Fusing model-based and data-driven prognostic methods for real-time model updating

Li, Tianzhi; Moradi, Morteza; Khan, Michel Gokan; Guarese, Renan; Kronqvist, Jan; Romero, Mario; Xiao, Ming; Wang, Xi Vincent

DOI
10.1016/j.ymssp.2025.113200

Publication date
2025

Document Version
Final published version

Published in
Mechanical Systems and Signal Processing

Citation (APA)
Li, T., Moradi, M., Khan, M. G., Guarese, R., Kronqvist, J., Romero, M., Xiao, M., & Wang, X. V. (2025). Fusing model-based and data-driven prognostic methods for real-time model updating. *Mechanical Systems and Signal Processing*, *238*, Article 113200. https://doi.org/10.1016/j.ymssp.2025.113200

Mechanical Systems and Signal Processing 238 (2025) 113200

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Full Length Article

Fusing model-based and data-driven prognostic methods for real-time model updating

Tianzhi Li [a,g,*], Morteza Moradi [f], Michel Gokan Khan [b,g], Renan Guarese [c,g], Jan Kronqvist [b], Mario Romero [e], Ming Xiao [d], Xi Vincent Wang [a]

[a] Department of Production Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
[b] Department of Mathematics, KTH Royal Institute of Technology, Stockholm, Sweden
[c] Department of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden
[d] Department of Information Science and Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
[e] Department of Science and Technology, Linköping University, Norrköping, Sweden
[f] Center of Excellence in Artificial Intelligence for Structures, Prognostics & Health Management, Aerospace Engineering Faculty, Delft University of Technology, Delft, the Netherlands
[g] Digital Futures, Stockholm, Sweden

ARTICLE INFO

Keywords:
Prognostics and health management
Model-based prognostics
Data-driven prognostics
Fusion
Mutual updating
Particle filter

ABSTRACT

Prognostic methods broadly fall into two categories—model-based and data-driven—both of which have shown effectiveness across a range of engineering applications. Model-based approaches require an explicit representation of the degradation process, defining failure as the point when the physical damage state exceeds a predetermined threshold. Data-driven methods, on the other hand, leverage sensor data to directly predict end-of-life (EOL) or related prognostic information. Although both approaches offer insights that could be complementary and potentially fused, most existing fusion methods either combine the outputs from multiple methods or adopt a data-driven method to assist the model-based method. To further enhance the prognostic performance, this study proposes a fusion-based prognostic approach in which the output of one method is actively used to update the model of the other through either the crossover operator or the likelihood function. The proposed approach is validated using both an aluminum fatigue dataset and the Prognostics and Health Management (PHM) 2010 cutter wear dataset, demonstrating improved prognostic accuracy compared to either method used independently.

1. Introduction

Degradation is an inherent phenomenon of any engineering system and structure, leading to the accumulation of damage over operational time. Once a critical level of damage is reached, structural integrity can no longer be guaranteed, which could lead to dangerous accidents. This necessitates the use of methods to predict the remaining useful life (RUL) of the structure [1], also known as prognostics, fault prognosis, and damage prognosis. Over the years, a wide range of prognostic methods have been proposed across different application scenarios, including rotating machinery, civil infrastructure, battery, and aerospace structures. Prognostic methods can be broadly classified as model-based or data-driven, depending on whether they rely on an explicit damage state or an implicit health indicator [2,3].

* Corresponding author at: Department of Production Engineering, KTH Royal Institute of Technology, Stockholm, Sweden.
E-mail address: tianzhil@kth.se (T. Li).

https://doi.org/10.1016/j.ymssp.2025.113200
Received 23 April 2025; Received in revised form 3 August 2025; Accepted 4 August 2025
Available online 14 August 2025

Model-based approaches require an explicit mathematical description of the system's degradation process to estimate its RUL. They often start by defining an explicit damage index: for metallic plates, it can be crack length [4–6] or crack shape [7]; for gears, pitting level or wear depth [8]; for bearings, vibration amplitude [3]; and for cutter flutes, wear level [9]. Failure is considered to happen when the damage index surpasses a critical threshold, at which point the structure or system can no longer function. It then becomes necessary to establish a degradation model—often referred to as a prognostic model—to project future damage indexes and states until the threshold is reached, at which the RUL can be calculated based upon. Degradation models are categorized into two main types: physics-based models [4,10,11] obtained by physics-based knowledge and equations on degradation; and data-driven models [5,12] developed using experimental degradation data.

To take the degradation uncertainties into account in the prognostic model, an updating strategy given online measurement is required, and it often includes developing a measurement model that links the damage state to sensor measurements and applying a state estimation algorithm to the state space model with the latest sensor data. For example, Chen and Yuan et al. [4,11] first applied a particle filter (PF) to estimate the degradation parameters and crack length in a metallic plate with multiple fatigue cracks. The process and measurement equations were derived from a crack propagation model and a measurement model relating crack length to guided waves. The proposed framework can be taken as an efficient metallic prognostic solution, enabling accurate future crack length forecasts and RUL prediction.

When an explicit degradation model cannot be established based on physical damage using either physical knowledge or available damage data, data-driven approaches can be an alternative. These approaches are developed using a machine learning model that extracts implicit health indicators (HIs) from extensive sensor data, such as acceleration [13,14], strain [15], guided waves [16], acoustic emission [17], or multiple sensor signals [18–20]. Typical data-driven models include Gaussian process regression (GPR) [15], non-homogeneous Poisson process [21], convolutional neural network (CNN) [14,20], recurrent neural network (RNN) [13], and transformer [19]. Then, the derived HI is used as an input into a data-driven prognostic model either to project future HI and estimate the time exceeding the predetermined threshold as EOL or to directly predict RUL.

In general, there is no universally best prognostic method; thus, fusion approaches, especially those that combine degradation information with data-driven algorithms, are often preferred. These methods can be broadly classified into three categories: embedding physical degradation information into a data-driven prognostic model [22–26], integrating prognostic results from model-based and data-driven methods [27], and fusing prognostic results for online model updating [28–30].

The first category [22–26]—embedding physics into a data-driven model—is preferred when limited degradation information is available, but extensive sensor data exists. Prognostic information is provided by a single data-driven model, where physical information can be incorporated into input preparation and/or model construction, improving prognostic performance. For example, Chao et al. [22] developed physics-based performance models of an aircraft engine, applied an unscented Kalman filter to infer unobservable model parameters, and fused the parameters and sensor data as input to a deep neural network. This data-driven prognostics model, augmented with physics-informed features, can yield more accurate prognostics for turbofan engines. Similar strategies have also been explored by Yin et al. [23], Cao et al. [24], Zhu et al. [25], and Wang et al. [26] in the application of rolling bearings. One can also refer to Li et al. [1], which includes a comprehensive summary of similar advanced methods.

The second category [27]—integrating different methods—is viable when sufficient physical knowledge and data are available to develop both model-based and data-driven methods. Each method produces a set of prognostic results, which are either simply averaged or combined through a weighting criterion. For example, targeting fatigue cracks in railway cables, Zang et al. [27] adopted two methods—one based on Paris's law and PF and another using a feedforward neural network—generating two sets of RUL prediction results and producing the final RUL prediction using a simple mixture. Although the idea behind it is quite similar to ensemble learning for data-driven methods [31,32] or fusion of multiple model-based methods [33], fusing outputs from different types of methods can still potentially produce more accurate predictions, as one single type of method may not fully capture the complex degradation process.

The last category [28–30]—integration and updating—builds upon the previous one. Since different methods can provide the same information, such as EOL, the parameters of a single model can be refined in real-time by comparing posterior distributions of EOL or other relevant information from different methods. This concept—referred to as prognostic information-aided model updating—is first introduced in battery capacity prognostics [29,30], where a battery degradation model and a data-driven observation predictor are built to describe the evolution of battery health state and to provide the future observations, respectively, and then the degradation model is updated by the current and future observations. Later, Li [28] proposed a similar fusion framework but for fatigue prognostics, where an explicit degradation model can be updated based on prognostic results from a data-driven model, resulting in more accurate fatigue crack prognostics under guided wave monitoring. However, the updating process in the above literature [28–30] is unidirectional, relying solely on data-driven results to update the degradation model, while the reverse strategy has not been explored. Therefore, this strategy may not always guarantee superior performance, especially when the data-driven model cannot fully capture the uncertainties arising from the degradation process.

Continuously updating a data-driven model with the latest sensor measurement represents a typical online unsupervised learning problem, which has been investigated to some extent, e.g., online transfer learning [34–36], online incremental learning [37], and inverse solution [38]. For learning purposes, these methods require extra EOL information. For example, information from previous testing units or previous prediction results can be used to build pseudo-HI labels, and then a learning scheme can be implemented with its data and labels. This offers an opportunity to develop a mutual updating scheme—an idea that, however, remains unexplored.

Targeting the fusion of model-based and data-driven prognostic approaches, this work proposes a fusion-based framework to achieve real-time mutual model updating. Three contributions arise from this work. First, we demonstrate that existing fusion schemes, including simple mixture [27] and unidirectional updating [28–30], cannot provide stable improvements over a single method. Then,

we propose a mutual updating scheme that can update the models in both methods in real-time and provide superior performance. Finally, we further demonstrate the generalizability of the proposed scheme under different fusion strategies, including the crossover operator and likelihood function. The proposed approach is validated using both the aluminum fatigue dataset and the PHM 2010 cutter wear dataset [39], demonstrating an enhanced prognostic performance compared to using either one of the methods by itself.

The rest of this paper is organized as follows: Section 2 introduces the model-based and data-driven methods, as well as the real-time fusion scheme. Sections 3 and 4 provide the results of the proposed method applied to the fatigue crack and PHM 2010 cutter wear datasets, respectively, outlining the superior performance of the proposed fusion framework. Finally, Section 5 concludes this paper.

2. Proposed framework

Fig. 1 illustrates the proposed three-step fusion method, which includes a model-based prognostic method, a data-driven prognostic method, and an online fusion scheme. The model-based approach often includes the construction of degradation and damage quantification models and the state and parameter estimation. The estimation results are then used for future projection and RUL calculation. The data-driven approach relies solely on one model linking sensor data and HI, and the RUL can be obtained by direct prediction or state estimation, depending on the specific model used. Finally, the proposed online fusion scheme includes updating the model from one method with EOL information from the other method.

