Applied Energy 268 (2020) 114934

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Adaptive detection and prediction of performance degradation in off-shore turbomachinery

Marta Zagorowska[a,*], Frederik Schulze Spüntrup[b], Arne-Marius Ditlefsen[c], Lars Imsland[b], Erling Lunde[d], Nina F. Thornhill[a]

[a] Department of Chemical Engineering, Imperial College London, South Kensington, SW7 2AZ London, UK
[b] Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway
[c] ABB AS, Ole Deviks vei 10, 0666 Oslo, Norway
[d] Equinor ASA Research Center, Arkitekt Ebbells veg 10, 7053 Ranheim, Norway

HIGHLIGHTS

- Moving window is combined with regression analysis for prediction of degradation.
- The algorithm is tested on degradation data from two offshore applications.
- The predictions are accompanied by confidence intervals.
- The algorithm predicts the degradation accurately in the short and medium terms.
- The results are promising for improving performance-based maintenance.

ARTICLE INFO

Keywords:
Turbomachinery
Condition monitoring
Degradation
Degradation of performance
Regression

ABSTRACT

Performance-based maintenance of machinery relies on detection and prediction of performance degradation. Degradation indicators calculated from process measurements need to be approximated with degradation models that smooth the variations in the measurements and give predictions of future values of the indicator. Existing models for performance degradation assume that the performance monotonically decreases with time. In consequence, the models yield suboptimal performance in performance-based maintenance as they do not take into account that performance degradation can reverse itself. For instance, deposits on the blades of a turbomachine can be self-cleaning in some conditions. In this study, a data-driven algorithm is proposed that detects if the performance degradation indicator is increasing or decreasing and adapts the model accordingly. A moving window approach is combined with adaptive regression analysis of operating data to predict the expected value of the performance degradation indicator and to quantify the uncertainty of predictions. The algorithm is tested on industrial performance degradation data from two independent offshore applications, and compared with four other approaches. The parameters of the algorithm are discussed and recommendations on the optimal choices are made. The algorithm proved to be portable and the results are promising for improving performance-based maintenance.

1. Introduction

Performance-based maintenance of industrial machinery relies on an assessment of the current condition of the machinery and on prognosis of future loss of performance [1,2]. Loss of performance is commonly caused by degradation, defined as a 'detrimental change in physical condition, with time, use, or external cause' [3].

Hong et al. [4] indicate that degradation could be characterized by a degradation indicator that calculates the loss of performance as the machine operates. However, the calculation can be difficult because the mathematical formula is based on physics that is not always well understood, and also because the calculation makes use of operational

* Corresponding author.
E-mail addresses: m.zagorowska@imperial.ac.uk (M. Zagorowska), frederik@schulzespuentrup.de (F. Schulze Spüntrup), arne-marius.ditlefsen@no.abb.com (A.-M. Ditlefsen), lars.imsland@ntnu.no (L. Imsland), elunde@equinor.com (E. Lunde), n.thornhill@imperial.ac.uk (N.F. Thornhill).

https://doi.org/10.1016/j.apenergy.2020.114934
Received 22 October 2019; Received in revised form 25 January 2020; Accepted 28 March 2020
Available online 11 May 2020

measurements. A degradation indicator calculated from operational measurements can be highly variable if the calculation does not account for disturbances. Short-term variability caused by disturbances can mask the overall trend of the indicator.

Verheyleweghen and Jäschke [5] proposed a framework that shows how diagnostics and prognostics can help to optimize the operation of energy-relevant systems. The subsea gas compression system in their study can be assessed by analyzing vibration data to gain insights about the reliability of the system. *Reliability* is the 'ability of an item to perform a required function under given conditions for a given time interval' [3]. Their proposed framework leads to better decision-making and therefore an optimal strategy for profit maximisation considering the expected system degradation. In a similar way, Mahamad et al. [6] used a prediction of the remaining useful lifetime to improve reliability and reduce maintenance cost. Their approach could handle noisy data and was able to reduce operating costs by scheduling maintenance in a better way.

While the two aforementioned works aim to predict reliability, the work in this article considers degradation of efficiency in a compressor affected by fouling.

Predictive models are an important key to optimize washing schedules for gas turbine or compressors systems. Stalder [7] reviewed the state of the art of washing technology and highlighted the improvements for the turbine efficiency. Similarly, Aretakis et al. [8] perform an economic analysis of compressor washing. While these analyses compare different washing patterns, they assume very simple degradation patterns that are predictable. However, the data in this paper shows that degradation patterns are not always foreseeable and therefore more advanced prognostics are required. Therefore, a research gap can be closed, as these advanced algorithms will allow for optimized scheduling of compressor washing and restore efficiency, thus leading to energy savings. Schulze Spüntrup et al. [9] presented such an application, but they also assume a linear degradation pattern. This proves the possibility of scheduling maintenance actions based on specific degradation indicators. A more advanced model for the degradation indicator will enable results that are adaptive to the real-world changes in industrial turbomachinery.

Maintenance decisions will be based on the degradation indicator. Hence there is a need to estimate the true trend of the degradation indicator from the calculated degradation indicator, where *calculated degradation indicator* means the values calculated directly from process measurements. This can be done by fitting a low order regression model to the calculated values of the indicator, in order to provide a smoother estimate of the true value. The functional form of the regression model should be based on physical considerations, for instance it is known that degradation from fouling follows an exponential trend. A review of the state of the art in Section 2.2 will show that previous approaches have assumed that degradation is non-decreasing [10–14]. However, this is not true in general because some modes of operation can reduce degradation. For instance, compressor blades can be self-cleaning under some conditions. Hence there is a need for an improved structure of the regression model that can adapt to both increasing and decreasing behaviour of the degradation indicator.

The regression model can also be used to extrapolate the smoothed value of the degradation indicator. This will give a prediction of the future trend, which is useful for maintenance scheduling. Such predictions must be accompanied by confidence intervals if they are to be used for decision-making in an industrial setting [15].

This study proposes a data-driven algorithm for improved estimation and prediction of a degradation indicator for turbomachinery. It combines a moving window approach with adaptive regression analysis to predict the expected value of degradation and quantify the uncertainty of the prediction. The algorithm is tested on industrial degradation data from offshore compressors and is compared with four other approaches from the literature. The parameters of the algorithm such as the prediction window and the threshold for model adaptation are discussed, and the paper provides recommendations for optimal choices of these parameters.

Table 1
Abbreviations and names.

Abbreviation	Explanation
CT	Tuning data set from offshore compressor
C1, C2, C3	Test data sets from offshore compressor
P1, P2, P3, R	Selected time windows
E1, E2, E3, ES	Maintenance events
T1	Test data set from offshore turbine
ADP	Adaptive Degradation Prediction, the proposed algorithm
Mean	Mean value approximation
Linear	Linear regression over varying window
LinearC	Linear regression over fixed window
Fixed Exponential	Exponential approximation with fixed starting point

Table 2
Nomenclature.

Symbol	Explanation
d	Degradation indicator
$\hat{d}$	Estimated degradation indicator
Y	Measured performance variable, healthy
Y_{D}	Measured performance variable, degraded
η	Efficiency calculated from measurements, healthy
η_D	Efficiency calculated from measurements, degraded
η_0	Performance efficiency after maintenance event
$\tilde{\eta}$	Expected performance efficiency without degradation
t	Time
t_k^i	Time instant k in i-th window
b_0^i, b_1^i	Coefficients of linear regression in i-th approximation window
ε	Random error
$\hat{\sigma}$	Expected standard deviation
b_0^i, b_2^i, b_3^i	Coefficients of exponential regression in i-th approximation window
θ	Degradation increment to detect
$\tilde{\theta}$	Scaled degradation increment to detect
K^i	Number of samples in i-th window
N	Number of approximation windows for data set
P_i	Sum of squared errors in i-th prediction window
t_0	A point for prediction
T^i_{Start}	Beginning of i-th approximation window
T^i_{End}	End of i-th approximation window
$\mathcal{T}^i_{\mathrm{Start}}$	Beginning of i-th detection window
$\mathcal{T}^i_{\mathrm{End}}$	End of i-th detection window
T_{app}	Default approximation and detection window
T^i_{app}	Actual i-th approximation window
$\mathcal{T}^i_{\mathrm{app}}$	Actual i-th detection window
T_{pred}	Prediction window
T_{f}	Period of calculations
Δ	A half of prediction interval
y	Linear approximation in a detection window
m, c	Coefficients of the linear approximation y in a detection window

The article is structured as follows. Table 1 and Table 2 give lists of abbreviations and nomenclature. Section 2 gives a brief summary of relevant previous work, followed by Section 3 that discusses turbomachinery degradation in the context of maintenance events in industrial settings. Then the new algorithm is described, with examples of its predictive performance in Section 4. Section 5 presents an industrial case study showing the application of the algorithm to the industrial data sets, while Section 6 gives a comparison with other approaches. The paper ends with conclusions and recommendations.

2. Background and state of the art

Degradation of turbomachinery is typically associated with fouling, i.e. deposits forming on the surfaces inside the equipment. However,

measuring the level of fouling is only possible during manual inspections. To get an insight into the current degradation without performing an inspection, a *degradation indicator* is used.

2.1. Efficiency degradation

Typically, the degradation indicator d shows a relative deviation of the performance variables from the healthy values [16]:

$$d = \frac{Y - Y_{\mathrm{D}}}{Y} \tag{1}$$

where Y denotes the healthy value, and Y_D is the value in a degraded state. Usually, efficiency is the performance variable used for degradation assessment, with $Y = \eta$ denoting the expected efficiency and $Y_D = \eta_{\mathrm{D}}$ showing the actual value, with η_D as the efficiency in degraded state [17]. The expected value Y can be calculated based on data from the manufacturer of the turbomachinery or has to be estimated from thermodynamics [12]. The actual value η_{D} can be obtained from measurements using simplified thermodynamic relationships based on experimental correlations to find unknown thermodynamic quantities [18]. For the purpose of this work, it is assumed that the degradation indicator d is available in real-time.

2.2. Degradation modelling for turbomachinery

This paper proposes a data-driven algorithm that models the degradation indicator from Eq. (1) as a function of time. Modelling degradation as a function of time has been described by Meeker and Escobar [19]. They proposed a classification of approximation functions that includes a linear form and a negative exponential form i.e. a curve with a rate of change that slows down as it approaches an asymptote. They classified the main approaches to degradation modelling that had already been in use for modelling turbomachinery degradation. A review of methods used for degradation modeling, including reliability analysis in industrial applications was done by Bagdonavičius et al. [20] and Bagdonavičius and Nikulin [21], who indicated that regression models are well-suited for these purposes.

Tarabrin et al. [17] described fouling of a compressor as having an exponential approach to an asymptotic value that was reached after 1000–2000 h of operation. Cicciotti [12] used such a model in a real-time framework. Assuming that the degradation started immediately after the compressor was turned on, he fitted the exponential model to degradation data as new data points arrived and the window used for approximation increased. The expanding window allowed him to adapt to the varying time of stabilisation of the degradation indicator. He also considered maintenance events that could restore partial performance, but the exponential model was assumed strictly increasing. A similar approach was also assumed by Puggina and Venturini [10] who approximated degradation of a simulated gas turbine as a linear function of time.

Li and Nilkitsaranont [13] considered a simulated gas turbine engine in which the simulation included degradation and added random noise. The degradation was a linear function of time initially and then started to change more rapidly. They estimated the underlying degradation by linear regression and then switched to fitting of a quadratic form to follow the more rapid degradation. They were able to give predictions with confidence intervals for the remaining useful life.

Tsoutsanis and Meskin [22] also simulated a gas turbine and proposed a moving window approach with adaptation of the window size. They assumed degradation was monotonically increasing, and treated it as locally linear within each window. They showed a good fit between predictions of degradation from extrapolation from the previous window and the actual degradation.

Other prognostic models for this were developed by Hanachi et al. [23] and Kiakojoori and Khorasani [24]. While they differ in their methodology (a regression-based prognostic model and a dynamic neural network), their output is a prognosis of the degradation of gas turbines. Fentaye et al. [25] proposed an approach based on neural networks to overcome the difficulties related to disturbances. However, they did not test the method with real operating data from an industrial machine. Fentaye et al. [26] presented a survey on degradation modelling methods used in gas processing turbomachinery. As indicated by Cavarzere and Venturini [27] who compared regression with more complex tools, such as Kalman filtering or Bayesian forecasting, the regression is well suited to linear and nonlinear modelling and will be used in the current work.

2.2.1. Problem statement

It is problematical to apply the approaches discussed above to real data from offshore compressors and turbines. In practice, the underlying degradation indicator is not strictly increasing. Also, the degradation indicator calculated from operational measurements is significantly affected by disturbances, making it challenging to estimate and predict the true underlying degradation.

2.2.2. Contributions of the paper

The current work addresses the problem of detection and prediction of performance degradation in offshore turbomachines and presents a real data set with degradation of efficiency. It proposes an algorithm that uses both linear and exponential regression in an expanding moving window framework. It estimates the underlying degradation indicator in between maintenance events, makes predictions of future degradation, and gives confidence bounds for the predictions.

