo, L., and Del Vecchio, C. (2022). A fuzzy logic-based approach for fault diagnosis and condition monitoring of industry 4.0 manufacturing processes. *Engineering Applications of Artificial Intelligence*, 115, 105317. doi:10.1016/j.engappai.2022.105317.

Nectoux, P., Gouriveau, R., Medjaher, K., Ramasso, E., Chebel-Morello, B., Zerhouni, N., and Varnier, C. (2012). Pronostia: An experimental platform for bearings accelerated degradation tests. 1–8. URL `https://www.nasa.gov/content/prognostics-center-of-excellence-data-set-repository`.

Ramezani, S.B., Cummins, L., Killen, B., Carley, R., Amirlatifi, A., Rahimi, S., Seale, M., and Bian, L. (2023). Scalability, explainability and performance of data-driven algorithms in predicting the remaining useful life: A comprehensive review. *IEEE Access*, 11, 41741–41769. doi:10.1109/ACCESS.2023.3267960.

Sankararaman, S. (2015). Significance, interpretation, and quantification of uncertainty in prognostics and remaining useful life prediction. *Mechanical Systems and Signal Processing*, 52-53, 228–247. doi:10.1016/j.ymssp.2014.05.029.

Zhai, X., Wei, X., and Yang, J. (2019). A comparative study on the data-driven based prognostic approaches for rul of rolling bearings. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1751–1755. doi:10.1109/SSCI44817.2019.9002764.

Zio, E. (2022). Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice. *Reliability Engineering & System Safety*, 218, 108119. doi:10.1016/j.ress.2021.108119.