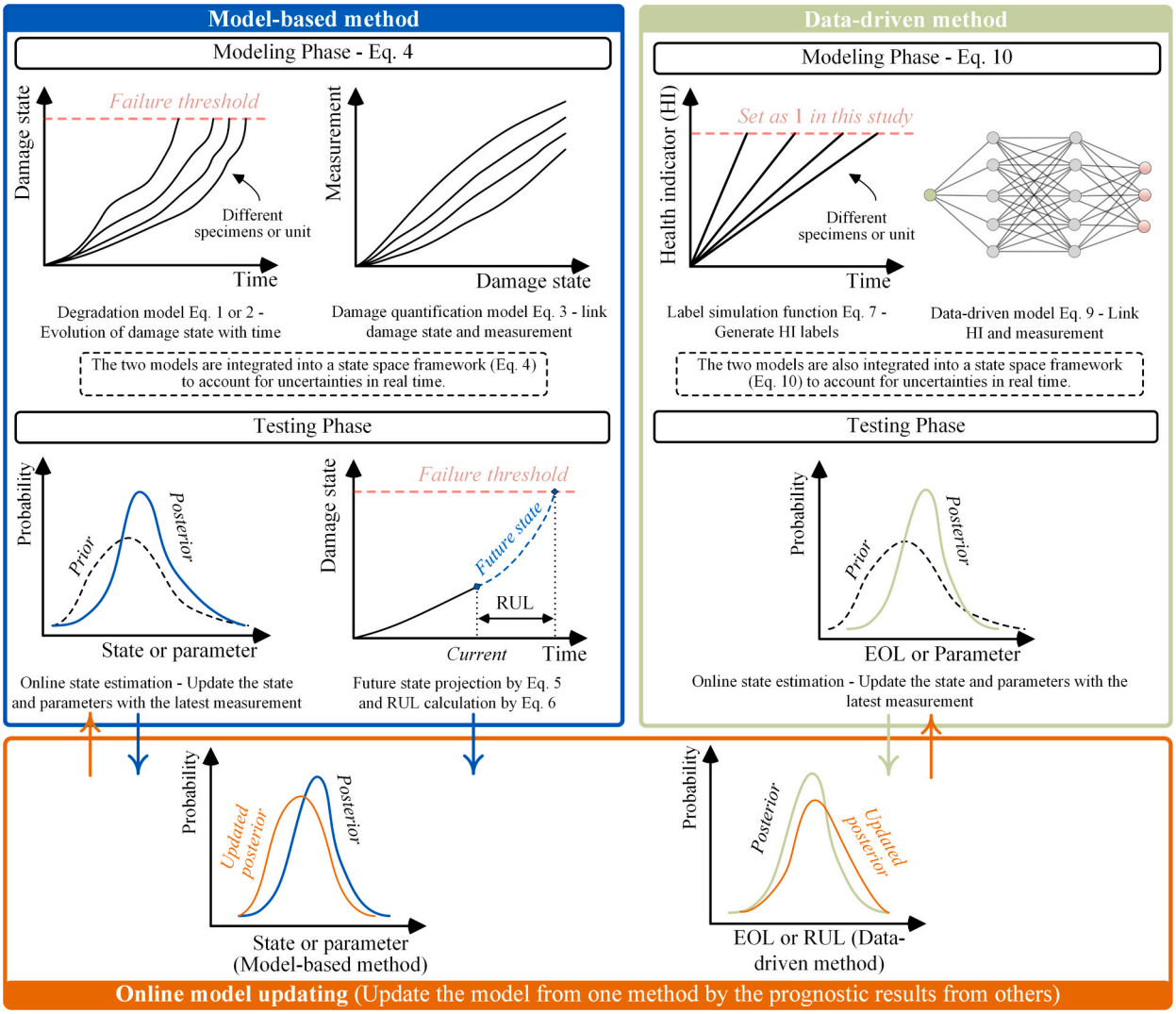

Fig. 1. Framework of online fusing model-based and data-driven methods.

2.1. Model-based prognostic method

Model-based prognostic methods have been studied in certain application scenarios, including metallic plates [28], bearings [3], and cutter flutes [9], where an explicit damage state—such as crack length in metals, vibration amplitude in bearings [3], or wear level in cutter flutes—can be used to evaluate the health status of a specimen or unit. The damage progressively accumulates over operational time as a result of degradation. Failure is defined as the point at which the damage reaches a critical threshold, where the unit cannot function anymore. These methods typically follow two phases, including modeling and testing.

First, a degradation model is always required, and it can be described as [3,4,28]

$$x_k = x_{k-1} + \Delta t_k f(x_{k-1}) \tag{1}$$

or

$$x_k = f(t_k) \tag{2}$$

where $f(\cdot)$ is the degradation function, x is the damage state, k denotes the k-th time step, t is the service time or a related parameter that can be interpreted as service time, and Δt is the time interval between two adjacent time steps.

The first model [4,28] reflects a discrete-time and recursive degradation process, where the evolution of damage state is modeled as a function of its previous state and the time interval. The formulation inherently assumes a Markovian property, where the future state primarily depends on the immediately preceding state—a widely adopted assumption in the field of prognostics [40]. The function $f(\cdot)$ can vary depending on the nature of the system. For physics-based models, $f(\cdot)$ is derived from first principles, such as crack propagation laws or material wear mechanisms, providing physical interpretability. For data-driven models, $f(\cdot)$ is typically obtained by statistically fitting historical degradation data, offering flexibility in scenarios where the degradation mechanisms are poorly understood.

On the other hand, the second model [3] considers the damage state as a function of time instant, without explicitly modeling the state transitions. Such a model is often suitable when abundant historical degradation data is available and the damage accumulation exhibits a stable and monotonic trend, and it is built by a data-driven modeling strategy that relies purely on experimental data. The two types of models will be adopted in this work, while only the first one will be used to explain the remaining part of this section.

Since the damage state is often unobservable during the degradation process—for example, invisible damage in composite structures or wear on cutter flutes—it is necessary to use sensors to obtain appropriate measurements, such as fiber Bragg grating strain sensors [41] or guided wave-based monitoring systems [28], to infer the extent of damage. As it is unlikely to link the damage state and measurements via physical knowledge, the sensor signal is often processed into one or more damage-sensitive statistical features, serving as inputs for data-driven models that describe the relationship between the state and features. This approach has been successfully implemented using a variety of models, including multilayer perceptron [41], polynomial functions [28], and Gaussian process regression [6]. Typically, the measurement equation can be expressed as

$$\boldsymbol{y}_k = g(x_k) \tag{3}$$

where $g(\cdot)$ is a data-driven function linking the damage state x and the measurement $\boldsymbol{y}$.

To account for the uncertainties within the degradation process, the parameters within the two functions are typically treated as unknown variables for estimation. A common strategy involves incorporating these parameters into the state vector and building the state space model [4,28]

$$\begin{cases} \begin{bmatrix} \theta_{f,k} \\ x_k \\ \theta_{g,k} \end{bmatrix} = \begin{bmatrix} \theta_{f,k-1} + \boldsymbol{\omega}_{1,k} \\ x_{k-1} + \Delta t_k f\left(x_{k-1}, \theta_{f,k}, \omega_{2,k}\right) \\ \theta_{g,k-1} + \boldsymbol{\omega}_{3,k} \end{bmatrix} \\ \boldsymbol{y}_k = g\left(x_k, \theta_{g,k}\right) + \boldsymbol{\nu}_k \end{cases} \tag{4}$$

where θ_f and θ_g stand for the parameters within the degradation and measurement models, respectively, and $\boldsymbol{\omega}_1$ and $\boldsymbol{\omega}_3$ are their process noises, ω_2 is the process noise for the damage state, and $\boldsymbol{\nu}$ is a vector of measurement noise.

The testing phase often includes online estimation with the latest sensor data, and future state projection and RUL prediction. This model is updated by the data collected up to the latest step through a state estimation technique, including Kalman filter or its extensions and particle filter (PF). PF is used in this study due to its demonstrated performance in nonlinear systems, and it is presented in Appendix A. One can refer to [42,43] for more detailed information.

Finally, within the PF-based framework, by resorting to the degradation model and the estimation results at k-th step, the prognostic stage facilitates the projection of the future damage state at q steps ahead, as follows

$$x_{k+q} = x_{k+q-1} + \Delta t_{k+q} f\left(x_{k+q-1}, \theta_{f,k}\right) \tag{5}$$

The RUL prediction generally requires the definition of a failure threshold—a damage level obtained from the industrial standards or the operational experiences [3]. The system is considered to fail when the future state reaches the threshold, so the RUL turns out to be equal to the time interval between the step k and the failure step as

$$EOL_{m,k} = t_{k+q} - t_k \tag{6}$$

where 'm' stands for 'model-based'.

In general, model-based methods can achieve accurate, generalizable, and interpretable prognostics, provided that three key requirements are met [1,28]: (i) an explicit damage state is available; (ii) a damage state-based degradation model can be developed based on failure mechanisms or experimental data; and (iii) the state and parameters of the model can be estimated using measurement data.

2.2. Data-driven prognostic method

Data-driven prognostic methods or RUL prediction methods often require a machine learning model to link the sensor data with the health indicator (HI), which takes all damage types and degradations into account comprehensively. It is important to note that this method relies on several key assumptions. First, the underlying health condition follows a monotonic trend over time as long as no maintenance or self-healing is happening. Second, the sensor data or extracted features contain sensitive information to infer the HI through a learnable function. Furthermore, the underlying degradation patterns across different samples or units are expected to be similar within the same operating environment, ensuring the applicability of the learned model. Therefore, data-driven methods are often applied to engineering systems that exhibit gradual degradation, where direct measurement of damage is difficult, but sensor data reflecting system health is available. Typical examples include rotating machinery [44], cutting tools [45], and composite plates [16]. This class of methods typically follows two phases, including modeling (or training) and testing.

For the modeling phase, as the true comprehensive HIs for any engineering system/structure are not available, the HI labels for training the machine learning model can be hypothetically generated by a function. This label simulator function can be characterized by both the current service time and EOL, such as a linear function, as the ratio of the current time to EOL [46]

$$s_k = \frac{t_k}{EOL} \tag{7}$$

where s denotes the HI. Note that each data collection process corresponds to one step. Different linear and nonlinear functions have been used as the simulator of HI labels, as elaborated in [14,16,17]. In this work, the above linear function is adopted to simulate HI labels, while the proposed method can also be adapted to nonlinear simulators.

Then, given the simulated labels, a data-driven function $h(\cdot)$ or $q(\cdot)$ can be built to link the sensor data and the HI as

$$s_k = q(\boldsymbol{u}_k) \tag{8}$$

or

$$\boldsymbol{u}_k = h(s_k) \tag{9}$$

where $\boldsymbol{u}$ can be the sensor signals or processed features. In order to provide an accurate performance evaluation between different methods, this study defines $\boldsymbol{u}$ as the above measurement $\boldsymbol{y}$.

The former model is a regression model, which can be built by a wide range of deep learning models, including CNN [16] or RNN [13]. During the testing phase, sensor data will be processed through this model to directly provide the HI and other prognostic information. In practical applications, the degradation patterns may vary across units or under different operating conditions, making the initially trained model prone to actual degradation over time. Therefore, to maintain accurate prognostics, the model requires an online updating scheme with the latest testing data. Since EOL cannot be obtained during the run-to-failure process, it is impossible to acquire the true label for each data stream in real time. Then, updating this model with the latest testing data represents an online unsupervised learning problem, which has been achieved within the framework of transfer learning [34–36] and incremental learning [37].

As to the latter, it can be embedded into an inverse prognostic solution [38], whose state space model can be described as

$$\begin{cases} \begin{bmatrix} EOL_k \\ s_k \\ \theta_{h,k} \end{bmatrix} = \begin{bmatrix} EOL_{k-1} + \omega_{4,k} \\ t_k/EOL_k + \omega_{5,k} \\ \theta_{h,k-1} + \boldsymbol{\omega}_{6,k} \end{bmatrix} \\ \boldsymbol{y}_k = h(s_k, \theta_{h,k}) + \boldsymbol{\nu}_k \end{cases} \tag{10}$$

where θ_h a vector of model parameters, ω_4, ω_5, and $\boldsymbol{\omega}_6$ are the process noises for the EOL, HI, and parameters, respectively. During the testing phase, this model will be processed by a state estimation algorithm like PF to produce the EOL or RUL distribution, the same as that of the model-based approach.