The paper gives guidelines for tuning the parameters of the algorithm, and shows that the tuning settings are portable from one machine to another. The performance of the algorithm is tested in an online monitoring scheme applied to data from offshore gas processing facilities. The benefit of using an on-line updated approximation model is that the underlying performance degradation indicator would be available for decision-making about when to perform maintenance. This will result in additional operational profits and reduced energy consumption, as argued by Aretakis et al. [8] and Schulze Spüntrup et al. [9].

3. Industrial data for efficiency degradation

3.1. Degradation of compressor efficiency

The studies on degradation modelling in this paper used a data set from an offshore compressor, courtesy of Equinor ASA, and a data set from an offshore turbine from Brekke et al. [28]. These data sets are discussed now, in order to introduce, motivate and explain the problems that will be addressed in the paper.

Fig. 1 shows the efficiency η_{D} of an offshore compressor. The data were collected over a period of approximately two years, with one sample per day. The black line represents the efficiency calculated from temperature and pressure measurements using the approach described by Campbell et al. [29] and Mokhatab et al. [18] who combined thermodynamics with experimental formulas to estimate unknown thermodynamic quantities. The black line is noisy, but shows a clear long-term trend (dashed red line). Prediction of this trend is the objective of this work. The noisy spikes show short-term variations, typically due to variations in the gas composition. These variations mask the underlying trend. Their influence on the prediction will be discussed in Section 6.

The overall degradation of efficiency consists of various types. Each can be (partly) reversed by specific maintenance actions [8,30]. The red triangular area at the top of Fig. 1 denotes the non-recoverable degradation, which can only be reversed by exchanging the internal parts of the compressor. The three maintenance types, *online washing*, *offline washing* and *compressor inspection*, reverse effects such as fouling. Online washing is the least expensive option and quick, but cannot remove all the deposits on the compressor blades. As the online washing cannot

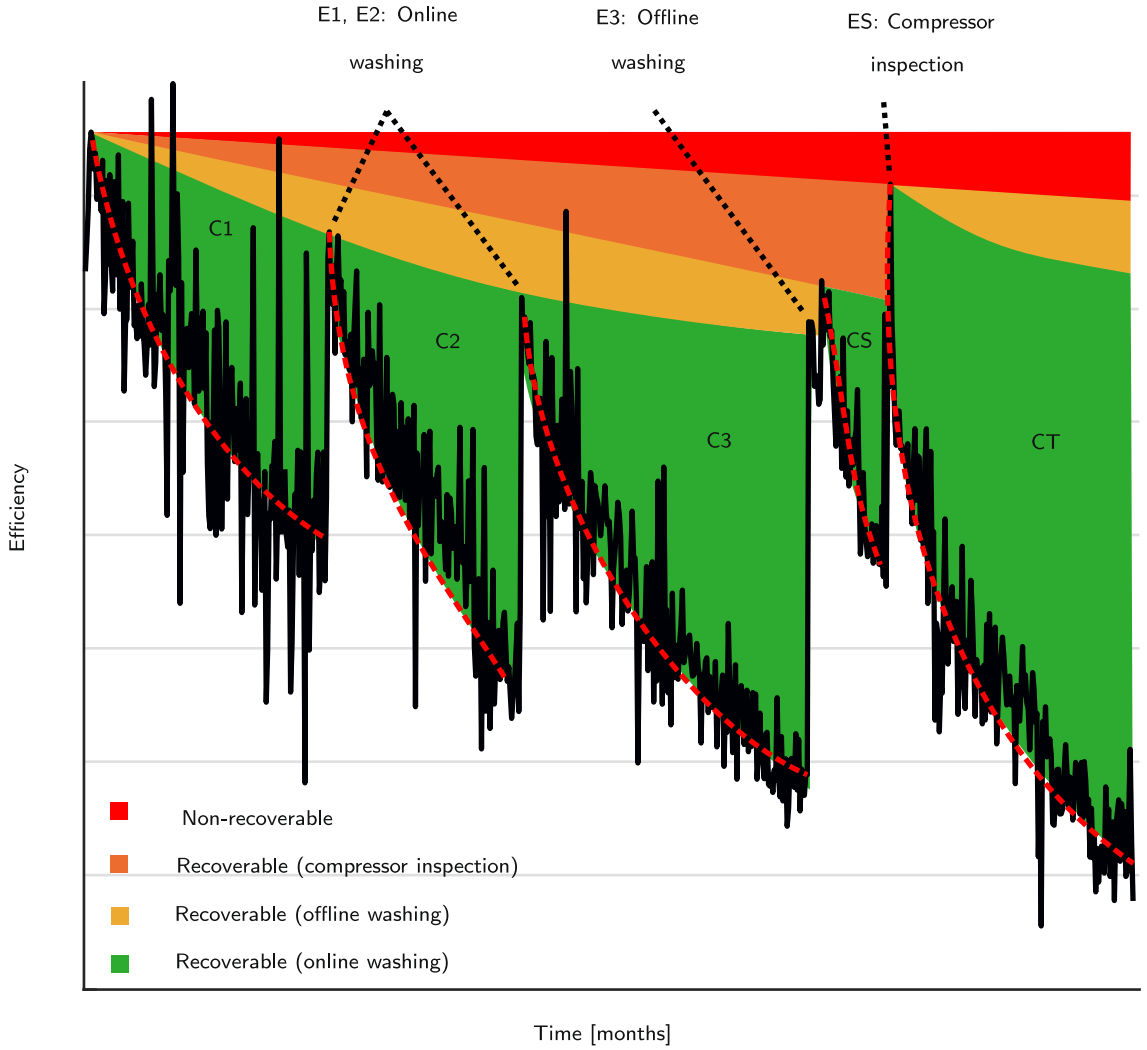


Fig. 1. Efficiency degradation with maintenance types over a period of approximately 18 months. The episodes (C1, C2, C3, CS, CT) are used for developing and testing the algorithm. Each episode ends with a maintenance event (E1, E2, E3, ES).

fully remove the deposits, a remainder builds up which requires offline washing. Offline washing is more thorough and removes more deposits than online washing. During a compressor inspection, the compressor is shut down and the casing opened which enables better washing and a check of all mechanical components in the compressor system. The trend data in Fig. 1 aligns well with the cleaning events at the end of the episodes C1, C2, and C3, and shows that the efficiency of the compressor was restored. After the compressor inspection was performed at the end of the episode CS, only the non-recoverable degradation remained.

3.1.1. Determining the degradation of efficiency by smoothing of data

As indicated in Section 2, the value of expected efficiency is necessary to calculate the degradation indicator d according to Eq. (1) or its modification from Eq. (1)

$$d = \widetilde{\eta} - \eta_{\mathrm{D}} \tag{2}$$

The expected efficiency in Eq. (1) and (2) can be taken to be the efficiency after a complete overhaul, which can be calculated, $\widetilde{\eta} = \eta_0$.

Fig. 2 shows the degradation indicator calculated from episode CT in Fig. 1. The calculated degradation indicator needs to be smoothed in order to give a better estimate of the true underlying value. The exponential model fits more accurately than a linear trend. This is expected because according to Syverud [31], the rate of increase of fouling slows down as the deposits built up.

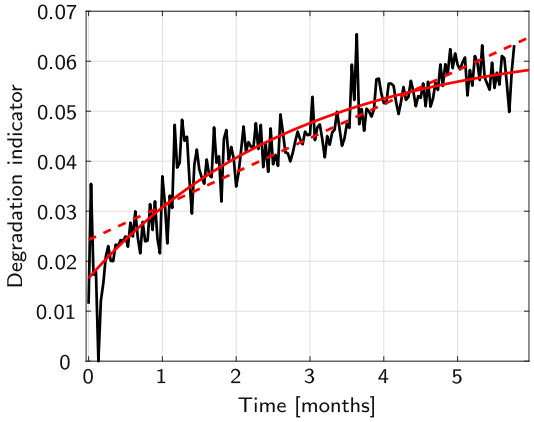


Fig. 2. Degradation indicator for the compressor from period CT.

Fig. 3 shows the three periods that were used for testing the algorithm. They are also noisy and indicate a need for a smoothing algorithm. Moreover, the red lines confirm that neither linear nor exponential approximations are good enough.

3.1.2. Degradation of turbine efficiency

A further industrial data set comes from a GE LM2500 engine

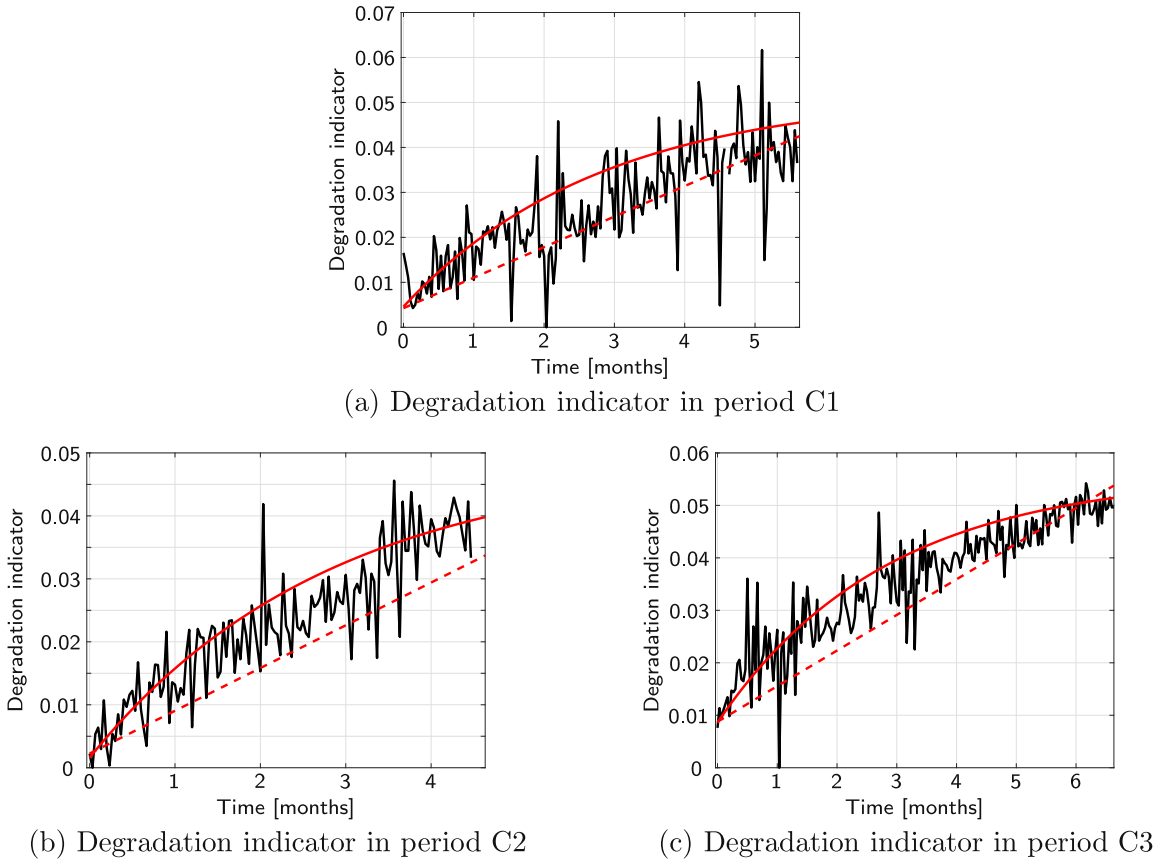


Fig. 3. Degradation indicator for the compressor divided in periods.

operating offshore in the Norwegian Sea and was obtained from Brekke et al. [28] using software developed by Rohatgi [32]. The original data were collected during a period of operation of approximately three months without maintenance events that could mitigate the loss of performance. The samples were collected every minute and this work uses the first sample from each day.

The degradation indicator for the turbine is presented in Fig. 4 and cannot be approximated with either linear or exponential trend.

3.1.3. Summary

The degradation data presented in Figs. 2–4 confirms that there is a need for improved approximation algorithms that would capture the changes in the degradation indicator, at the same time smoothing the noisy variations in the datasets. As indicated by Zagorowska et al. [33], a good candidate for the approximation of the degradation indicator would combine both linear and exponential functions. The algorithm proposed in the current work presents a model that switches in a systematic way between these two functional forms.

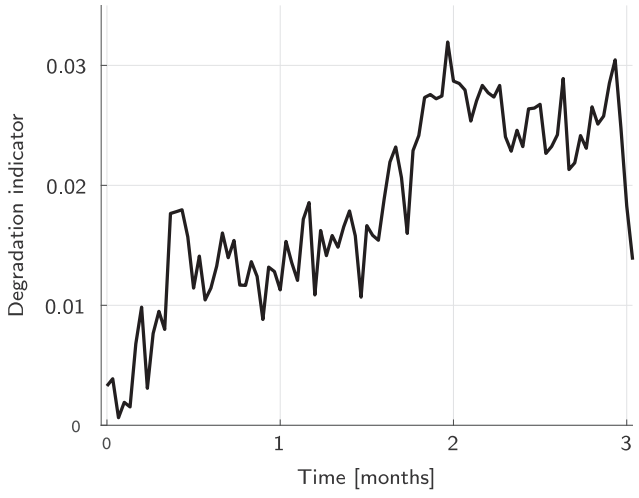


Fig. 4. Degradation indicator for a turbine from Brekke et al. [28] (T1).