Both data-driven models can resort to the EOL information provided by model-based methods for learning or updating purposes, while this study opts for the latter, due to its flexibility to integrate information from model-based methods through the construction of a state space model as well as state and parameter estimation.

2.3. Real-time model updating

Since different methods can be designed to yield the same prognostic information, such as EOL, it is intuitive to refine a single model's state and parameters by comparing the posterior distributions of that information. However, the current investigation [28] updates only the degradation model in Eq. (4) using results from a data-driven method, leaving a mutual updating scheme unexplored. This gap may be since online learning or updating techniques for data-driven approaches have only recently begun to be investigated. In this context, we propose a mutual updating scheme that can be implemented at each time step, and then, explore if and how the proposed scheme can affect the prognostic performance.

For the data-driven method, the EOL distribution can be modified by the EOL results from the model-based method (or any other method if available). For example, one can adopt a crossover operator to modify its EOL as

$$EOL_k^{new} = (1-\alpha)EOL_k^{old} + \alpha EOL_{m,k} \tag{11}$$

where EOL_m is obtained by Eq. (6), 'old' and 'new' denote the samples before and after applying the fusion scheme, respectively, and α is a coefficient within a range of $[0, \quad 1]$. Note that the proposed scheme is generic enough so that the other strategies for refining the posterior distributions can be included. For more detailed information, one can refer to [47,48], although these specific strategies are not discussed here for the sake of simplicity.

For the model-based method, two strategies—one based on the crossover operator and another based on likelihood function—are adopted here in order to demonstrate the generalizability of the proposed method. The first one is taken from [28], and its implementation at the k-th time step is given as follows:

(i) Defining the 'accurate' EOL range based on the results from the data-driven method.
(ii) Detecting the model-based EOL samples that fall outside the predefined range and identifying their corresponding particles of damage state and parameters as 'inaccurate' samples.
(iii) Implementing a crossover operator for those 'inaccurate' samples by taking the damage state x_k as one example

$$x_k^{new} = (1-\beta)x_k^{old} + \beta x_k^{r} \tag{12}$$

where β is a coefficient within a range of $[0 \quad 1]$, 'old' and 'new' denote the inaccurate and modified samples, respectively, and 'r' represents samples randomly selected from all samples. The other methods for improving the posterior distributions can also be applied, as mentioned above, albeit for simplicity's sake, these particular methods are not covered here.

In terms of likelihood, Bayesian state estimation methods typically require the likelihood of the current state based on observations [4,28], without incorporating EOL information. To address this limitation, this study proposes a likelihood function that integrates both EOL and damage information into the likelihood function:

$$\log p_{new}(\boldsymbol{y}_k|\boldsymbol{z}_k) = (1-\gamma)\log p(\boldsymbol{y}_k|\boldsymbol{z}_k) + \gamma \log p\left(\boldsymbol{y}_k|EOL_{m,k}\right) \tag{13}$$

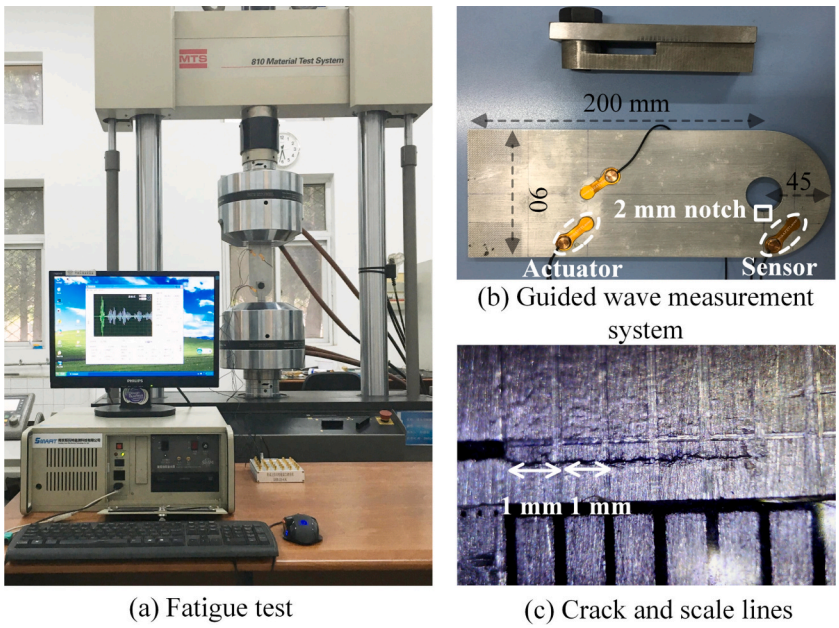

Fig. 2. Experimental setup of aluminum lug structure under fatigue loading monitored by guided waves.

where $\boldsymbol{z}_k$ is the state vector within Eq. (4), γ is a coefficient within a range of $[0 \quad 1]$, and $p(\boldsymbol{y}_k|\boldsymbol{z}_k)$ and $p(\boldsymbol{y}_k|EOL_{m,k})$ are the likelihood functions derived from Eqs. (4) and (10), respectively. Note that the model-based and data-driven methods each have their own measurement equations, resulting in separate likelihood functions. Since no unified probabilistic model has been developed to integrate the two sources, the fusion is carried out at the decision level by multiplying the two likelihoods, without considering the correlation between different model parameters. This design reflects a trade-off between modeling complexity and practical applicability. The latter—Eq. (10)—includes a vector of nuisance parameters θ_h, so the profile likelihood [49] is adopted to profile out the nuisance parameters:

$$\log p(\boldsymbol{y}_k|EOL_{m,k}) = \max_{\theta_{h,k}} \log p(\boldsymbol{y}_k|EOL_{m,k}, \theta_{h,k}) \tag{14}$$

3. Case study I: Aluminum fatigue crack dataset

This section validates the proposed method through the fatigue crack dataset. Section 3.1 explains the setup of fatigue testing and guided wave monitoring. Section 3.2 focuses on the extraction of statistical features from guided wave signals. Section 3.3 introduces state space modeling for model-based and data-driven methods. Lastly, Sections 3.4 and 3.5 present the results from the single-method approaches and the fusion-based approaches, respectively, using all experimental specimens.

3.1. Experimental study

Fig. 2 (a) and (b) show the fatigue test of an aluminum lug structure and a guided wave-based monitoring system [6]. The specimen has a thickness of 5mm and a 5mm-diameter hole. The MTS810 electro-hydraulic servo material test system is used to apply a sinusoidal tensile fatigue load to the structure. The maximum and minimum values and the frequency are 18kN, 1.8kN, and 10Hz, respectively. During fatigue loading, one transducer operates as the actuator, emitting a 3-cycle Hanning-windowed sine burst with a central frequency of 160kHz, while another one serves as the sensor, collecting guided waves with a sampling frequency of 50MHz. The above data collection is triggered every 50 and 70 load cycles during the initial and final stages of the test, respectively.

Fig. 2 (c) is the crack image captured by a digital microscope, where the scale lines are spaced at 1mm intervals, serving as a reference. The fatigue test starts and ends, when the crack lengths are 3mm and 22mm, respectively. The fatigue loading has been periodically paused for crack observation, and the crack growth data at each guided wave collection instant is fitted by the observed crack lengths. The fatigue test described above is repeated on four identical lug structures, designated T1, T2, T3, and T4.

Fig. 3 (a) displays the fatigue crack growth trajectories of four specimens. These trajectories exhibit variations, primarily stemming from factors such as manufacturing tolerances and assembly discrepancies. For all the specimens, the initial crack length and the threshold for RUL calculation are set as 3mm and 22mm, respectively. Fig. 3 (b) plots the guided waves at 3 selected load cycles for specimen T1. The first wave package is a crosstalk, arising from the electromagnetic induction between the actuator and sensor circuits [50]. The window for feature extraction is thus defined as 0.56×10^{-4}s to 2×10^{-4}s, a total of 7200 data points.

Cross-validation is performed for each specimen, as given in Table 1. One will be used for testing, while the other three will be utilized to extract the features and to build the model-based and data-driven prognostic models.

3.2. Feature extraction from guided waves

In each testing scenario, a convolutional neural network-based autoencoder is utilized for unsupervised feature extraction, as it has been demonstrated in handling signals such as vibration data [51,52] and guided waves [53]. The model architecture is shown in Fig. 4. The encoder takes the signal of length 7200 as input. Four convolutional blocks are stacked, each consisting of a one-

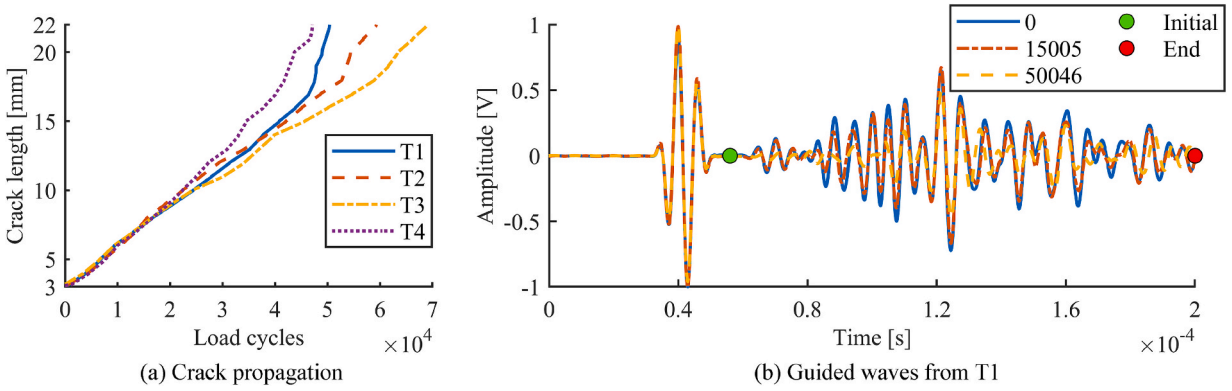

Fig. 3. Fatigue crack propagation data and guided wave signals. Note: (i) '0', '15005', and '50046' denote the signals collected at 0, 15005, and 50,046 load cycles, respectively; (ii) 'Initial' and 'End' represent the boundaries of the window used for feature extraction.

Table 1
Four testing scenarios for the fatigue crack dataset.

Scenario	Testing specimen	Training specimen (Feature extraction & Modeling)
1	T1	T2, T3, T4
2	T2	T1, T3, T4
3	T3	T1, T2, T4
4	T4	T1, T2, T3

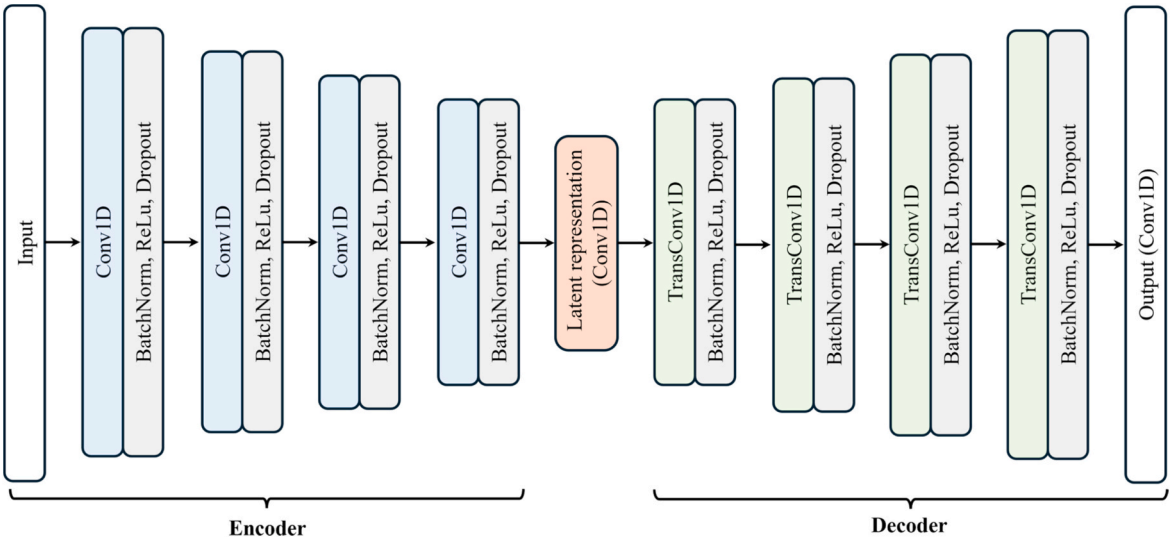

Fig. 4. Autoencoder for feature extraction from guided waves.

dimensional convolutional layer with 32 filters, a batch normalization layer, a ReLU activation, and a dropout layer (drop rate 0.1). The convolutional layers have kernel sizes of 30, 20, 10, and 6, and strides of 20, 15, 6, and 4, respectively. The latent (bottleneck) representation is a convolutional layer with eight filters, each corresponding to one variable. Note that the eight latent variables will be processed to provide eight features for building the prognostic models.

The decoder similarly contains four transposed convolutional blocks, each comprising a one-dimensional transposed convolutional layer with 32 filters, a batch normalization layer, a ReLU activation, and a dropout layer (drop rate 0.1). The transposed convolutional layers have kernel sizes of 6, 10, 20, and 30, with strides of 4, 6, 15, and 20, respectively, followed by a one-dimensional convolutional layer that reconstructs the feature map into a single output channel. For the sake of simplicity, detailed theoretical foundations and more advanced architectural designs are not elaborated here; readers interested in these aspects are referred to [54,55].

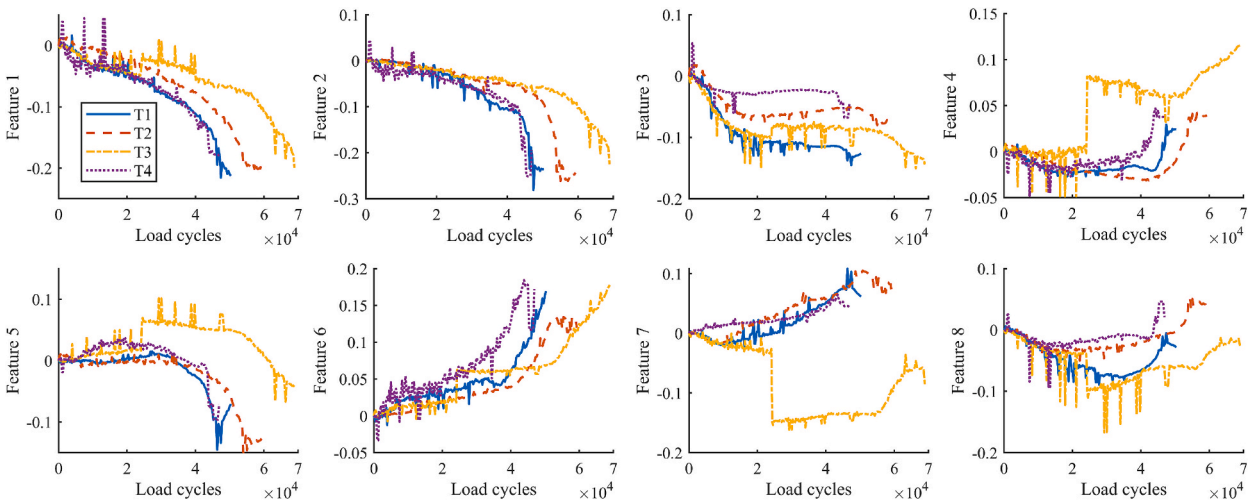

Fig. 5. Features extracted by autoencoder from Scenario 1 for the fatigue crack dataset.

The autoencoder is trained using the Adam optimizer under the following settings: a maximum of 200 epochs, a mini-batch size of 100, an initial learning rate of 0.01, a learning rate drop period of 10 epochs with a drop factor of 0.5, data shuffling at each epoch, and 70 % of the data for training and 30 % for validation to avoid overfitting. Validation is performed every 15 iterations with a patience of 8 validation checks.

Since the initial HI or crack length is identical across different specimens, the initial features should be set consistently. Specifically, each of the eight latent variables is adjusted by subtracting its initial value, setting the starting point to zero, resulting in eight features for this study. The features obtained under Scenario 1 are presented in Fig. 5, demonstrating satisfactory monotonicity. One can, however, note that the features 4, 5, 7, and 8 from specimen T3 significantly change at about the load cycle 2.4×10^4, which might be due to variations in the connection between the sensor and the specimen during testing.

3.3. State space modeling

The proposed method is flexible and can accommodate various prognostic models. However, for validation purposes, only one model-based method and one data-driven method are employed for each testing scenario. The former includes building a crack propagation model and a measurement model linking the crack length and features. The crack propagation model is taken from [33] and is given by

$$x_k = x_{k-1} + \left(c_{1,k}{x_{k-1}}^3 + c_{2,k}{x_{k-1}}^2 + c_{3,k}x_{k-1} + c_{4,k}\right)\Delta t_k \tag{15}$$

where x represents the model-based damage state, specifically crack length in this section, and c_1, c_2, c_3, and c_4 are polynomial coefficients, and Δt is the number of load cycles between two adjacent time steps. Then, the state space model for the model-based method can be built as

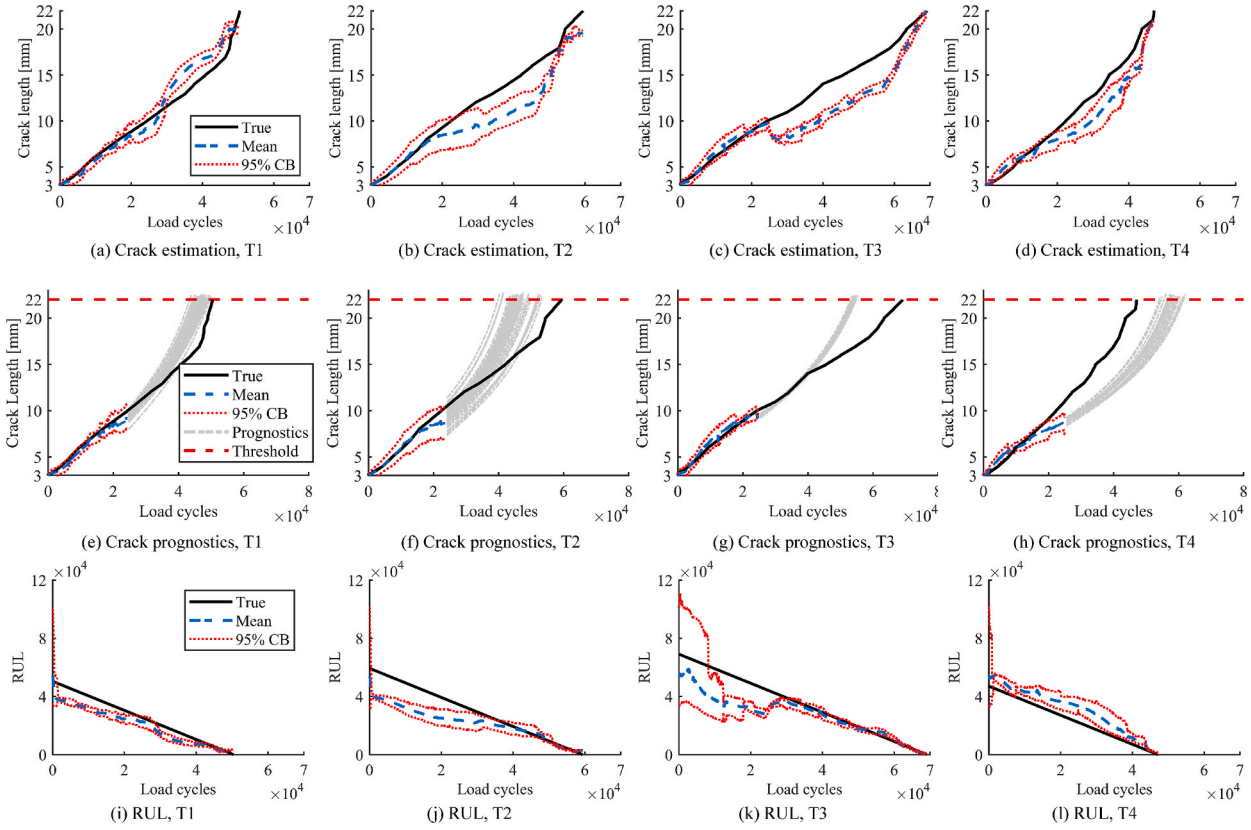

Fig. 6. Estimation and prognostic results by applying the model-based method to the fatigue crack dataset.

$$\begin{cases} \begin{bmatrix} c_{1,k} \\ c_{2,k} \\ c_{3,k} \\ c_{4,k} \\ x_k \\ \boldsymbol{b}_k \end{bmatrix} = \begin{bmatrix} c_{1,k-1} + \omega_{c,1,k} \\ c_{2,k-1} + \omega_{c,2,k} \\ c_{3,k-1} + \omega_{c,3,k} \\ c_{4,k-1} + \omega_{c,4,k} \\ x_{k-1} + \left(c_{1,k} {x_{k-1}}^3 + c_{2,k} {x_{k-1}}^2 + c_{3,k} x_{k-1} + c_{4,k}\right)\Delta t_k + \omega_{2,k} \\ \boldsymbol{b}_{k-1} + \boldsymbol{\omega}_{3,k} \end{bmatrix} \\ \boldsymbol{y}_k = g(x_k) + \boldsymbol{b}_k + \boldsymbol{\nu}_k \end{cases} \tag{16}$$

where $\omega_{c,1}$, $\omega_{c,2}$, $\omega_{c,3}$, $\omega_{c,4}$, ω_2, and $\boldsymbol{\omega}_3$ are zero-mean Gaussian process noises, the measurement $\boldsymbol{y}$ contains the eight extracted features, the function $g(\cdot)$ is built by a multilayer perceptron (MLP) containing two eight-node hidden layers, given that MLP can approximate complex nonlinear relationships between the damage state and the features while maintaining a relatively simple architecture that facilitates real-time implementation, and a vector of bias parameters $\boldsymbol{b}$ is used to account for the uncertainties arising from the MLP, a strategy taken from [56].