4. Algorithm description

The aim of the algorithm is to estimate the value of the underlying degradation indicator $\hat{d}$ from the calculated degradation indicator d shown as the black lines in Figs. 2–4. Section 3 evinced that a single linear or exponential approximation is insufficient for the estimation of $\hat{d}$. This section introduces a moving window algorithm with a model for the underlying degradation that has constant, linear, and exponential terms:

$$\hat{d}(t) = b_0^i + b_1^i t + b_2 \exp(-b_3^i t) \tag{3}$$

where i denotes the current approximation window. It adapts by switching between a linear and exponential approximation as the window moves. The switching is done by adjusting the values of the model parameters b_0^i to b_3^i, as described in Section 4.2.1.

4.1. Moving windows

The algorithm makes use of three moving windows:

- An approximation window in which the measured degradation indicator (black lines in Figs. 2–4) is approximated with either a linear or an exponential function. The i-th approximation window is defined by a time interval $[T^i_{\text{Start}}, T^i_{\text{End}}] = [T^i_{\text{Start}}, T^i_{\text{Start}} + T^i_{\text{app}}]$.
- A detection window needed for switching the functional form in the

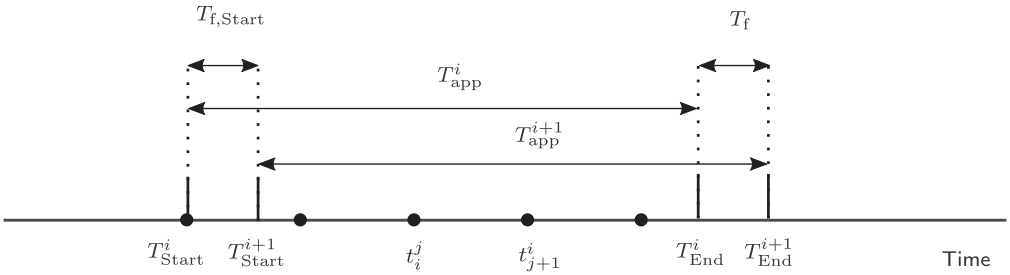


Fig. 5. Explanation of moving approximation window, adapted from Zagorowska et al. [33].

approximation window. The i-th approximation window is defined as a time interval $[\mathcal{T}^i_{\mathrm{Start}}, \mathcal{T}^i_{\mathrm{End}}] = [\mathcal{T}^i_{\mathrm{Start}}, \mathcal{T}^i_{\mathrm{Start}} + \mathcal{T}^i_{\mathrm{app}}]$.

- A prediction window in which the model fitted to data in the approximation window is extrapolated. It is defined by an interval $[T^i_{\mathrm{End}}, T^i_{\mathrm{End}} + T_{\mathrm{pred}}]$ with constant T_{pred}.

Online smoothing of the calculated degradation indicator, and its forward prediction, require a moving window approach because of the varying nature of the degradation indicator. Fig. 5 illustrates the idea of a moving approximation window. The i-th approximation window starts at T^i_{Start}, ends at T^i_{End} and lasts for T^i_{app}. The subsequent approximation window, $i+1$, is described with T^{i+1}_{Start} and T^{i+1}_{End}. The time difference between the ends of two consecutive windows, $T_f = T^i_{\mathrm{End}} - T^{i+1}_{\mathrm{End}}$, is constant, and characterises the update rate of the approximation. The dots on the time axis denote the measurement instants t^i_j in i-th window.

The approximation window T^i_{app} and the difference between two consecutive starting points for the approximation $T_{\mathrm{f,Start}} = T^{i+1}_{\mathrm{Start}} - T^i_{\mathrm{Start}} \in \{0, T_{\mathrm{f}}\}$ are derived from the adaptation procedure described in subsequent sections.

4.2. Fitting of the model to data within an approximation window

4.2.1. Numerical considerations

Identification by regression of all four parameters b_0 to b_3 in Eq. (3) can give non-unique estimates. If the underlying degradation indicator does not change much within a window, as seen for instance from 0.5 to 1 months in Fig. 4, then the linear term dominates and there are many combinations of b_1, b_2, b_3 that would fit such data. In this case, the algorithm must select one combination. This is done by setting b_2 to zero when the underlying degradation indicator does not change significantly within a window. The model is then linear, $\hat{d} = b_0 + b_1 t$, and is fitted to the data in the approximation window using linear regression.

On the other hand, the underlying degradation indicator changes significantly at other times, as seen for instance between 1.5 and 2 months in Fig. 4. The literature reviewed in Section 2.2 shows that, for underlying physical reasons, an exponential model provides a good fit during such periods. If the underlying degradation indicator changes significantly within a detection window then the parameter b_1 is set to zero and the model is $\hat{d} = b_0 + b_2\exp(-b_3 t)$. The model is fitted using non-linear regression.

The above arguments suggest that adaptation requires an assessment for each window of whether there has been a significant change in the underlying degradation indicator within the window. This is done with a parameter θ as described in Appendix C.

An additional logical step is needed when starting to use the exponential model, because it is necessary to determine the starting point for exponential approximation. This is done by examining whether significant changes in the value of $\hat{d}$ began in the current (i-th) window, or in the previous window. If there was no significant change in the previous window, it means that the degradation indicator only started to increase or decrease significantly in the current window. The middle of the current detection window is taken as a starting point for the exponential approximation, and the exponential model is fitted to the data in the second half of the window. Taking the middle of the detection window as a starting point for the approximation results in shortening of the approximation window from the default value T_{app} to T^i_{app}. This is to ensure that the exponential model is fitted to the non-linear part from the detection window.

4.2.2. Linear regression

It is assumed that the observed value of $d(t^i_j)$ at each time moment $t^i_j \in [T^i_{\mathrm{Start}}, T^i_{\mathrm{End}}]$, $j = 1, \ldots, K^i$, with K^i denoting the number of samples in i-th approximation window, is a random variable [34], whose mean value is the true underlying value of the degradation indicator at time t^i_j. The expected value of the degradation indicator d at each t^i_j, $E(d(t^i_j))$, is a smoothed estimate of the true underlying value of the degradation indicator. Such an estimate removes variability that is present when the degradation is calculated from operating data. The expected value is calculated as:

$$E(d(t^i_j)) = b^i_0 + b^i_1 t^i_j \tag{4}$$

where b^i_0, b^i_1 are unknown coefficients found using linear regression.

The estimated degradation indicator in i-th approximation window $[T^i_{\mathrm{Start}}, T^i_{\mathrm{End}}]$ is

$$\hat{d} = b^i_0 + b^i_1 t \tag{5}$$

4.2.3. Nonlinear regression

When linear regression is not sufficient, an exponential model is used in i-th approximation window:

$$f(t, b^i) = b^i_0 + b^i_2\exp(-b^i_3 t) \tag{6}$$

The model (6) is obtained from Eq. (3) by setting $b_1 = 0$. Using non-linear regression with constraints $b^i_0 \in \mathbb{R}$, $b^i_2 \in \mathbb{R}$ and $b^i_3 > 0$ yields an estimate of the degradation indicator in i-th approximation window $[T^i_{\mathrm{Start}}, T^i_{\mathrm{End}}]$

$$\hat{d} = f(t, b^i) \tag{7}$$

To find the parameters in both linear and non-linear cases, the function `fit` from the Curve Fitting Toolbox in Matlab is used.

4.3. Determining the change in degradation

The process for choosing the functional form for approximation is conducted in i-th detection window by fitting a linear model of the form $y = mx + c$ to the data within the detection window, where x is time. Although this looks similar to the linear model $\hat{d} = b_0 + b_1 t$ discussed earlier, y is not necessarily the same as $\hat{d}$. The quantity $\hat{d}$ is an estimate of the true underlying value of the degradation indicator within the approximation window. If it transpires from the analysis in the detection window that a linear model is appropriate for the given approximation window, then $\hat{d} = y$ in the approximation window. However, y will be generated for all detection windows, even for those where an exponential model is appropriate. If an exponential model is

appropriate, then y is not the same as $\hat{d}$ throughout the approximation window.

An estimate of the change in $\hat{d}$ from the start to the end of the detection window is given by $\Delta y = m(\mathcal{T}_{\text{End}} - \mathcal{T}_{\text{Start}})$, where m is the gradient of the fitted straight line, and $\mathcal{T}^i_{\text{End}}$ and $\mathcal{T}^i_{\text{Start}}$ are the times at the end and start of the i-th detection window.

4.4. Choosing the model structure

The model structure is chosen by comparing Δy in i-th detection window with

$$\widetilde{\theta} = \frac{\theta \cdot \mathcal{T}^i_{\text{app}}}{T_{\text{app}}} \tag{8}$$

where θ is a constant parameter and $\mathcal{T}^i_{\text{app}} = \min\{T_{\text{app}}, \mathcal{T}^i_{\text{End}} - \mathcal{T}^i_{\text{Start}}\}$. The selection of an appropriate value for θ is discussed in Appendix C. Eq. (8) can be rewritten

$$\widetilde{\theta} = \begin{cases} \theta, & \text{if } T_{\text{app}} \leqslant \mathcal{T}^i_{\text{End}} - \mathcal{T}^i_{\text{Start}} \\ \frac{\theta \cdot (\mathcal{T}^i_{\text{End}} - \mathcal{T}^i_{\text{Start}})}{T_{\text{app}}}, & \text{if } T_{\text{app}} > \mathcal{T}^i_{\text{End}} - \mathcal{T}^i_{\text{Start}} \end{cases} \tag{9}$$

The first case is valid if the default window is used for detection, whereas the second case is used after shortening of the window, as indicated in Section 4.2.1.

If $|\Delta y| \leqslant \widetilde{\theta}$, then the underlying degradation will be determined using linear regression by setting parameter b_2 to zero and fitting the model $\hat{d} = b_0 + b_1 t$ to the data in the approximation window. If $|\Delta y| > \widetilde{\theta}$, the underlying degradation will be determined from nonlinear regression by fitting the model $\hat{d} = b_0 + b_2 \exp(-b_3 t)$. The test uses $|\Delta y|$ rather than Δy because the degradation indicator can decrease in some circumstances.

The logical steps in the algorithms are presented in the flow chart in Fig. 6. The left path represents fitting the linear model, the middle path represents the first window after detection that $|\Delta y| > \widetilde{\theta}$, and the right hand path presents fitting the exponential model in subsequent windows. The three blocks are described in detail in Appendix A.

4.5. Prediction intervals

Prediction consists of extrapolation of the approximating linear or exponential function into the future, i.e. for $t \in [T^i_{\text{end}}, T^i_{\text{end}} + T_{\text{Pred}}]$. The uncertainty of the prediction is captured by prediction intervals assuming that the errors $\varepsilon_i = d - \hat{d}$ are normally distributed [34].

4.5.1. Prediction intervals

The prediction intervals at a selected percentile $100(1-\alpha)$ at time $t = t_0$ are given by:

$$\hat{d}(t_0) - \Delta \leqslant d(t_0) \leqslant \hat{d}(t_0) + \Delta \tag{10}$$

with

$$\Delta = t_{\alpha/2, K^i - 2} \sqrt{\hat{\sigma}^2 (1 + \hat{a}_0^{\mathsf{T}} (\mathbf{A}^{\mathsf{T}} \mathbf{A})^{-1} \hat{a}_0)} \tag{11}$$

where $t_{\alpha/2, K^i - 2}$ is the $100(1-\alpha)$ percentile of Student's t-distribution with $K^i - 2$ degrees of freedom and where $\hat{\sigma}^2$ is the estimated variance of d. The $\hat{a}_0$ is a vector of derivatives of $f(t, b^i)$ calculated at t_0 and $\mathbf{A}$ is a matrix of derivatives of $f(t, b^i)$ calculated at times $t_p \in [T_{\text{Start}i}, T^i_{\text{End}} + T_{\text{pred}}]$ with $f(t, b^i)$ given by the appropriate form of the right hand side of Eq. (3). The formulas for $\hat{a}_0$ and $\mathbf{A}$ for the linear and exponential cases are given in Table B.5 in Appendix B. The numerical values for Δ were obtained in Matlab using the function `predint`.

The prediction intervals given by Eq. (10) show that $100(1-\alpha)\%$ of the data points in the prediction window will fall in the prediction interval. This paper assumes $\alpha = 5$ which means that 95% of the data points will be within the prediction interval.

Fig. 7 illustrates the prediction intervals in the online moving window algorithm for four windows, R and P1, P2, P3. Window R is enlarged in the bottom right and presents the approximation (thick solid red curve) of the degradation indicator (black) in the current window. The red dotted line shows the expected value $\hat{d}$ in the prediction window defined by T_{pred} (light green shading). The thin blue line shows the prediction intervals for data from the approximation window $[T^i_{\text{Start}}, T^i_{\text{End}}]$. The dotted lines show the prediction intervals for $t \in [T^i_{\text{End}}, T^i_{\text{End}} + T_{\text{pred}}]$.

Windows P1, P2, and P3 present how the prediction intervals depend on the size of the approximation window. The shortest approximation window P1 has the largest prediction intervals. As indicated by Montgomery and Runger [34], the prediction interval is a decreasing function of the number of samples K^i. Therefore, increasing number of samples K^i in i-th approximation window, for example by expanding the duration of the time window P1 to P2 and P3, results in narrowing of the prediction intervals. The same effect can be obtained by increasing the number of samples for fixed window size.