The data-driven model [38] is formulated as

$$\begin{cases} \begin{bmatrix} EOL_k \\ s_k \\ \boldsymbol{b}_k \end{bmatrix} = \begin{bmatrix} EOL_{k-1} + \omega_{4,k} \\ t_k / EOL_k + \omega_{5,k} \\ \boldsymbol{b}_{k-1} + \omega_{6,k} \end{bmatrix} \\ \boldsymbol{y}_k = h(s_k) + \boldsymbol{b}_k + \boldsymbol{\nu}_k \end{cases} \tag{17}$$

where ω_4, ω_5, and $\boldsymbol{\omega}_6$ are zero-mean Gaussian process noises, the measurement $\boldsymbol{y}$ contains the eight extracted features, t is interpreted as the number of load cycles, $h(\cdot)$ is also a multilayer MLP with two eight-node hidden layers, and a vector of bias parameters $\boldsymbol{b}$ accounts for the uncertainties, as described earlier. Each model is implemented with the three routines: model-based method, data-driven method, and proposed method. Note that the algorithm hyperparameters for all methods are the same, and they are given in Appendix Table A.2.

3.4. Results from individual methods

For the model-based method, one PF is applied to one prognostic model to provide one group of estimates of the crack length and degradation parameters, which are then adopted for projecting the future state and RUL. The true crack lengths are taken from Fig. 3 (a). The crack length estimation results are plotted in Fig. 6 (a) – (d). The samples of crack length from each model have the average values remaining close to the true values, and their confidence boundaries (CBs) shrinking with the load cycle step. The estimation results at each load cycle step can be used to project the future crack length and then calculate the RUL. For illustration purposes, the length prognostic results based on the estimation results at about 2.4×10^4 load cycles are plotted in Fig. 6 (e) – (h). In general, each group of future states is distributed around the true crack lengths with a certain level of deviation. This is also observed in Fig. 6 (i) – (l), where the average RUL is close to the target RUL with a certain bias, and the confidence boundaries shrink with the increasing load cycle steps. On the other hand, the data-driven method provides less satisfactory RUL predictions, as given in Fig. 7.

For the model-based method, the prognostic performances are quantified by five metrics, i.e., the root-mean-square error (RMSE) and mean absolute percentage error (MAPE) of crack length estimation, the RMSE, MAPE, and prognostic horizon (PH) for RUL prediction. For the data-driven method and simple mixture [27], the prognostic performances are quantified by three metrics, i.e., the RMSE, MAPE, and prognostic horizon (PH) for RUL prediction. Note PH is usually taken as the difference between the load cycle step of the latest measurement and the failure threshold, provided the RUL prediction at that step meets a pre-defined specification [57]. Following the strategy proposed in [58], this study considers the PH as the number of load cycles when 60 % of the RUL distribution first falls within a range of 'true RUL $\pm$ 10%N'.

Table 2 presents the metrics, including the RMSE, MAPE, and ratio of pH to EOL, for all four specimens. Lower RMSE and MAPE

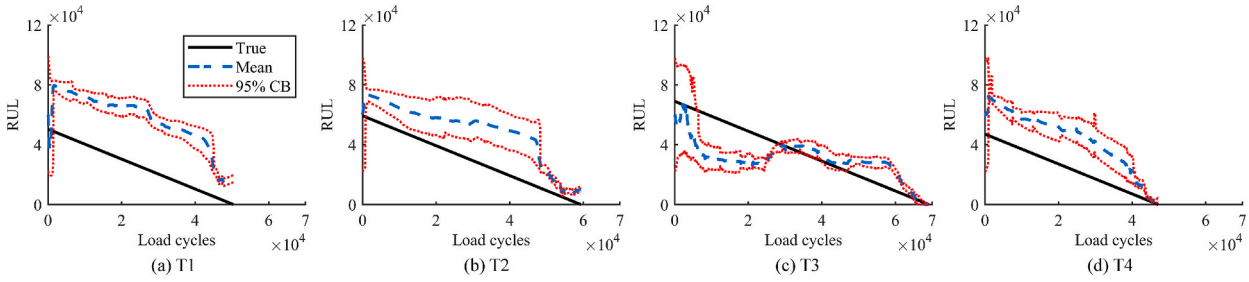

Fig. 7. RUL prediction results by applying the data-driven method to the fatigue crack dataset.

Table 2
Performance metrics of model-based and data-driven methods for the fatigue crack dataset.

	Crack length (Model-based method)		RUL (Model-based method)			RUL (Data-driven method)			RUL (Mixture)		
	RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	1.23	8.53	6.57	35.28	0.58	32.53	253.78	N	13.56	123.63	N
T2	2.08	11.94	11.68	36.29	0.48	20.71	128.59	0.11	7.79	60.43	0.11
T3	2.26	13.83	11.88	19.17	0.61	15.59	40.81	0.61	13.47	28.12	0.61
T4	1.70	12.61	8.06	45.86	0.87	21.75	111.40	0.07	14.84	78.63	0.07

Note: 'N' means that the PH criterion is not satisfied.

represent better performances, while the ratio of pH to EOL is within the range of [0, 1], and a larger ratio denotes a better performance. First, it can be observed that the model-based method generally achieves better performance across the specimens. This is attributed to the incorporation of physical knowledge related to fatigue crack propagation, which effectively constrains the estimation and projection processes. Second, the simple mixture approach [27], which directly combines the prognostic outputs of the data-driven and model-based methods, does not lead to an improvement over the model-based method. For example, most of the RMSEs and MAPEs lie within those from the model-based data-driven methods, and the ratios of pH to EOL are identical to those from the data-driven method. The above phenomena highlight the need for a more systematic fusion scheme to achieve advanced performance.

3.5. Results from unidirectional model updating

To evaluate the effects of each scheme on prognostic performance, two additional routines are separately implemented for each specimen:

- Fusion 1: The samples of damage state and degradation parameters from the model-based method are modified by data-driven results through either the crossover operator or likelihood, and the RUL prediction results are from the model-based method only.
- Fusion 2: The EOL samples from the data-driven method are modified by model-based results, and the RUL prediction results are from the data-driven method only.

Table 3 lists the performance metrics from Fusion 1 and Fusion 2. First, some results from Fusion 1 exhibit improvement compared to those from the single model-based method, while some do not. The main reason is that the data-driven EOL distribution is inaccurate; as such, its contribution to the fusion scheme is limited. Second, all results from Fusion 2 are noted as more accurate than those from the single data-driven method, showing that the data-driven model can be successfully updated by including additional EOL knowledge. As a consequence, by taking advantage of fusing two methods, the proposed method often yields the best performance.

Table 3
Results from unidirectional model updating for the fatigue crack dataset.

		Crack length (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
		RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	Fusion 1 (Crossover)	1.00	9.88 ↓	3.45	32.96	0.98	–	–	–
	Fusion 1 (Likelihood)	2.77 ↓	21.40 ↓	12.55 ↓	113.61 ↓	0.84 ↓	–	–	–
	Fusion 2	–	–	–	–	–	5.27	23.88	0.72
T2	Fusion 1 (Crossover)	1.36	5.13	14.19 ↓	49.06 ↓	0.15 ↓	–	–	–
	Fusion 1 (Likelihood)	2.97 ↓	16.87 ↓	11.62	93.83 ↓	0.59	–	–	–
	Fusion 2	–	–	–	–	–	9.25	28.28	0.94
T3	Fusion 1 (Crossover)	1.29	8.58	14.67 ↓	37.31 ↓	0.27 ↓	–	–	–
	Fusion 1 (Likelihood)	4.14 ↓	25.14 ↓	11.74	33.55 ↓	0.59 ↓	–	–	–
	Fusion 2	–	–	–	–	–	12.24	28.45	0.61
T4	Fusion 1 (Crossover)	1.21	9.05	5.66	39.20	0.87	–	–	–
	Fusion 1 (Likelihood)	2.28 ↓	16.42 ↓	15.68 ↓	96.52 ↓	0.06 ↓	–	–	–
	Fusion 2	–	–	–	–	–	7.94	52.74	0.89

Note: The symbol '↓' denotes 'performance of certain fusion method worse than one single method'.

3.6. Results from new methods

To provide a comprehensive evaluation, two fusion routines, one based on a crossover operator and one based on likelihood, are separately implemented for each specimen. The former is taken as one reference example, and the RUL prediction results from model-based and data-driven schemes are separately shown in Fig. 8. First, the results of the two schemes show a certain level of similarity, which means the prognostic information from one scheme is successfully delivered to another scheme for model updating. Second, the proposed method exhibits an obvious improvement compared with the use of a single method.

For the model-based method, it has been demonstrated in [28–30] that incorporating information from an additional data-driven prognostic model—referred to as prognostic information-aided model updating— can enhance estimation accuracy and overall prognostic performance. However, this improvement largely depends on the quality of the data-driven model: if it is inaccurate, the contribution to the model-based method will be limited, as already discussed in Section 3.5. Conversely, prognostic information derived from other sources, such as prior knowledge or physics-based models, can significantly enhance the performance of data-driven method—not only as validated in [22,37,38], but also as observed in Section 3.5. Therefore, when prognostic information from the model-based approach is integrated into the data-driven approach, the latter becomes more accurate, which, in turn, can further improve the performance of the model-based scheme. As a consequence, both methods can benefit from each other's strengths at each time step, enabling a mutual updating process that enhances overall prognostic accuracy over time.

Table 4 lists the performance metrics from the crossover operator-based fusion method. The similarity between the results of the two schemes is even more evident. Moreover, only one out of thirty-two performance metrics is lower than that of a single method, demonstrating the superior performance of the proposed method. Table 5 lists the performance metrics from the likelihood-based fusion method. The above conclusions are still applicable, demonstrating superior performance under different fusion strategies.

3.7. Analyses of fusion parameters

The sections above set the two fusion parameters α and β as 0.01. To evaluate the effects of fusion parameters on prognostic performance, the crossover operator-based fusion methods with four additional sets are separately implemented for each specimen:

- Fusion parameters α and β are set as 0.005,
- Fusion parameters α and β are set as 0.05,
- Fusion parameters α and β are set as 0.1,
- Fusion parameters α and β are set as 0.2.

Table 6 presents performance metrics obtained under various fusion parameters. The variation in these results clearly shows that fusion settings significantly impact prognostic accuracy. Moreover, the optimal fusion parameters differ across metrics for the same specimen. For example, for the specimen T1, the fusion parameters having the lowest RMSE and MAPE of crack length are 0.1 and 0.05, respectively. Notably, the proposed method with each set of fusion parameters often delivers better performance than relying on a single method. Thus, while fusion parameters do affect prognostic accuracy, the proposed fusion approach consistently outperforms any individual method in this study.

The above sections consider the fusion scheme implemented at each time step. To assess the impact of fusion timing on prognostic

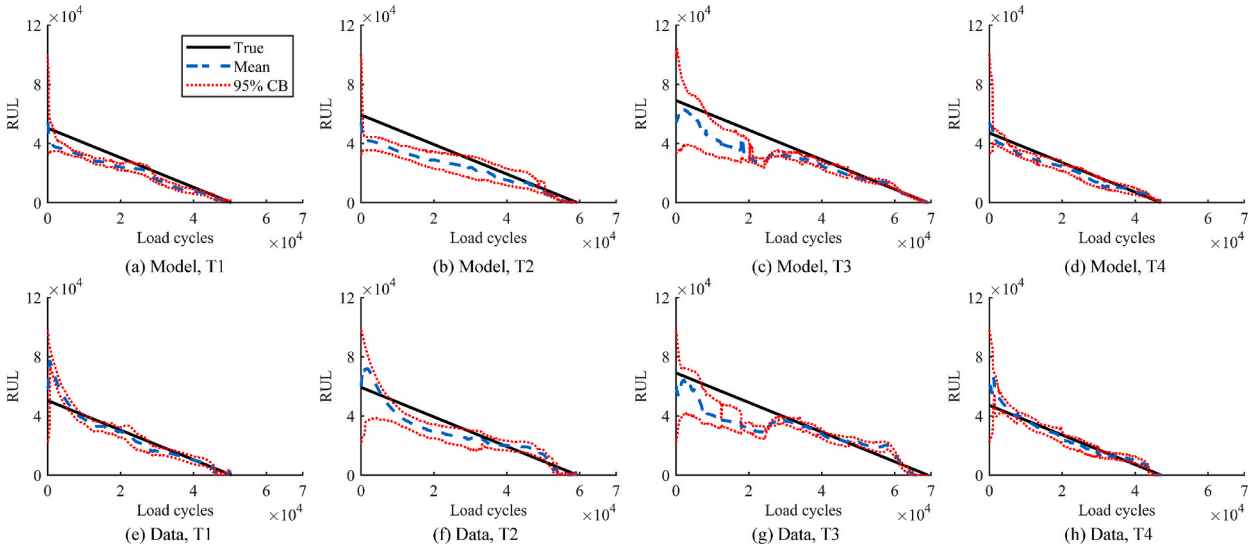

Fig. 8. RUL prediction results by applying crossover operator-based fusion method for the fatigue crack dataset.

Table 4
Performance metrics of crossover operator-based fusion methods for the fatigue crack dataset.

	Crack length (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
	RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	1.11	6.87	6.49	28.03	0.58	6.01	22.78	0.89
T2	1.36	9.17	9.67	27.33	0.46 ↓	7.01	25.10	0.92
T3	2.21	13.00	10.04	19.13	0.74	10.60	25.60	0.98
T4	0.69	6.94	3.86	21.09	0.98	3.78	16.37	0.91

Note: The symbol '↓' denotes 'performance of proposed method worse than one single method'.

Table 5
Performance metrics of likelihood-based fusion methods for the fatigue crack dataset.

	Crack length (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
	RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	0.84	7.27	3.74	29.18	0.97	6.26	34.75	0.80
T2	1.63	10.14	9.14	25.03	0.52	7.18	27.37	0.93
T3	2.25	13.44	11.48	27.97 ↓	0.61	9.74	23.79	0.62
T4	0.60	5.69	4.93	22.56	0.98	5.25	19.81	0.86

Note: The symbol '↓' denotes 'performance of proposed method worse than one single method'.

Table 6
Effect of fusion parameters on crossover operator-based fusion method for the fatigue crack dataset.

Parameter		Crack length (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
		RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	0.005	0.93	6.67	4.64	20.50	0.66	11.02	51.04	0.44
	0.01	1.11	6.87	6.49	28.03	0.58	6.01	22.78	0.89
	0.05	0.77	**6.01** *	3.97	17.19	0.71	3.95	19.56	0.97
	0.1	**0.67** *	6.10	3.21	21.12	0.90	3.14	20.65	**0.99** *
	0.2	1.02	6.95	**1.38** *	**8.58** *	**0.98** *	**1.38** *	**8.84** *	**0.99** *
T2	0.005	1.39	8.91	10.08	28.55	0.44 ↓	6.51	**21.64** *	0.85
	0.01	1.36	9.17	9.67	27.33	0.46 ↓	7.01	25.10	0.92
	0.05	1.24	7.86	11.15	26.82	**0.55** *	10.61	27.93	0.56
	0.1	**0.89** *	**5.86** *	10.67	30.59	0.35 ↓	10.44	32.13	**0.99** *
	0.2	1.30	8.51	**6.38** *	**22.92** *	0.53	**6.29** *	23.16	0.54
T3	0.005	2.16	13.10	12.57 ↓	28.89 ↓	0.44 ↓	10.68	25.15	**0.98** *
	0.01	2.21	13.00	10.04	19.13	0.74	10.60	25.60	**0.98** *
	0.05	**1.91** *	**11.40** *	11.88	25.34 ↓	0.51 ↓	11.41	27.03	0.51 ↓
	0.1	2.50 ↓	14.30 ↓	**9.15** *	33.46 ↓	**0.97** *	**9.18** *	26.63	0.63
	0.2	2.40 ↓	13.76	10.09	**17.06** *	0.61	10.05	**17.67** *	0.61
T4	0.005	1.15	10.85	5.24	35.03	0.71 ↓	4.86	30.00	0.85
	0.01	0.69	**6.94** *	3.86	**21.09**	**0.98** *	3.78	16.37	0.91
	0.05	**0.66** *	7.43	5.77	25.03	0.96	5.40	25.14	**0.98** *
	0.1	1.27	9.86	5.25	57.57 ↓	**0.98** *	5.48	52.14	0.96
	0.2	0.86	8.35	**2.07** *	**15.71** *	**0.98** *	**2.18** *	**10.21** *	**0.98** *

Note: The symbols '*' and '↓' denote 'best performance among the fusion parameters' and 'performance of certain fusion method worse than one single method', respectively.

performance, crossover operator-based fusion is also applied at intervals of every 5, 25, and 100 time steps for each specimen. Table 7 presents the performance metrics under different fusion intervals. The results clearly indicate that fusion timing significantly affects prognostic accuracy. In general, though longer fusion intervals can occasionally provide the best performance, shorter intervals, such as every step and every five steps, yield more stable improvement. As previously discussed in Sections 3.5 and 3.6 and supported by [22,28–30,37,38], accurate prognostic information from a data-driven model can improve the model updating for the model-based method, and reliable model-based information can refine the data-driven method. Thus, more frequent fusion facilitates mutual enhancement between the two approaches, leading to superior overall performance.

Table 7
Effect of fusion timing on likelihood-based fusion method for the fatigue crack dataset.

Parameter		Crack length (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
		RMSE [mm]	MAPE	RMSE $[\times 10^3]$	MAPE	PH / EOL	RMSE $[\times 10^3]$	MAPE	PH / EOL
T1	1	1.11	**6.87** *	6.49	28.03	0.58	**6.01** *	**22.78** *	**0.89** *
	5	0.91	7.59	**2.65** *	**17.89** *	**0.99** *	16.2	98.67	0.09
	25	**0.86** *	8.44	5.62	39.96 ↓	0.97	27.23	159.36	0.08
	100	1.26 ↓	8.63 ↓	6.84 ↓	31.23	0.97	36.87 ↓	452.70 ↓	N
T2	1	**1.36** *	9.17	9.67	27.33	0.46 ↓	**7.01** *	**25.10** *	**0.92** *
	5	1.44	**9.12** *	9.38	27.76	0.47 ↓	9.36	75.97	0.66
	25	2.02	11.99 ↓	**8.99** *	39.65 ↓	**0.52** *	26.77 ↓	168.01 ↓	0.09 ↓
	100	1.75	10.47	9.18	**25.57** *	0.51	30.63 ↓	158.60 ↓	0.09 ↓
T3	1	**2.21** *	**13.00** *	**10.04** *	19.13	0.74	**10.60** *	**25.60** *	0.98
	5	2.25	13.97 ↓	11.43	**18.76** *	0.62	11.70	30.33	0.62
	25	2.79 ↓	17.57 ↓	11.78	22.56 ↓	0.62	15.45	42.63 ↓	**0.99** *
	100	3.68 ↓	21.55 ↓	24.31 ↓	117.4 ↓	**0.82** *	17.15 ↓	52.15 ↓	**0.99** *
T4	1	**0.69** *	**6.94** *	3.86	**21.09** *	**0.98** *	**3.78** *	**16.37** *	**0.91** *
	5	0.79	8.42	4.11	28.67	0.79 ↓	10.33	67.4	0.07
	25	0.92	10.94	10.46 ↓	35.76	0.28 ↓	19.39	118.1 ↓	0.07
	100	1.50	11.99	**3.04** *	24.31	**0.98** *	8.82	59.48	0.84

Note: The symbols '*' and '↓' denote 'best performance among the fusion parameters' and 'performance of certain fusion method worse than one single method', respectively.

4. Case study II: Cutter wear dataset

This section validates the proposed method through the cutter wear dataset. Section 4.1 provides the experimental setup for cutter wear tests. Section 4.2 focuses on the extraction of statistical features from multiple-sensor signals. Section 4.3 outlines state space modeling for model-based and data-driven methods. Lastly, Section 4.4 presents the prognostic results of the five above routines using all cutters.

4.1. Experimental setup

The PHM 2010 cutter dataset [39] has been considered a benchmark dataset for tool wear analysis and used in lots of prognostic studies. The entire dataset contains six cutters, i.e., from C1 to C6. Each cutter has been tested under the same operating conditions. The spindle speed of the cutter is set at 10,400 RPM, with a feed rate of 1555mm/min. The radial and axial cutting depths are set as 0.125mm and 0.2mm, respectively.

Each cutter has three flutes that experience wear during service time, while only the wear levels of the three cutters, i.e., C1, C4, and C6, are recorded. Therefore, only the three cutters are used in this study. For each cutter, the wear levels of three flutes are recorded for 315 cuts, and the lifetime is defined as the time when the average flute wear reaches 0.16mm, following a similar strategy recommended by the PHM 2010 data challenge. Therefore, the three cutters have lifetimes of 306, 278, and 238 cuts, respectively. Fig. 9 plots the lifetime-long wear values for each cutter, exhibiting a similar pattern and enabling the construction of a wear-based degradation model.

At each cut, the sensor data is collected at the sampling frequency of 50kHz, and includes three-direction force, three-direction vibration, and the root-mean-square (RMS) of the acoustic emission signals. For each data sequence, the first four seconds of data are used for feature extraction, a strategy taken from [38]. Cross-validation is performed for each cutter. Specifically, one cutter will be

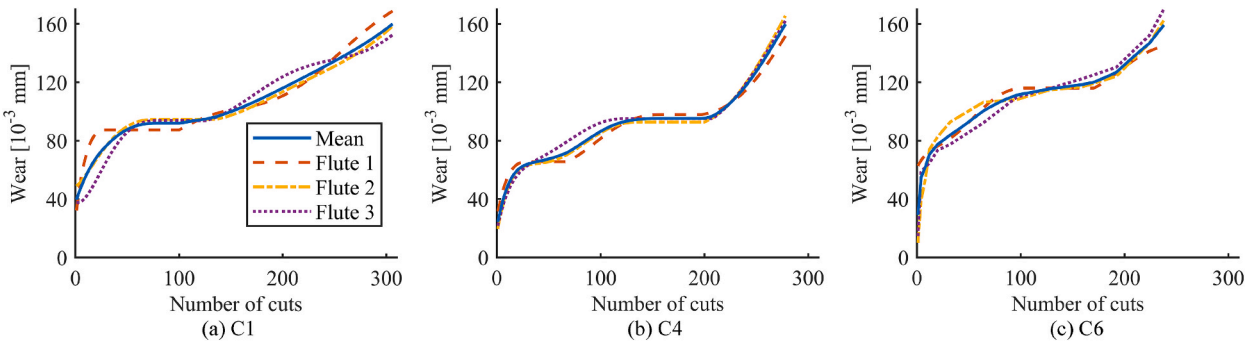

Fig. 9. Wear measurements of three flutes for three cutters C1, C4, and C6.

used for testing, while the other two will be utilized for modeling, as given in Table 8.