5. Industrial case study

The proposed adaptive algorithm was applied to real data sets coming from off-shore applications as described in Section 3. For the purpose of this work, the data from Fig. 1 were divided in two sets:

- Tuning data (CT) were used for tuning of the algorithm
- Test data were used for comparison with other approaches (C1, C2 and C3 in Fig. 1)

The objective of testing the algorithm on data sets coming from the same compressor that was used for tuning is to evaluate the adaptation method and the quality of predictions. Furthermore, the algorithm is tested on the turbine data set T1 to evaluate whether the settings of the algorithm are portable.

5.1. Tuning

To apply the algorithm from Section 4 to industrial data, it is necessary to define the values of its settings: the default approximation window T_{app}, the prediction window T_{pred}, and the desired threshold θ. The values of the settings can be found by analysing historical datasets. The algorithm was tuned using the tuning data set CT. Then the same settings were applied to test data C1, C2, C3 and T1 to analyse the accuracy of prediction and the influence of the tuned parameters on the performance.

The value of T_f was set to one day, as the new values of the degradation indicator were available once per day.

This section discusses the choice of the approximation and prediction window and presents the results of the tuning. Further details on the choice of the detection threshold θ are in Appendix C.

5.1.1. Approximation window

As indicated by Tarabrin et al. [17], compressor degradation due to fouling typically increases over 1000–2000 h of operation (42–84 days) and then stabilizes. Thus, all variations shorter than 42 days in the calculated degradation indicator (the black lines in Fig. 1) are considered to be due to disturbances. This suggests that the default window size should be in the range 42–84 days of operation to mitigate the influence of the disturbances. The approximation window for the compressor was chosen to be 42 days, $T_{\text{app}} = 42$, to capture the fastest degradation, while being longer than variations due to disturbances.

Syverud [31] indicated that the duration of disturbances would depend on the type of the turbomachinery. Thus, a procedure for adapting the approximation window to a different piece of equipment is shown in Section 5.2.2.

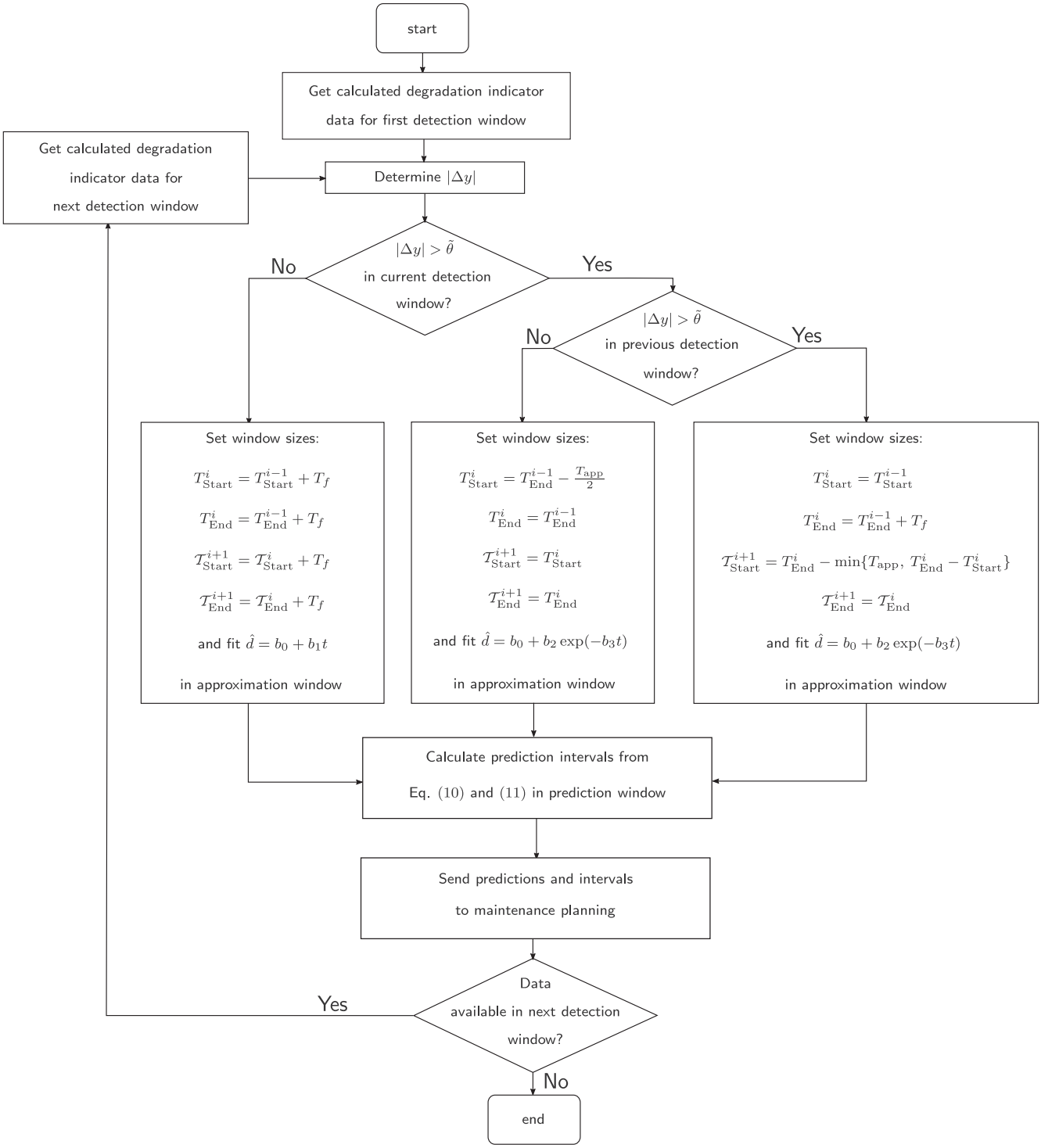


Fig. 6. Flowchart of the Adaptive Degradation Prediction algorithm (ADP).

5.1.2. Prediction window

The prediction window will have an impact on the accuracy of the prediction and the prediction interval. Two cases are considered:

- T_{pred} = 10 days - short-term prediction window
- T_{pred} = 30 days - medium-term prediction window.

The choice of the prediction window depends on the application. In the current work, short-term predictions are applicable to scheduling of online washings, whereas medium-term predictions relate to offline washings [9].

Both prediction windows were explored in order to examine their influence on the algorithm and to give a recommendation that would be portable to other applications. They were chosen to make sure that the approximation window and the prediction window together fit in 2000 h, as this is the maximal expected duration of a single degradation period, from the start to stabilisation:

$$T_{\text{app}} + T_{\text{pred}} = 42 + 10 = 52\,\text{days} < 84\,\text{days} \tag{12}$$

for the short prediction T_{pred} = 10 days and

$$T_{\text{app}} + T_{\text{pred}} = 42 + 30 = 72\,\text{days} < 84\,\text{days} \tag{13}$$

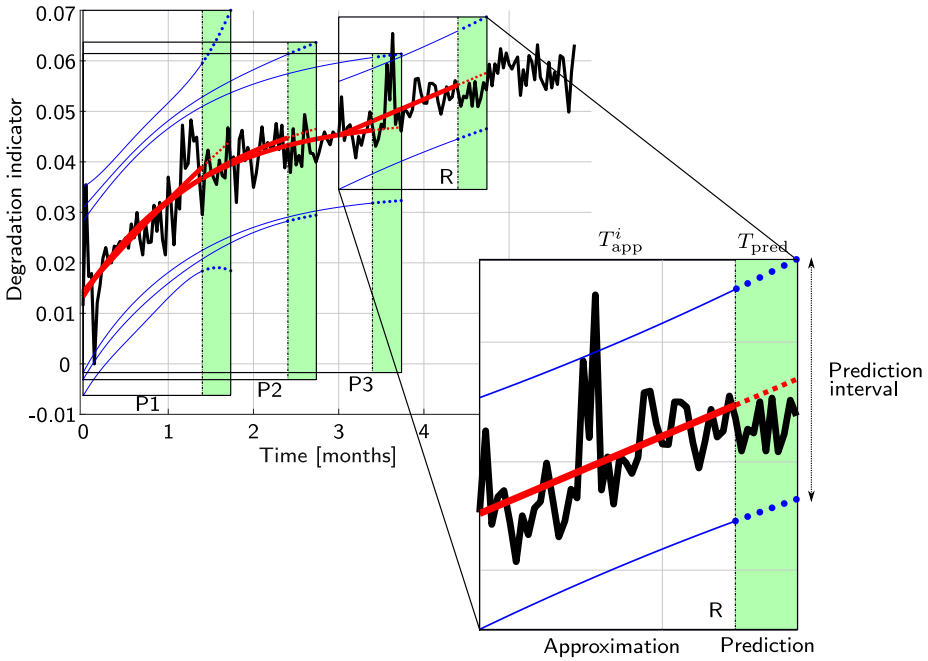


Fig. 7. An online application of the algorithm on the tuning data set, with the structure of a time window in window R, and presenting the expanding approximation windows in time windows P1, P2, and P3.

for medium-term prediction with $T_{\text{pred}} = 30$ days.

5.1.3. Tuning results

The default T_{app}, θ, and T_{pred} were chosen based on the knowledge about past degradation data. Following the discussions in Sections 5.1.1, 5.1.2 and Appendix C, the default values were set to: $T_{\text{app}} = 42$, $\theta = 0.005$ for $T_{\text{pred}} = 10$ days (short term prediction), and $\theta = 0.006$ for $T_{\text{pred}} = 30$ days (medium term prediction). The objective of this section is to illustrate the results of the algorithm applied to period CT, before testing it on the data sets from C1, C2, C3, and T1.

The results of approximation and prediction on the data set CT are depicted in Fig. 8. Figs. 8a and b show the degradation indicator d (black) with the approximating functions $\hat{d}$ (red) for the two prediction periods, $T_{\text{pred}} = 10$ days and $T_{\text{pred}} = 30$ days. In both cases, the red approximation $\hat{d}$ follows the underlying trend of the degradation indicator. However, the influence of the prediction window T_{pred} is noticeable in the third month, where the red curves are above the black indicator, i.e. the prediction diverges from the value calculated from the measurements if the prediction window is longer. However, when the approximation adapts to the significant changes of the degradation indicator, the predictions become accurate.

The moments when the algorithm switches from fitting of a linear model to fitting of an exponential model are visible in Figs. 8c and d. Red filled circles show that $\widetilde{\theta}$ was exceeded (black horizontal lines), i.e. where $|\Delta y| > \widetilde{\theta}$. The first red circle means that a check is performed on the value of Δy in the preceding window. As it is the first one, the algorithm chooses the middle path in Fig. 6, and uses exponential approximation in the second half of the current window. The middle point of the window is then fixed, whereas T_{End} increases as new data arrive. As long as the change was detected, the algorithm approximated the degradation indicator with an exponential function over an expanding approximation window (right hand path in Fig. 6). When a change was no longer detected, $|\Delta y| \leqslant \widetilde{\theta}$ (left path in Fig. 6), the algorithm switched to linear approximation (grey circles). Subsequently, the degradation indicator increased again, and the algorithm detected that $|\Delta y| > \widetilde{\theta}$ and switched back to exponential approximation (month four for both $T_{\text{pred}} = 10$ days and $T_{\text{pred}} = 30$ days).

5.1.4. Prediction intervals for tuning data set

As long as the algorithm is detecting $|\Delta y| > \widetilde{\theta}$, the approximation window is extended from the value $\frac{T_{\text{app}}}{2}$. Therefore, the uncertainty of prediction is reduced because more data points are available for approximation. The reduction of the uncertainty is also visible in the prediction intervals, as both for $T_{\text{pred}} = 10$ days and $T_{\text{pred}} = 30$ days, the prediction intervals are wide at the beginning, but then they narrow down and follow the degradation indicator more closely. Fig. 8a shows the prediction intervals for the whole data set and indicates resetting of the prediction intervals when $|\Delta y| > \widetilde{\theta}$ was detected, as the prediction intervals widen in month four.

The longer approximation window resulted in detecting longer periods when $|\Delta y| \leqslant \widetilde{\theta}$ (there are more grey circles between the threshold lines for longer T_{pred}). A period with $|\Delta y| \leqslant \widetilde{\theta}$ means that there are fewer switches between linear and exponential approximation. This tendency for less frequent changes had an impact on the prediction intervals. As the detection of $|\Delta y| > \widetilde{\theta}$ is related to shortening of the approximation window, frequent changes result in frequent shortenings, and, in consequence, fewer datapoints are used for approximation. Hence shorter approximation windows yield more uncertainty and wider prediction intervals. Therefore, the more frequently $|\Delta y| > \widetilde{\theta}$ is detected, the more uncertain the prediction.

5.2. Testing

The values of the parameters used for testing come from the tuning procedure, described in Section 5.1, and gathered in Table 3.