4.2. Feature extraction from multiple sensors

In order to demonstrate the proposed method under different feature extraction methods, the supervised feature extraction strategy [38] is used here. A two-dimensional (2D) CNN is adopted for feature extraction for the seven-channel sensor signals, each with a length of 200000. Three convolutional blocks are stacked in sequence. Each of the first two blocks consists of a 2D convolutional layer with 8 filters, a batch normalization layer, a ReLU activation, a max-pooling layer, and a dropout layer (drop rate 0.1). Their convolutional kernel sizes are $[1 \times 500]$ and $[1 \times 100]$, with strides of $[1 \times 50]$ and $[1 \times 20]$, respectively. The third block includes a 2D convolutional layer with 8 filters, followed by batch normalization and a ReLU activation, using a kernel size of $[1 \times 20]$ and a stride of $[1 \times 4]$. After these convolutional blocks, two fully connected layers are introduced, with 32 and 4 nodes, each having a sigmoid activation. Finally, a fully connected layer with a single output neuron produces the regression result, i.e., the HI. Interested readers may refer to [59] for more detailed theoretical insights and alternative practical architectures.

The Adam optimizer for 2D-CNN training has the same settings as in Section 3.2, except that a mini-batch size of 40 is used here. Validation is also performed every 15 iterations with a patience of 8 validation checks. The four nodes from the last hidden layer are considered features, as shown in Fig. 10, which will be used for further model construction.

4.3. State space modeling

The wear propagation function is taken from Eq. (2). It is built by a polynomial function as

$$x_k = l_{1,k}t_k^{\,3} + l_{2,k}t_k^{\,2} + l_{3,k}t_k + l_{4,k} \tag{18}$$

where t is interpreted as the number of cuts, x represents the physical damage state, specifically the average wear level in this section, l_1, l_2, l_3, and l_4 are polynomial coefficients. The state space model is similar to the one used in Section 3.3, as follows

$$\begin{cases} \begin{bmatrix} l_{1,k} \\ l_{2,k} \\ l_{3,k} \\ l_{4,k} \\ x_k \\ \boldsymbol{b}_k \end{bmatrix} = \begin{bmatrix} l_{1,k-1} + \omega_{l,1,k} \\ l_{2,k-1} + \omega_{l,2,k} \\ l_{3,k-1} + \omega_{l,3,k} \\ l_{4,k-1} + \omega_{l,4,k} \\ l_{1,k}t_k^{\,3} + l_{2,k}t_k^{\,2} + l_{3,k}t_k + l_{4,k} + \omega_{2,k} \\ \boldsymbol{b}_{k-1} + \boldsymbol{\omega}_{3,k} \end{bmatrix} \\ \boldsymbol{y}_k = g(x_k) + \boldsymbol{b}_k + \boldsymbol{\nu}_k \end{cases} \tag{19}$$

where $\omega_{l,1}$, $\omega_{l,2}$, $\omega_{l,3}$, $\omega_{l,4}$, ω_2, and $\boldsymbol{\omega}_3$ are zero-mean Gaussian process noises, the measurement $\boldsymbol{y}$ contains the four extracted features, and $g(\cdot)$ is a multilayer perceptron (MLP) containing two six-node hidden layers.

The construction of a data-driven model is the same as Eq. (17), while the measurement $\boldsymbol{y}$ contains the four extracted features, and $h(\cdot)$ is a multilayer perceptron (MLP) containing two six-node hidden layers. The hyperparameters used for the cutter dataset are listed in Appendix Table A.3.

4.4. Results from all methods

The performance of the proposed method is evaluated under five separate routines:

- Model: Model-based method.
- Data: Data-driven method.
- Fusion 1: The samples of damage state and degradation parameters from the model-based method are modified by data-driven results through either the crossover operator or likelihood, and the RUL prediction results are from the model-based method only.
- Fusion 2: The EOL samples from the data-driven method are modified by model-based results, and the RUL prediction results are from the data-driven method only.
- New method: Either the crossover operator or the likelihood.

Table 8
Three testing scenarios for the cutter wear dataset.

Scenario	Testing specimen	Training specimen (Feature extraction & Modeling)
1	C1	C4, C6
2	C4	C1, C6
3	C6	C1, C4

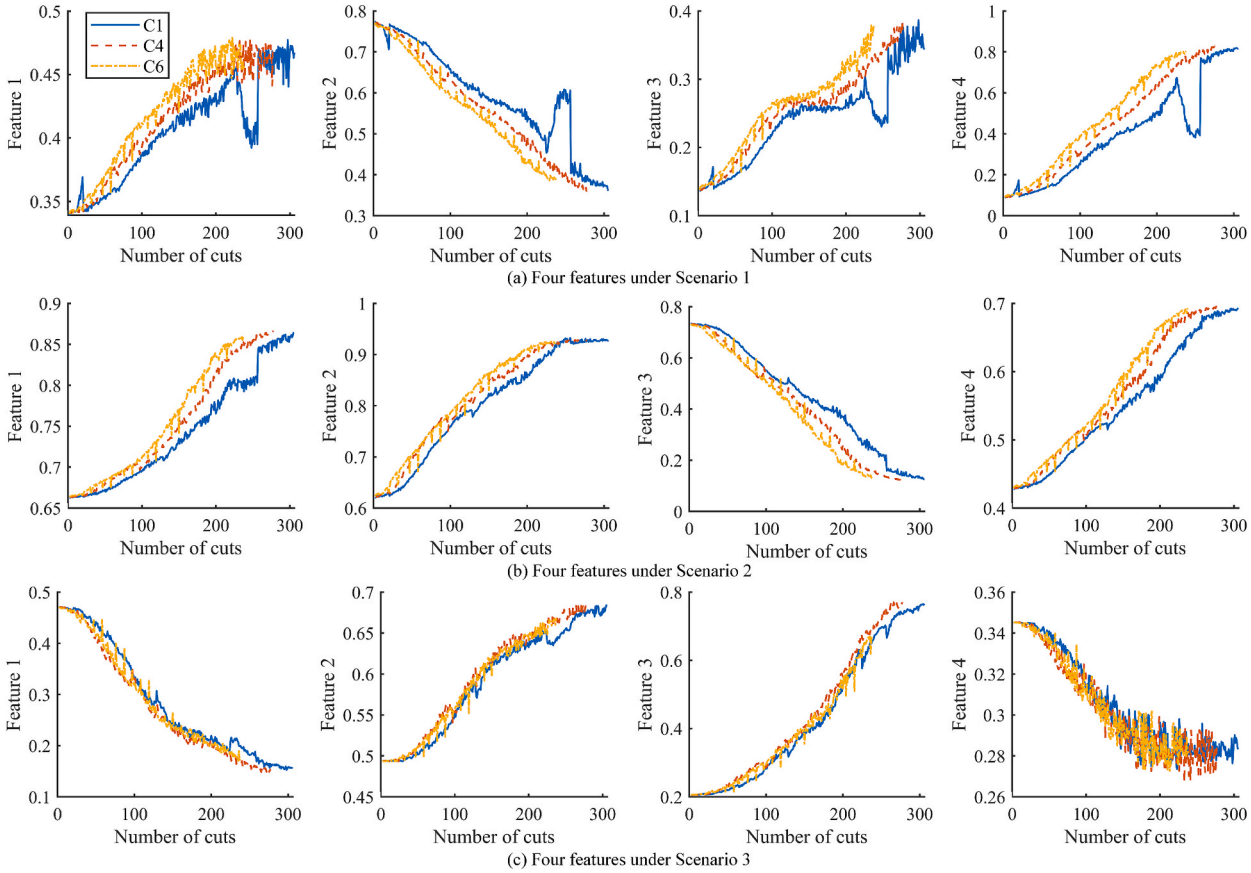

Fig. 10. Four features extracted from the last hidden layer of 2D-CNN for all three scenarios for the cutter dataset.

For simplicity, only the prognostic results from the crossover operator-based fusion method are plotted in Fig. 11, showing satisfactory performance. The performance metrics from the routines above are compared in Table 9. Previous conclusions are still applicable here and thus not repeated. In summary, the new method can provide more robust performance by fusing model-based and data-driven prognostic methods.

5. Conclusions

This work has proposed a real-time fusion scheme, where either the model-based or data-driven prognostic method is online, being improved by the results from the other method. Based on the results from both the aluminum fatigue and the PHM 2010 cutter wear datasets, four observations can be drawn here. First, the model-based prognostic method generally provides better performance over the data-driven method due to the use of physical knowledge. Second, the fusion scheme with simple mixture or unidirectional updating cannot provide stable improvements over the model-based method. Third, the proposed mutual updating scheme has been proven to provide superior performance. Finally, we further demonstrate the generalizability of the proposed scheme under different fusion strategies, including the crossover operator and likelihood.

Several limitations arise from this study, including the mutual updating mechanisms, the experimental validation, and the assumptions of information availability and the Markov process. Regarding the mutual updating scheme, the proposed fusion scheme exclusively incorporates single prognostic information, namely EOL, while the utilization of other prognostic outputs, such as predicted future states, degradation trajectories, or even future measurements, remains unexplored. These additional sources of information may carry insights beyond EOL, potentially leading to more efficient model updating. Moreover, while this study has demonstrated the proposed method via likelihood and crossover-based fusion, it has not introduced novel strategies for determining optimal fusion weight or fusion timing. These fusion strategies are directly applied without theoretical enhancement or adaptive mechanisms. Future research will focus on developing adaptive or dynamic fusion strategies that can automatically adjust fusion weights and trigger fusion events based on model uncertainty, data quality, or degradation stage. Finally, the proposed fusion strategy primarily involves information transmission and parameter updating between model-based and data-driven components. However, the deeper complementary nature of the two approaches has not yet been fully explored. Model-based methods offer clear interpretability and strong physical consistency, while data-driven approaches can efficiently capture complex and nonlinear degradation

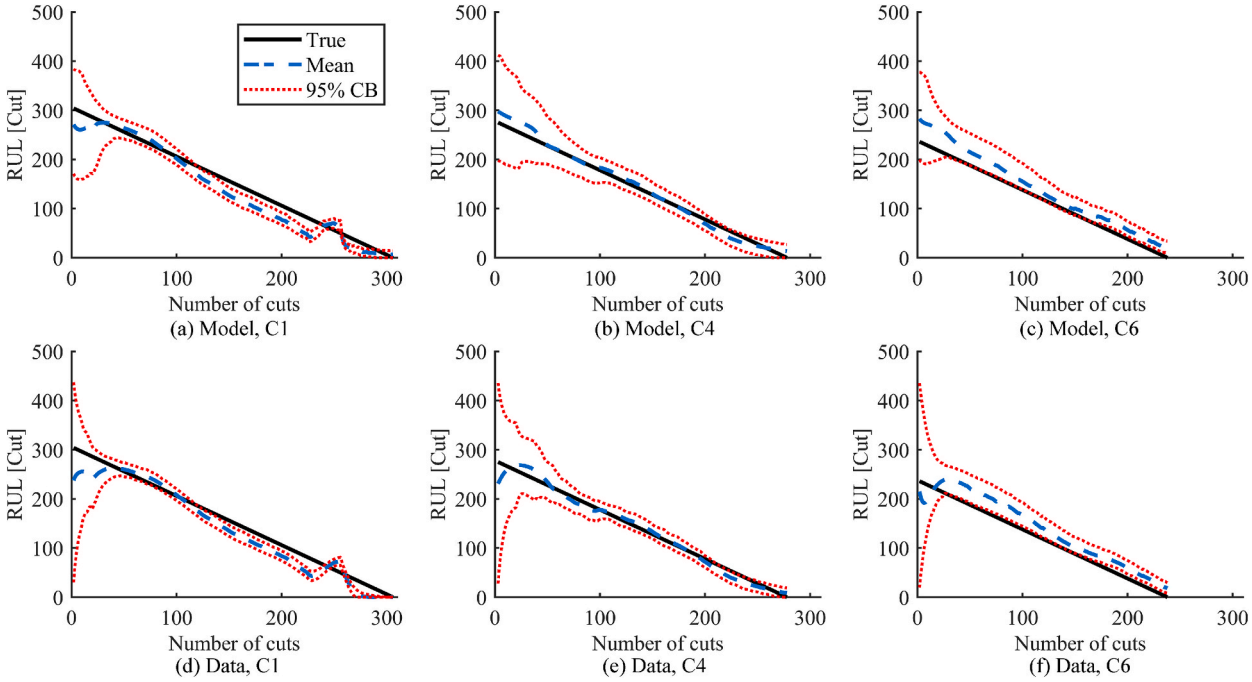

Fig. 11. RUL prediction results by applying crossover operator-based method to the cutter wear dataset.