5.2.1. Testing on compressor data sets

The results of application to the data sets C1, C2, and C3 are depicted in Fig. 9. The left column shows the approximation for $T_{\text{pred}} = 10$ days, and the right side shows the approximation for $T_{\text{pred}} = 30$. In both cases, the red curves follow the trend of the black degradation indicator. Nonetheless, Fig. 9 shows that medium term predictions are less accurate than short-term prediction. In particular, the inaccuracies are visible for C3 in the third month (Fig. 9f), where the algorithm underestimates future degradation values in the medium-term window compared to the shorter prediction window in Fig. 9f. This is due to the

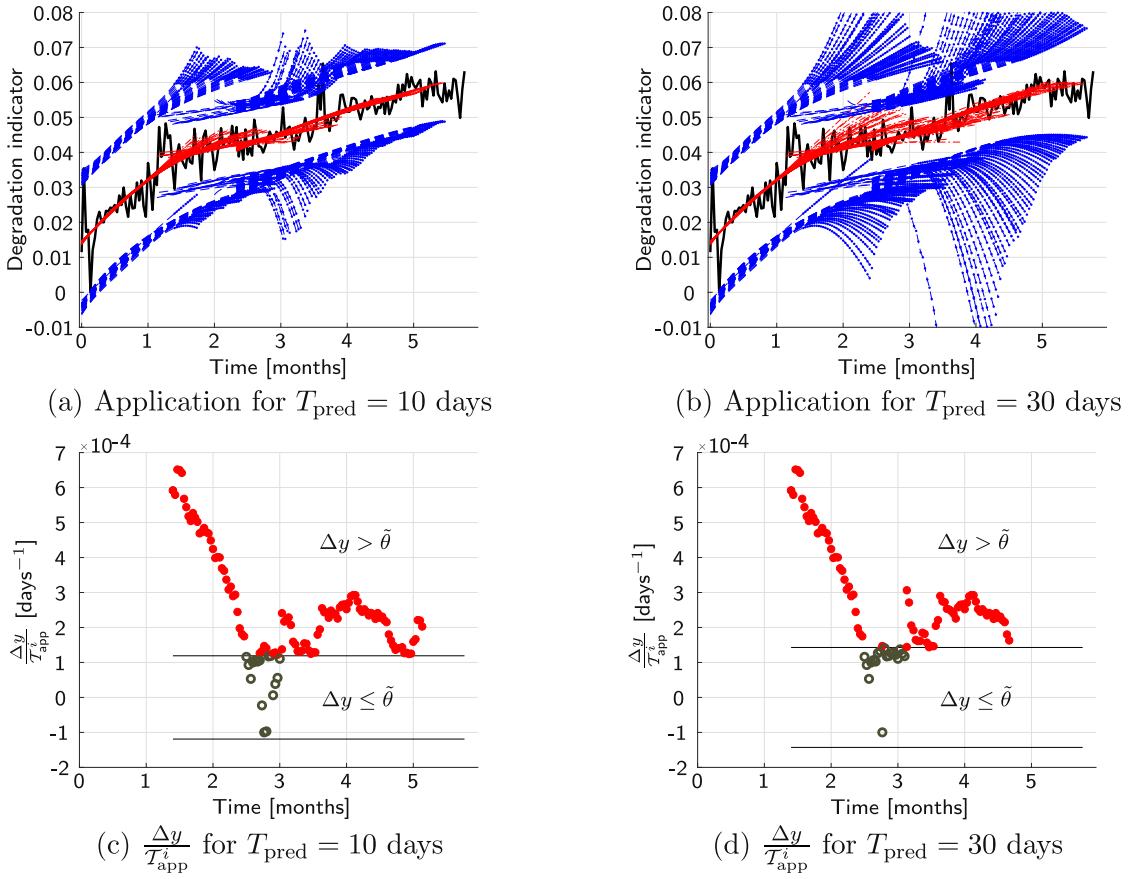


Fig. 8. Application of the algorithm to the tuning data set CT for $T_{\text{pred}} = 10$ days in (a) and $T_{\text{pred}} = 30$ days in (b), together with respective $\frac{\Delta y}{T^i_{\text{app}}}$ (c and d).

assumption that the underlying degradation indicator can be approximated by extrapolation of the same function for $t \in [T^i_{\text{End}}, T^i_{\text{End}} + T_{\text{pred}}]$.

The medium-term prediction window resulted also in more uncertainty in the prediction, characterized by the blue prediction intervals. The prediction intervals for $T_{\text{pred}} = 30$ days are approximately twice as wide as the prediction intervals for the shorter window. In particular after detection of $|\Delta y| > \tilde{\theta}$ resulting in a shortening of the approximation window, the prediction intervals increases for medium term window.

The results confirm that tuning of the algorithm enables application for multiple periods of degradation for the same compressor. The settings of the algorithm can be applied for real time approximation and prediction of the degradation indicator after tuning on a historical dataset.

5.2.2. Testing for turbine data set

To further evaluate the performance of the algorithm and to check the portability of the settings, the algorithm was applied to the data set T1 from an offshore turbine. Contrary to the previous data sets, the data set T1 is shorter (three months). The stabilisation periods are also of comparable duration to period with increasing degradation indicator (two weeks). For instance, in the first month in Fig. 10, the black degradation indicator stabilizes around 0.012 after two weeks and remains around this value for another two weeks. To follow the underlying degradation indicator, the default approximation window was fixed to two weeks, $T_{\text{app}} = 14$ days. It is also assumed that the degradation indicator d in a turbine is similar to the degradation indicator in a compressor. Therefore, the value of θ for a turbine has the same interpretation as discussed in Section 5.1, but with a shorter T_{app}. The parameter $\theta = \theta_{\text{comp}}$ is scaled to $\theta = \theta_{\text{turb}}$ using the formula:

$$\theta_{\text{turb}} = \frac{\theta_{\text{comp}} T_{\text{app,turb}}}{T_{\text{app,comp}}} \tag{14}$$

where $T_{\text{app,turb}}$ denotes the default approximation window for the turbine, $T_{\text{app,comp}}$ denotes the default approximation window for the compressor.

The prediction windows of 3.5 and 7 days were chosen to fulfill $14 + 3.5 < 21$ days and $14 + 7 < 21$ days which was considered the length of a period when $\Delta y > \tilde{\theta}$ for the turbine dataset by Zagorowska et al. [33]. The corresponding values of θ are $\theta = 0.0017$ for 3.5 days and $\theta = 0.002$ for 7 days.

The results of the application of the algorithm to the data set T1 are depicted in Fig. 10. Again, the red curves follow the underlying degradation indicator $\hat{d}$ for both short and medium term prediction. However, due to more pronounced transitions between fitting of an exponential model and fitting of a linear model, the inaccuracies

Table 3
Algorithm settings for the case studies.

	Compressor (CT, C1, C2, C3)	Turbine (T1)
Medium-term prediction window	$T_{\text{pred}} = 30$ days, $\theta = 0.006$	$T_{\text{pred}} = 7$ days, $\theta = 0.002$
Short-term prediction window	$T_{\text{pred}} = 10$ days, $\theta = 0.005$	$T_{\text{pred}} = 3.5$ days, $\theta = 0.0017$
Default approximation window	$T_{\text{app}} = 42$ days	$T_{\text{app}} = 14$ days

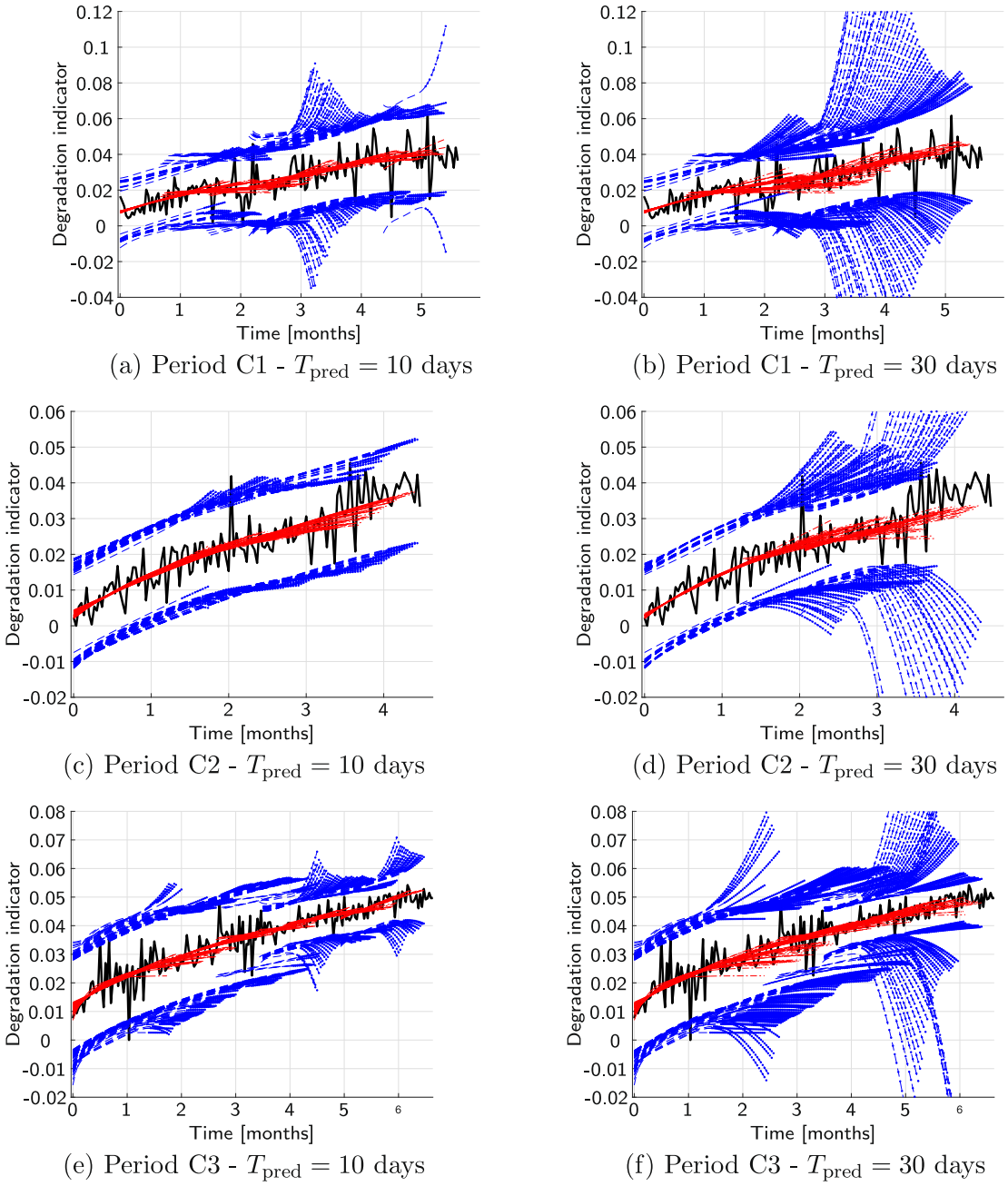


Fig. 9. Results for test data sets, C1, C2, C3, for the offshore compressor.

resulting from the longer prediction window are more visible (Fig. 10b). The algorithm overestimates the degradation in weeks three and 10, i.e. when the underlying degradation indicator stabilizes. When the degradation indicator starts to increase, the predictions underestimate the underlying value, which is visible in week eight. The inaccuracies suggest that the algorithm should be primarily applied for short-term prediction. However, the prediction intervals (dashed blue lines) capture the uncertainty related to medium term prediction and can be used for decision support in planning.

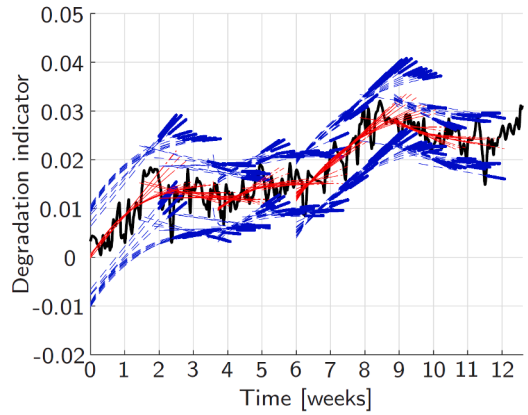
The T1 data set also shows how the algorithm successfully adapts to decreasing values of the underlying degradation indicator. In weeks 3–5 and 10–12, a decrease in the indicator was detected, and the algorithm switched from a linear to a decreasing exponential function. These transitions for both prediction windows are depicted in Fig. 10c and d. The red filled circles denoting $\Delta y > \widetilde{\theta}$ are below the threshold, so the linear approximation used for detection is decreasing and $\Delta y < -\widetilde{\theta}$.

Finally, the dashed blue lines in Fig. 10a and b show the influence of the number of samples on the prediction intervals. The prediction intervals are narrower for the turbine data set than for the compressor data set, despite shorter approximation windows. The narrow intervals were obtained thanks to frequent sampling, i.e. for the turbine data set the degradation data is available every minute, whereas the compressor had one sample per day.

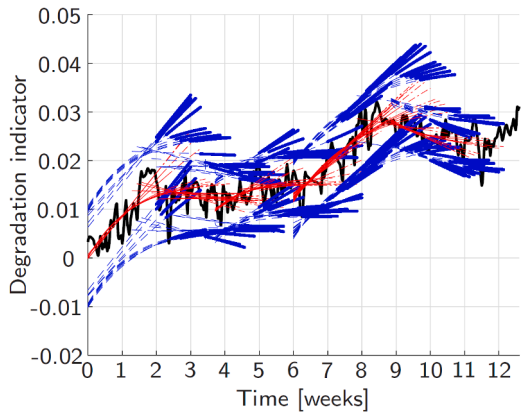
6. Comparison with prediction with other approaches

The algorithm was compared with four other approaches:

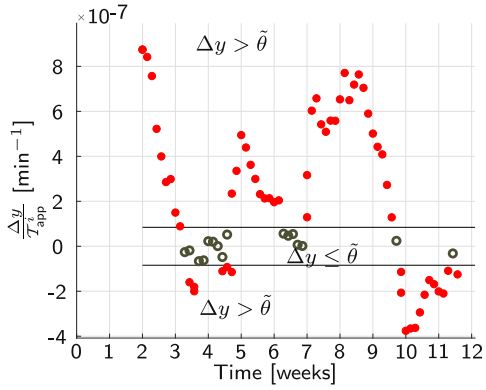
- Mean value over constant approximation window (Mean), 42 days for the compressor and 14 for the turbine
- Linear regression over constant approximation window (LinearC), 42 days for the compressor and 14 for the turbine



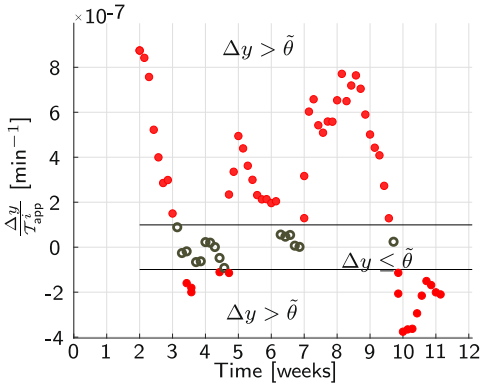
(a) Degradation indicator (black) with approximating lines (red) for $T_{\text{pred}} = 3.5$ days



(b) Degradation indicator (black) with approximating lines (red) for $T_{\text{pred}} = 7$ days



(c) $\frac{\Delta y}{T^i_{\text{app}}}$ for $T_{\text{pred}} = 3.5$ days



(d) $\frac{\Delta y}{T^i_{\text{app}}}$ for $T_{\text{pred}} = 7$ days

Fig. 10. Application to the turbine data set T1 from Brekke et al. [28]. The results of approximation are in a and b and the corresponding values of $\frac{\Delta y}{T^i_{\text{app}}}$ in c and d.

- Linear regression over varying approximation window as the algorithm (adaptation of the algorithm proposed by Tsoutsanis et al. [14]) (Linear)
- Exponential approximation with fixed starting point and expanding approximation window [12] (Fixed Exponential)

6.1. Performance metric for comparison

The performance of the algorithms was measured by a performance metric given by Eq. (15):

$$I = \sum_{i=1}^{N} \frac{P^i}{K^i} \tag{15}$$

where

$$P^i = \sum_{k=1}^{K^i} \left(d(t_k^i) - \hat{d}\,(t_k^i)\right)^2 \tag{16}$$

where i denotes the number of windows for each data set, K^i is the number of samples in i-th prediction window, $k = 1, ..., K^i$ and $\hat{d}\,(t_k^i)$ denotes the estimated value of the degradation indicator at time t_k^i. The metric from Eq. (15) measures the error between the expected degradation indicator $\hat{d}$ and the measured values d. Thus, the smaller the value of I, the better the prediction. The metric from Eq. (15) is equivalent to integrating the mean value of a sum of squares of the differences in each prediction window $[T^i_{\text{End}}, T^i_{\text{End}} + T_{\text{pred}}]$ over the whole data set. Taking the mean value mitigates the influence of short spikes in the data set.

6.2. Compressor comparison

The performance of the algorithm was assessed against other approaches over the two prediction windows, 10 days and 30 days. The results of the comparison for the tuning data set from Fig. 2 are depicted in Fig. 11 which shows the value of P^i as a function of the window number i for both prediction windows $T_{\text{pred}} = 10$ days (Fig. 11a) and and $T_{\text{pred}} = 30$ days (Fig. 11b). The red curve with dots depicts the performance of Adaptive Degradation Prediction, ADP, the light blue with crosses curve shows the results of the linear approximation over the same window as the algorithm, Linear, dotted magenta presents the linear approximation over the fixed window $T_{\text{app}} = 42$ days, LinearC, dark blue with circles curve shows exponential approximation with expanding approximation window, Fixed Exponential, and dashed green shows the results of averaging over constant $T_{\text{app}} = 42$ days, Mean.

The red curve with dots shows that the ADP algorithm presented in this paper with moving window and switching ability gives the best results. This result is also confirmed in the first and sixth rows in Table 4. For both prediction windows, $T_{\text{pred}} = 10$ and $T_{\text{pred}} = 30$, the ADP in the first part ($i < 50$) behaves like the Fixed Exponential approximation (dark blue with circles) and is better than both linear approximations (light blue with crosses and dotted magenta). In particular for medium-term prediction window, i.e. longer T_{pred}, the linear approximations were inaccurate at the beginning. This is due to the fact that the linear approximations do not follow the underlying degradation indicator and assume that it can be approximated by extrapolation of a

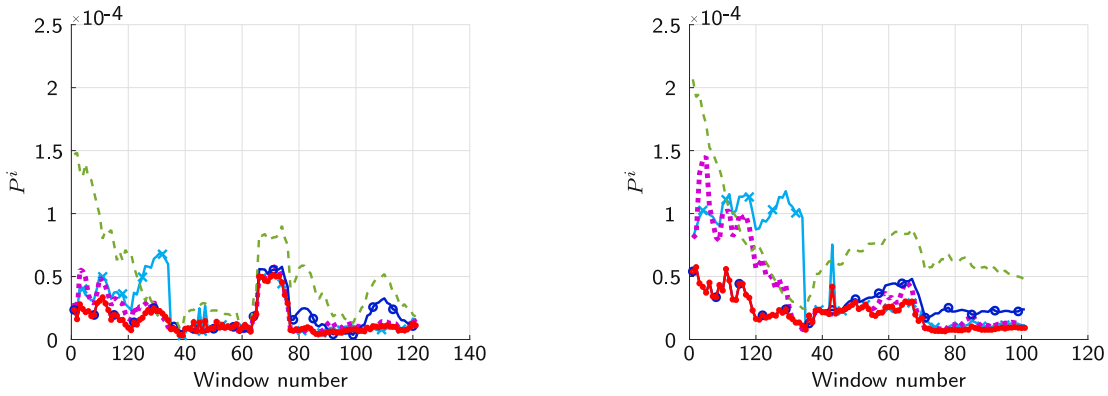


Fig. 11. Comparison with other approaches for the tuning data set CT. Legend: red with dots – ADP; light blue with crosses – Linear; dotted magenta – LinearC; dark blue with circles – Fixed Exponential; dashed green – Mean.

Table 4
Value of performance indicator I for five algorithms, for two prediction windows. The values show the difference in percentage from the best obtained value for each data set: CT, C1, C2, C3 for the compressor, T1 for the turbine

	Data set and prediction window	ADP	Linear	LinearC	Fixed Exponential	Mean
1.	CT – short	0	50	28	22	211
2.	C1 – short	2	6	2	0	27
3.	C2 – short	4	0	0	4	143
4.	C3 – short	5	5	0	5	141
5.	T1 – short	0	30	10	70	30
6.	CT – medium	0	124	81	38	252
7.	C1 – medium	0	8	4	1	55
8.	C2 – medium	0	25	15	15	205
9.	C3 – medium	18	5	0	14	241
10.	T1 – medium	0	38	23	54	23

linear function. As the linear approximations at the beginning use the data from the first 1.5 months from Fig. 2, they overestimate the degradation indicator. In the second part ($i > 50$), the ADP was able to follow the underlying degradation indicator, whereas the Fixed Exponential approximation was no longer good enough. This is because Fixed Exponential approximation extrapolates the approximation from the first three months into the future, whereas Fig. 2 shows that the underlying degradation indicator started to increase in month four.

Table 4 compares the three data sets C1, C2, and C3 (rows 2–4 and 7–9) using the metric I from Eq. (15). The values show the difference in percentage from the best obtained value for each data set, thus zero means that the method performed the best. For short term prediction linear approximations are enough, but their performance decreases for medium-term prediction. This is due to the fact that the linear approximation does not follow the underlying degradation indicator over the whole data set, as indicated in Fig. 3, and is not able to predict correctly. In all cases, the worst results were obtained for mean value approximation which predicts the degradation indicator to be constant.

6.3. Turbine comparison

Fig. 12 shows the value of P^i as a function of the number of window i for both prediction windows $T_{\text{pred}} = 3.5$ days (Fig. 12a) and $T_{\text{pred}} = 7$ days (Fig. 12b) for the turbine data for five algorithms: ADP (red), Linear (light blue with crosses), LinearC (dotted magenta), Fixed Exponential (dark blue with circles) and Mean (dashed green).

All the algorithms presented in Fig. 12 give a high error at the beginning (window number smaller than five). This is a result of choosing the default $T_{\text{app}} = \mathcal{T}_{\text{app}} = 2$ weeks. It can be seen in Fig. 10 that the underlying degradation indicator is increasing in this period. Thus, extrapolating the approximating model into the future overestimates the value of the underlying degradation indicator, which is also confirmed in Figs. 10a and b where the red curves are above the black curve in week three. The best results in this period are obtained from the approximation with the Mean value showed with dashed green in Fig. 12 because it averages the expected degradation indicator over two weeks and does not overestimate future values. Nonetheless, ADP was able to adapt to the underlying degradation indicator, and as a result gave the best predictions from window six.

Furthermore, all the approaches in Fig. 12 based on moving window framework, i.e. ADP (red), Linear (light blue with crosses), and LinearC (dotted magenta) yield smaller values of P^i in windows 40–60, compared with Fixed Exponential (dark blue with circles) and Mean (dashed green). This is because the Mean algorithm estimates the future degradation indicator based data until week seven, while it starts to increase in week eight. Therefore, the Mean algorithm underestimates the future degradation indicator. The Fixed Exponential algorithm tries to fit one exponential model to the whole data set available to that point. Therefore, it is not able to take into account the increasing period in week eight. Using a moving window approach results in removing older data from the approximation window, and in consequence, the ADP, Linear, and LinearC algorithms are able to better follow the underlying degradation indicator.

Contrary to the results for the tuning data set depicted in Fig. 11, for the data set from Fig. 4, the linear approximation over a constant window, LinearC (pink dotted line in Fig. 12), has a similar performance as the ADP (red line). This is due to the fact that the prediction window for the turbine data set is shorter than for the compressor. Therefore, it is more likely that the extrapolated values would follow the underlying degradation indicator more closely over a shorter time period. This is also confirmed by comparing the performance of LinearC method in Fig. 12a and b. In both cases, the ADP (red) and the LinearC (magenta) are close, but for $T_{\text{pred}} = 7$ days, the value of P^i for LinearC is higher than the value for ADP, which means that linear approximation is well suited for shorter prediction windows.

The confirmation of the best results for the turbine data set is in rows five and 10 in Table 4.

7. Potential impact of the algorithm for power optimisation

Large turbomachines in oil and gas applications can require up to 80 MW and process up to 500 000 m^3 per hour [35]. The loss of efficiency due to fouling can vary from machine to machine, but it can reach up to 10% as indicated by Fentaye et al. [26]. The loss of efficiency is then translated into increased power consumption of up to 4.5% for large compressors [36]:

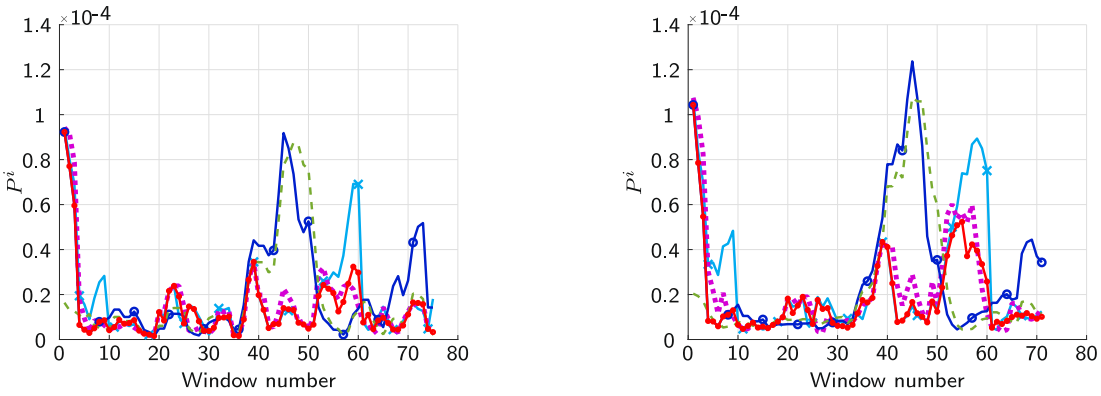


Fig. 12. Comparison of different approaches for turbine from Brekke et al. [28]. Legend: red with dots – ADP; light blue with crosses – Linear; dotted magenta – LinearC; dark blue with circles – Fixed Exponential; dashed green – Mean.

$$P_D = P + h(d) \tag{17}$$

where P_D is the power in degraded state, P is the power in up state, and $h(d)$ is a function translating the degradation indicator d to the effects it has on power. The ability to predict the degradation would enable improved operation of the machinery in two ways described in the subsequent sections.

7.1. Impact on performance-based maintenance

The expected value of degradation can be used for maintenance scheduling to provide the optimal schedule of washings and minimise the overall power consumption. Optimisation frameworks for maintenance scheduling with degradation modelling have been proposed by Hovland and Antoine [37], Nørstebø et al. [38], Xenos et al. [36], Schulze Spüntrup et al. [9], who all used linear models of degradation to minimise the power P_D in large oil and gas applications. Zulkafli and Kopanos [39] proposed a framework for maintenance scheduling for utilities, including compressor systems and modelled degradation as piecewise-linear between maintenance events. These frameworks would benefit from the more accurate model of degradation as provided by the new algorithm. In particular, the prediction would be more accurate and might extend the time between the maintenance activities such as compressor washings, or, conversely, accelerate an activity to prevent a failure.

7.2. Impact on decision support

The new algorithm may be used for decision support. In particular, the prediction bounds can provide an estimate of expected power increase over the prediction window $T_{\mathbf{pred}}$. Denoting the lower prediction bound as $d_{\mathbf{lower}}$ and the upper prediction bound as $d_{\mathbf{upper}}$, Eq. (17) yields:

$$P + h(d_{\mathbf{lower}}) < P_D < P + h(d_{\mathbf{upper}}) \tag{18}$$

Some industrial sites, in particular in off-shore settings, are not equipped with the infrastructure for online maintenance. The estimates obtained in Eq. (18) might help creating the decision-support environment for design changes, such as investing in a system for online washing. Knowing the prediction bounds of the power consumption will make it possible to assess the uncertainties in the payback time for the investment.

At the same time, the prediction intervals provided by the algorithm can be used for optimisation under uncertainty, as they indicate the value of probability that the data points will be within the interval. As an example, the optimisation framework for a gas transport network proposed by Cay et al. [40] can be extended to include uncertainty in degradation, not only in demand.

8. Discussion and conclusions

8.1. Synopsis

A data-driven algorithm for online prediction of turbomachinery degradation has been presented. The algorithm is based on time-trend analysis and combines a moving window approach with adaptive regression analysis and predicts the expected value of degradation and quantifies the uncertainty of the prediction. The adaptive switching between regression functions is also decided in a moving window fashion. The algorithm was used to predict the expected values of future degradation indicator and to quantify the uncertainty of the prediction by providing upper and lower prediction intervals in each window. An optimisation scheme was used for tuning of the parameters of the algorithm, such as the approximation and prediction window, and the threshold for change detection. The influence of the parameters was demonstrated and their impact on the accuracy of the prediction has been discussed. The adaptation properties were demonstrated in comparison with four window-based approaches: linear regression over a fixed and a varying approximation window, exponential regression, and analysis of mean value in a fixed window.

8.2. Discussion

The results using real data from industrial case studies show that the algorithm predicts the degradation indicator accurately in the short and medium terms. The moving window framework combined with nonlinear regression allows adaptation of the algorithm to the variations of the trend of the degradation indicator in between maintenance periods. A possible extension of the algorithm would be to adjust it for adaptation taking into account maintenance activities and the abrupt improvements denoted by E1, E2, E3, and ES in Fig. 1. This would allow a uniform analysis of degradation throughout extended periods of time.

The varying prediction intervals emphasize the uncertainty of the prediction. Thus, they provide additional information about the degradation indicator that can be used for scheduling under uncertainty. The predictions can be used as an input to decision-support systems such as production and maintenance scheduling systems. The benefit of using an online updated approximation function is that changes in the performance degradation are considered in the decision-making process of when to perform maintenance. As indicated in the turbine case study, the prediction intervals depend on the number of samples in the data set. Thus, increasing the sampling frequency when collecting the data would further improve the predictions and narrow down the prediction intervals.

The new algorithm assumes also that degradation data are available in real time. This indicates that both the efficiency in degraded state and the expected efficiency in up state have to be calculated online to use Eq. (1). Still, the new algorithm requires only one data set for tuning compared to more complex approaches based on machine learning and data science, such as neural networks. The new algorithm tackles in this way the challenges of insufficient data indicated by Fentaye et al. [26], who indicated that one of the challenges for most algorithms is the unavailability of data due to lack of sensors or their quality.

8.3. Conclusions

This study shows that it is possible to combine the existing approaches in degradation modelling to improve the accuracy of the prediction, thus making the algorithm useful in industrial performance-based application. Integrating the predictions results in industrial applications would lead to improved operations and energy savings by enabling performance-based maintenance.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Financial support is gratefully acknowledged from the Marie Curie Horizon 2020 EID-ITN project “PROcess NeTwork Optimization for efficient and sustainable operation of Europes process industries taking machinery condition and process performance into account PRONTO”, Grant agreement No. 675215.

The authors would also like to express their gratitude to Charlotte Skourup and Jan Wiik from ABB AS in Oslo for fruitful discussions and comments regarding the algorithm. Our thanks also go to Equinor ASA who kindly provided the data sets.

Appendix A. Variable detection and approximation window sizes

As indicated in Fig. 6 there are three possible paths for setting the sizes of the detection and the approximation window. By default, at the beginning the detection and prediction windows are the same with $T^i_{\text{app}} = T^{i+1}_{\text{app}}$ and $\mathcal{T}^i_{\text{Start}} = T^i_{\text{Start}}$, $\mathcal{T}^i_{\text{End}} = T^i_{\text{End}}$. This case represents the left path in Fig. 6, so $|\Delta y| \leqslant \widetilde{\theta}$. The i-th approximation window starts at T^i_{Start}, ends at T^i_{End} and lasts for T^i_{app}. The subsequent approximation window, $i+1$, is described with T^{i+1}_{Start} and T^{i+1}_{End}. The time difference between the ends of two consecutive windows, $T_{\text{f}} = T^i_{\text{End}} - T^{i+1}_{\text{End}}$, is constant, and characterises the update rate of the approximation. The dots on the time axis of Fig. 5 denote the measurement instants t^i_j in i-th approximation and detection windows.

If $|\Delta y| > \widetilde{\theta}$, the size of the approximation window and the detection window is derived as follows. The algorithm detects that the exponential model is suitable in i-th detection window, so $T^h_{\text{Start}} = \mathcal{T}^h_{\text{Start}}$ and $T^h_{\text{End}} = \mathcal{T}^h_{\text{End}}$, for $h \leqslant i$. The i-th approximation window is calculated as $T_{\text{app}^h} = T^h_{\text{End}} - T^h_{\text{Start}}$ and is equal to the detection window $\mathcal{T}_{\text{app}^h}$. Then the $i+1$ approximation window is shortened with $T^{i+1}_{\text{Start}} = T^i_{\text{Start}}$ that will remain fixed as long as the change is detected, and $T^{i+1}_{\text{End}} = T^i_{\text{End}} + T_f$. This corresponds to the middle path in Fig. 6.

Next, the algorithm moves to the right hand path in Fig. 6. If the shortened window is smaller than the default value T_{app}, the detection window will be the same as the approximation window, because $\min\{T_{\text{app}}, T^i_{\text{End}} - T^i_{\text{Start}}\} = T^i_{\text{End}} - T^i_{\text{Start}}$. The value of T^i_{Start} is then fixed as long as the change is detected $T^{i+j}_{\text{Start}} = T^i_{\text{Start}}$, whereas $T^{i+j}_{\text{End}} = T^i_{\text{End}} + jT_f$, where j is the number of detection windows where a change is detected. The fixed starting point and increasing endpoint of the approximation window indicate that the approximation window will expand as long as $|\Delta y| > \widetilde{\theta}$ is detected. The detection window, $\mathcal{T}^{i+j}_{\text{app}}$ is calculated from the formula $\min\{T_{\text{app}}, T^{i+j}_{\text{End}} - T^{i+j}_{\text{Start}}\}$. The detection window is then set as $\mathcal{T}^{i+j}_{\text{End}} = T^{i+j}_{\text{End}}$ and $T^{i+j}_{\text{Start}} = T^{i+j}_{\text{End}} - \min\{T_{\text{app}}, T^{i+j}_{\text{End}} - T^{i+j}_{\text{Start}}\}$. If the change is not detected any more, the approximation window and the detection window become the same again, $T^m_{\text{Start}} = \mathcal{T}^m_{\text{Start}}$ and $T^m_{\text{End}} = \mathcal{T}^m_{\text{End}}$ with $\mathcal{T}^m_{\text{app}} = T^m_{\text{app}} = T_{\text{app}}$.

The variable windows are depicted in Fig. A.13. The top axis shows the detection window, whereas the bottom one presents the approximation window. The exponential model is chosen in i-th detection window and applied in approximation windows until $i+j$ iteration. The prediction window is omitted for clarity as T_{pred} is constant.

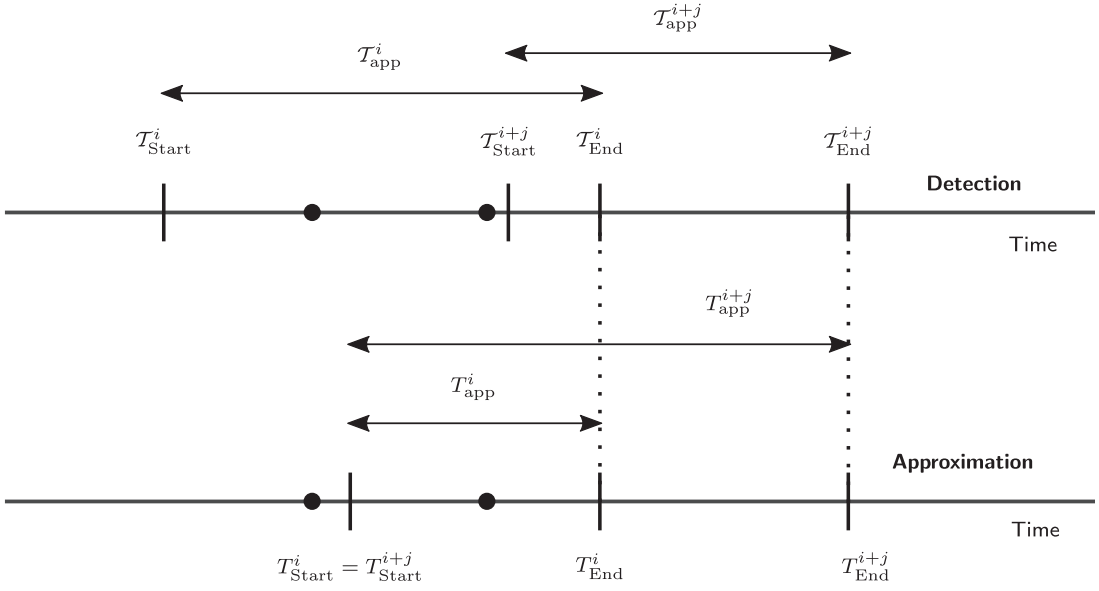


Fig. A.13. Variable detection and prediction windows for adaptation.

Appendix B. Formulas for calculation of prediction intervals

Table B.5 presents the formulas for calculating the prediction intervals for both the linear and the exponential model.

Table B.5
Parameters for calculating the prediction intervals for linear and exponential models.

Linear model	Exponential model
$\hat{a}_0 = \left[\frac{\partial f(t_0, b^i)}{\partial b_0^i} \quad \frac{\partial f(t_0, b^i)}{\partial b_1^i}\right]^\mathsf{T}$ $= [1 \quad t_0]^\mathsf{T}$	$\hat{a}_0 = \left[\frac{\partial f(t_0, b^i)}{\partial b_0^i} \quad \frac{\partial f(t_0, b^i)}{\partial b_2^i} \quad \frac{\partial f(t_0, b^i)}{\partial b_3^i}\right]^\mathsf{T}$ $= [1 \quad \exp(-b_3 t_0) \quad -t_0 b_2 \exp(-b_3 t_0)]^\mathsf{T}$
$\mathbf{A} = \left[\frac{\partial f(t_p, \beta^i)}{\partial b_0^i} \quad \frac{\partial f(t_p, b^i)}{\partial b_1^i}\right]_{p=1,\ldots,K^i}$ $= \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \\ 1 & t_K^i \end{bmatrix}$	$\mathbf{A} = \left[\frac{\partial f(t_p, \beta^i)}{\partial b_0^i} \quad \frac{\partial f(t_p, b^i)}{\partial b_2^i} \quad \frac{\partial f(t_p, b^i)}{\partial b_3^i}\right]_{p=1,\ldots,K^i}$ $= \begin{bmatrix} 1 & \exp(-b_3 t_1) & -t_1 b_2 \exp(-b_3 t_1) \\ 1 & \exp(-b_3 t_2) & -t_2 b_2 \exp(-b_3 t_2) \\ & \vdots & \\ 1 & \exp(-b_3 t_{K^i}) & -t_{K^i} b_2 \exp(-b_3 t_{K^i}) \end{bmatrix}$

Appendix C. Tuning of detection threshold

The next step is to find θ, i.e. to define how big a change is considered significant. The procedure to find θ is based on historical data and has two steps:

- Finding the minimal and maximal value of $\frac{\Delta y}{T_{\text{app}}}$ over the whole range of the tuning data set to find the range for θ
- Running an optimization procedure that finds a value of θ such that the prediction error is minimised

Fig. C.14 shows the values of $\frac{\Delta y}{T_{\text{app}}}$ over the whole period, for fixed $\mathcal{T}_{\text{app}}^i$ and T_{pred}. As $\mathcal{T}_{\text{app}}^i = T_{\text{app}}$ for all i, $\tilde{\theta} = \theta$ from Eq. (9). The minimal value of $\frac{\Delta y}{T_{\text{app}}}$ was denoted with a diamond, $\frac{\Delta y}{T_{\text{app}}} = 0.5 \times 10^{-4}$, and the maximal value with a filled circle, $\frac{\Delta y}{T_{\text{app}}} = 6.5 \times 10^{-4}$. This is equivalent to a change of between 0.02% and 2.7% of degradation indicator d during the approximation window T_{app}. As T_{app} was constant, $\Delta y = \theta$ and the value of θ is therefore sought in range [0.002, 0.027].

To find θ, an indicator measuring the mean difference $\overline{P}$ between the value calculated from the measurements, d_k^i, and the expected degradation $\hat{d}(t_k^i)$ in the i-th prediction window is used

$$\overline{P} = \frac{\sum_{i=1}^{N}\left(\sum_{k=1}^{K^i}\left(d(t_k^i) - \hat{d}(t_k^i)\right)^2\right)}{N} \tag{19}$$

where $t_k \in [T_{\text{End}}^i,\ T_{\text{End}}^i + T_{\text{pred}}]$ denotes the i-th prediction window with $k = 1, \ldots, K^i$ and $i = 1, \ldots, N$ with N denoting the number of windows for the whole tuning data set.

To show the influence of θ on prediction accuracy measured by the indicator (19), an exhaustive search was performed for $\theta \in [0.002, 0.027]$. As indicated with a square in Fig. C.15, the smallest value of $\overline{P}$ was obtained for $\theta = 0.005$ for $T_{\text{pred}} = 10$ days, and $\theta = 0.006$ for $T_{\text{pred}} = 30$ days. These values were also confirmed with Global Optimisation Toolbox in Matlab.

Fig. C.15 also shows that overly small and overly large values of θ would result in inaccurate predictions. If θ is too small, for example if $\theta < 0.003$ for the tuning data CT $|\Delta y| > \tilde{\theta}$ is detected more often. This forces the algorithm to use the exponential approximation for longer periods which

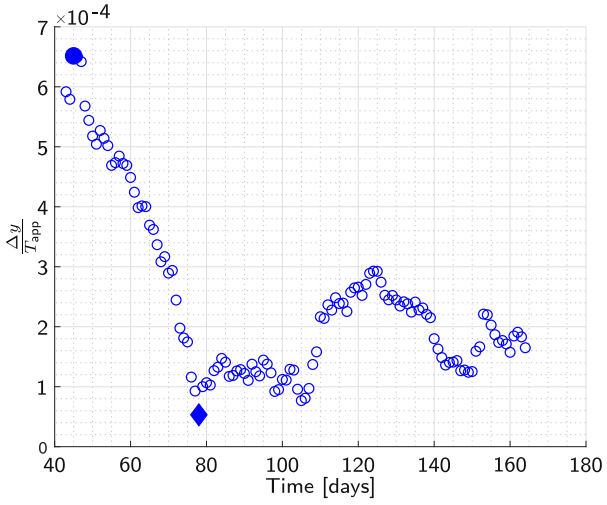
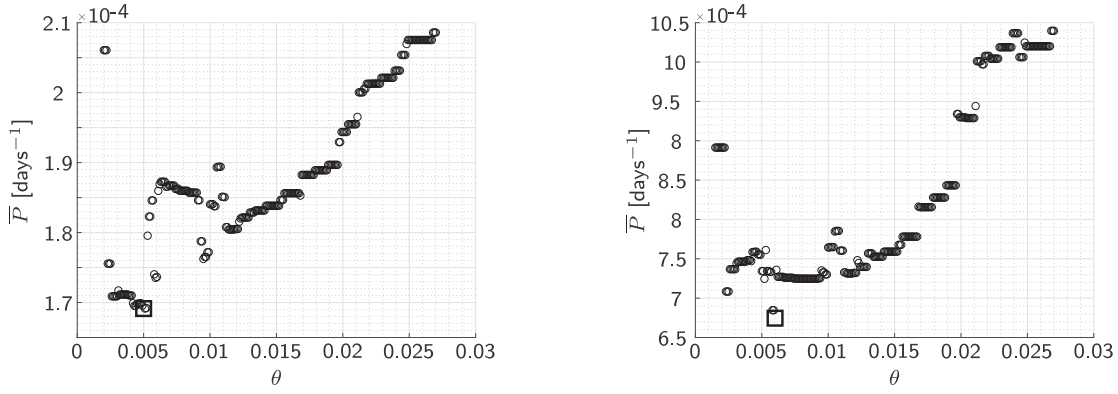


Fig. C.14. Minimal (diamond) and maximal (full circle) value of $\frac{\Delta y}{T_{\text{app}}}$ in i-th window.



(a) $T_{\text{pred}} = 10$ days (b) $T_{\text{pred}} = 30$ days

Fig. C.15. The indicator $\overline{P}$ for $\theta \in [0.002, 0.027]$.

results in overestimating the value of the degradation indicator as indicated in Fig. 3. At the same time, if θ is too large, $\theta > 0.011$, a linear regression would be used for prediction, as the $|\Delta y| \leqslant \widetilde{\theta}$. As presented in Fig. 3, linear approximation underestimates the degradation indicator.

References

[1] Jardine AKS, Lin D, Banjevic D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech Syst Sig Process 2006;20(7):1483–510.

[2] Tahan M, Tsoutsanis E, Muhammad M, Karim ZAA. Performance-based health monitoring, diagnostics and prognostics for condition-based maintenance of gas turbines: a review. Appl Energy 2017;198:122–44.

[3] BSI. BS EN 13306:2017. Maintenance – Maintenance terminology. Standard, British Standards Institution; 2017. https://bsol.bsigroup.com/.

[4] Hong Y, Meeker WQ, Escobar LA. Degradation models and analyses. In: Kotz S, Read CB, Balakrishnan N, Vidakovic B, Johnson NL, editors. Encyclopedia of statistical sciences. John Wiley & Sons Inc; 2011. p. 1–27.

[5] Verheyleweghen A, Jäschke J. Framework for combined diagnostics, prognostics and optimal operation of a subsea gas compression system. IFAC-PapersOnLine 2017;50(1):15916–21.

[6] Mahamad AK, Saon S, Hiyama T. Predicting remaining useful life of rotating machinery based artificial neural network. Comput Math Appl 2010;60(4):1078–87.

[7] Stalder JP. Gas turbine compressor washing state of the art: field experiences. J Eng Gas Turb Power 2000;123(2):363–70.

[8] Aretakis N, Roumeliotis I, Doumouras G, Mathioudakis K. Compressor washing economic analysis and optimization for power generation. Appl Energy 2012;95:77–86.

[9] Schulze Spüntrup F, Dalle Ave G, Harjunkoski I, Imsland L. Optimal maintenance scheduling for washing of compressors to increase operational efficiency. Kiss AA, Zondervan E, Lakerveld R, Özkan L, editors. 29th European symposium on computer aided process engineering, vol. 42. Elsevier; 2019. p. 1321–6.

[10] Puggina N, Venturini M. Development of a statistical methodology for gas turbine prognostics. J Eng Gas Turb Power-Trans ASME 2012;134(2):022401.

[11] Mo H, Sansavini G, Xie M. Performance-based maintenance of gas turbines for reliable control of degraded power systems. Mech Syst Sig Process 2018;103:398–412.

[12] Cicciotti M. Adaptive monitoring of health-state and performance of industrial centrifugal compressors Ph.D. thesis Imperial College London; 2015 http://hdl.handle.net/10044/1/51468 [accessed: 30 JUL 2019].

[13] Li YG, Nilkitsaranont P. Gas turbine performance prognostic for condition-based maintenance. Appl Energy Oct. 2009;86(10):2152–61.

[14] Tsoutsanis E, Meskin N, Benammar M, Khorasani K. A dynamic prognosis scheme for flexible operation of gas turbines. Appl Energy 2016;164:686–701.

[15] Hanachi H, Mechefske C, Liu J, Banerjee A, Chen Y. Performance-based gas turbine health monitoring, diagnostics, and prognostics: a survey. IEEE Trans Rel 2018;67(3):1340–63.

[16] Loboda I, Yepifanov S, Feldshteyn Y. A generalized fault classification for gas turbine diagnostics at steady states and transients. J Eng Gas Turb Power – Trans ASME 2007;129(4):977–85.

[17] Tarabrin AP, Schurovsky VA, Bodrov AI, Stalder J-P. An analysis of axial compressors fouling and a cleaning method of their blading. In: ASME 1996 international gas turbine and aeroengine congress and exhibition. vol. 1: Turbomachinery, ASME 1996 International Gas Turbine and Aeroengine Congress and Exhibition, Birmingham, UK, June 10–13, 1996; 1996. p. V001T01A093.

[18] Mokhatab S, Poe WA, Mak JY. Natural gas compression. In: Handbook of natural gas transmission and processing, 4th ed. Gulf professional publishing, Ch. 14; 2018. p. 441–4.

[19] Meeker WQ, Escobar LA. Statistical methods for reliability data. John Wiley & Sons Inc; 1998.

[20] Bagdonavičius V, Haghighi F, Nikulin M. Statistical analysis of general degradation path model and failure time data with multiple failure modes. Commun Stat – Theory Meth 2005;34(8):1771–91.

[21] Bagdonavičius V, Nikulin M. Statistical models to analyze failure, wear, fatigue, and degradation data with explanatory variables. Commun Stat – Theory Meth 2009;38(16–17):3031–47.

[22] Tsoutsanis E, Meskin N. Derivative-driven window-based regression method for gas turbine performance prognostics. Energy 2017;128:302–11.

[23] Hanachi H, Mechefske C, Liu J, Banerjee A, Chen Y. Enhancement of prognostic models for short-term degradation of gas turbines. 2017 IEEE international conference on prognostics and health management. 2017. p. 66–9.

[24] Kiakojoori S, Khorasani K. Dynamic neural networks for gas turbine engine degradation prediction, health monitoring and prognosis. Neural Comput Appl 2016;27(8):2157–92.

[25] Fentaye AD, Ul-Haq Gilani SI, Baheta AT, Li Y-G. Performance-based fault diagnosis of a gas turbine engine using an integrated support vector machine and artificial neural network method. Proc Inst Mech Eng, Part A: J Power Energy 2019;233(6):786–802.

[26] Fentaye AD, Baheta AT, Gilani SI, Kyprianidis KG. A review on gas turbine gas-path diagnostics: state-of-the-art methods, challenges and opportunities. Aerospace 2019;6(7):83.

[27] Cavarzere A, Venturini M. Application of forecasting methodologies to predict gas turbine behavior over time. J Eng Gas Turb Power-Trans ASME 2012;134:012401-1–8.

[28] Brekke O, Bakken LE, Syverud E. Compressor fouling in gas turbines offshore: Composition and sources from site data. In: ASME Turbo Expo 2009: Power for Land, Sea, and Air. Vol. Volume 5: Microturbines and Small Turbomachinery; Oil and Gas Applications. ASME; Int Gas Turbine Inst, 54th ASME Turbo Expo 2009, Orlando, FL, JUN 08–12, 2009; 2009. p. 381–91.

[29] Campbell JM, Maddox RN, Lilly LL. Gas conditioning and processing: the equipment modules vol. 2. Campbell: Petroleum Series; 1994.

[30] Meher-Homji CB, Chaker M, Motiwalla H. Gas turbine performance deterioration. Proceedings of the 30th turbomachinery symposium. 2001. p. 139–76.

[31] Syverud E. Axial compressor performance deterioration and recovery through online washing. Ph.D. thesis, NTNU; 2007. https://core.ac.uk/download/pdf/52098004.pdf [accessed: 4 JUL 2019].

[32] Rohatgi A. Webplotdigitizer, v. 4.1; 2018. https://automeris.io/WebPlotDigitizer [accessed: 10 AUG 2018].

[33] Zagorowska M, Ditlefsen A-M, Thornhill NF, Skourup C. Turbomachinery degradation monitoring using adaptive trend analysis. IFAC-PapersOnLine 2019; 52:679–84, 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2019.

[34] Montgomery DC, Runger GC. Simple linear regression and correlation. In: Applied statistics and probability for engineers. John Wiley & Sons, Ch. 11; 2014.

[35] Devold H. Oil and gas production handbook. ABB Oil and Gas; 2013.

[36] Xenos DP, Kopanos GM, Cicciotti M, Thornhill NF. Operational optimization of networks of compressors considering condition-based maintenance. Comput Chem Eng Jan. 2016;84:117–31.

[37] Hovland G, Antoine M. Scheduling of gas turbine compressor washing. Intell Automat Soft Comput 2006;12(1):63–73.

[38] Nørstebø VS, Bakken LE, Dahl HJ. Energy efficient operation of gas export systems. In: International petroleum technology conference. international petroleum technology conference. Society for Petroleum Industry; 2007.

[39] Zulkafli NI, Kopanos GM. Planning of production and utility systems under unit performance degradation and alternative resource-constrained cleaning policies. Appl Energy 2016;183:577–602.

[40] Cay P, Mancilla C, Storer RH, Zuluaga LF. Operational decisions for multi-period industrial gas pipeline networks under uncertainty. Optim Eng 2019;20(2):647–82.