Table 9
Performance metrics of single and fusion methods for the cutter wear dataset.

		Wear level (Model-based scheme)		RUL (Model-based scheme)			RUL (Data-driven scheme)		
		RMSE $[10^{-3}\text{mm}]$	MAPE	RMSE	MAPE	PH / EOL	RMSE	MAPE	PH / EOL
C1	Model	14.82	11.14	62.51	36.79	0.69	–	–	–
	Fusion 1 (Crossover)	14.51	11.09	34.53	26.77	0.67 ↓	–	–	–
	Fusion 1 (Likelihood)	14.01	10.24	42.11	32.92	0.69	–	–	–
	Data	–	–	–	–	–	69.03	45.67	0.27
	Fusion 2	–	–	–	–	–	56.93	35.94	0.92
	New (Crossover)	13.75	10.85	20.86	21.70	0.92	21.02	23.87	0.91
	New (Likelihood)	13.79	10.16	35.13	29.89	0.88	32.31	32.27	0.92
C4	Model	21.22	22.02	20.30	32.44	0.45	–	–	–
	Fusion 1 (Crossover)	21.58 ↓	22.34 ↓	11.41	26.47	0.72	–	–	–
	Fusion 1 (Likelihood)	21.18	22.00	19.55	30.03	0.47	–	–	–
	Data	–	–	–	–	–	24.57	23.95	0.62
	Fusion 2	–	–	–	–	–	14.16	22.85	0.83
	New (Crossover)	21.60 ↓	22.45 ↓	10.16	19.33	0.76	9.63	14.44	0.85
	New (Likelihood)	21.23 ↓	22.02	18.81	29.60	0.47	13.69	21.25	0.83
C6	Model	21.10	18.58	57.70	74.35	0.12	–	–	–
	Fusion 1 (Crossover)	21.76 ↓	19.16 ↓	55.52	86.53 ↓	0.42	–	–	–
	Fusion 1 (Likelihood)	20.00	17.66	52.12	58.37	0.53	–	–	–
	Data	–	–	–	–	–	59.20	149.86	N
	Fusion 2	–	–	–	–	–	56.64	83.82	0.93
	New (Crossover)	19.71	17.40	24.34	49.69	0.67	25.11	50.59	0.93
	New (Likelihood)	20.59	18.16	55.25	66.04	0.52	55.25	71.94	0.93

Note: (i) 'N' means that the PH criterion is not satisfied; (ii) The symbol '↓' denotes 'performance of certain fusion method worse than one single method'.

patterns. Future research will focus on establishing more sophisticated interaction mechanisms between the two paradigms, enabling them to promote each other, such as embedding physics-based degradation models such as Paris' law into the loss function of data-driven models to enforce physical constraints, or leveraging the data-driven residuals between predicted and observed prognostic results to identify systematic biases in the physics model and update its parameters accordingly.

As to the experimental validation, although a cross-validation strategy based on different specimens has been adopted, this approach may not fully eliminate the influence of specimen-specific characteristics or potential data correlation. Future work will explore more advanced grouping and validation strategies to rigorously assess model generalization under specimen variability.

Moreover, the current experiments do not cover variations in working conditions, materials, or structural configurations. Future studies will focus on applying the proposed framework to a wider range of datasets and application scenarios to assess its robustness and generalizability. This includes testing under different operating conditions, using datasets from different materials, and extending the approach to more complex or realistic structural systems. Finally, we acknowledge that deeper sources of uncertainty—such as parameter estimation error and observation noise—are not explicitly included. These aspects are important in real-world scenarios where both models and data are not ideal. Systematic investigations should be dedicated to these uncertainties by, for example, perturbing model parameters, broadening initialization ranges, or introducing artificial noise. More advanced uncertainty quantification techniques—such as Monte Carlo dropout, variational inference, deep ensembles for the data-driven scheme, and adaptive likelihood modeling or hierarchical Bayesian approaches for the model-based scheme—should be integrated to better capture those uncertainties.

Regarding the assumption of information availability, the proposed method requires extensive physical degradation knowledge, direct observation of damage state, and run-to-failure sensor data, which may not be satisfied in real-world scenarios. As an alternative, one may leverage numerical simulations to generate adequate training data. For instance, numerical cutter wear data may be used to build a wear estimation model, and numerical guided waves at varying crack lengths can inform damage quantification models. Moreover, while the Markov assumption can simplify prognostic modeling, it may not hold true in practical scenarios where degradation is influenced by operational and environmental factors. To better capture such non-Markov characteristics, future work will explore temporal models capable of describing extended historical dependencies, such as recurrent neural networks or attention-based models, allowing the model to incorporate memory effects and time-lagged patterns for describing non-Markov realistic degradation behavior.

CRediT authorship contribution statement

Tianzhi Li: Writing – review & editing, Writing – original draft, Visualization, Validation, Resources, Methodology, Investigation, Conceptualization. **Morteza Moradi:** Writing – review & editing, Visualization, Validation, Methodology, Investigation, Formal analysis. **Michel Gokan Khan:** Writing – review & editing, Visualization, Methodology, Investigation. **Renan Guarese:** Writing – review & editing, Visualization, Methodology, Conceptualization. **Jan Kronqvist:** Writing – review & editing, Resources, Methodology, Conceptualization. **Mario Romero:** Writing – review & editing, Visualization, Methodology, Conceptualization. **Ming Xiao:** Writing – review & editing, Resources, Investigation, Conceptualization. **Xi Vincent Wang:** Writing – review & editing, Visualization, Supervision, Resources.

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.

Acknowledgments

The authors would like to thank Prof. Shenfang Yuan and Prof. Jian Chen from Nanjing University of Aeronautics and Astronautics for providing the experimental data. The authors would like to thank Prof. Lihui Wang from KTH Royal Institute of Technology for his comments on the paper revision.

Appendix A. .1 particle filter

Within a Bayesian method, the estimation of the unknown state vector $\boldsymbol{z}_k$, can be derived from the available observations as follows:

$$p(\boldsymbol{z}_k|\boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{z}_k|\boldsymbol{z}_{k-1})p(\boldsymbol{z}_{k-1}|\boldsymbol{y}_{1:k-1})d\boldsymbol{z}_{k-1} \tag{A.1}$$

$$p(\boldsymbol{z}_k|\boldsymbol{y}_{1:k}) \propto p(\boldsymbol{y}_k|\boldsymbol{z}_k)p(\boldsymbol{z}_k|\boldsymbol{y}_{1:k-1}) \tag{A.2}$$

where $\boldsymbol{y}_{1:k}$ is the observations collected from time step 1 to k, the symbol $\propto$ signifies 'proportional to', $p(\boldsymbol{z}_k|\boldsymbol{z}_{k-1})$ and $p(\boldsymbol{y}_k|\boldsymbol{z}_k)$ represent the transition distribution and likelihood function, respectively, $p(\boldsymbol{z}_k|\boldsymbol{y}_{1:k-1})$ and $p(\boldsymbol{z}_k|\boldsymbol{y}_{1:k})$ depict the prior and posterior probability distribution function (PDF), respectively. This study employs the PF method as outlined in Table A.1.

Table A1
Particle filter algorithm.

Initialization: draw N particles $\{\boldsymbol{z}_0^i : i = 1, 2, \cdots, N_p\}$ from the distribution $p(\boldsymbol{z}_0)$.
For $k = 1, 2, \ldots,$
Prediction in PF: draw N particles $\{\boldsymbol{z}_k^i : i = 1, 2, \cdots, N_p\}$ by $p(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}^i)$.

(*continued on next page*)

Table A1 (*continued*)

Weight update: calculate the particle weight w_k^i through $w_k^i \propto p(\mathbf{y}_k|\mathbf{z}_k^i)$, and assign its normalized form $\widetilde{w}_k^i$ to each particle $\mathbf{z}_k^i$.
Calculate the effective sample size (ESS) by $N_{eff,k} = 1/\sum_{i=1}^{N_p}\left(\widetilde{w}_k^i\right)^2$.
If $N_{eff,k}/N < N_T$: **Resample** for those particles based on the normalized weights.
End

Appendix A. .2 algorithm hyperparameters

Tables A.2 and A.3 present the algorithm hyperparameters adopted for the aluminum fatigue crack and cutter wear datasets, respectively. They include PF hyperparameters for the model-based method [33], PF hyperparameters for the data-driven method, and fusion hyperparameters. These hyperparameters are taken from the works [28,60] and selected from a trial-and-error procedure. It is worth noting that the algorithm performance depends on a proper selection of these hyperparameters.

The selection of the number of particles involves finding a balance between accuracy and computational cost [61]. A larger number of particles generally brings more accurate estimates but comes with increased computational demands [62].

The initial ranges or distributions are required for PF initialization. The initial distributions for ln C and m are usually considered multivariate Gaussian and fitted by the crack propagation data [33,58]. Provided that the bias is close to zero at the initial step, all the initial bias samples are set as zero. The crack length samples are generated within a uniform distribution covering the initial observed length.

A random term, commonly referred to as process noise, is required in the process equation to depict the uncertainties in state evolution. At each time step, process noise samples are drawn from predefined distributions and assigned to the particles. In this study, the characteristics of the noise distributions—including their means and standard deviations (STDs)—are determined based on previous studies [28] and further refined through a trial-and-error procedure.

The computation of particle weights relies on the likelihood of each particle concerning the actual observation. This study assumes the measurement noise to be zero-mean Gaussian. The use of either a small or large STD can lead to inaccurate estimation, which is quite similar to the effect of process noise on estimation. In different applications, the STD can be either directly provided [63–65], empirically selected [66], or fitted by the data [11]. The STD used in this study is determined through a trial-and-error process, aligning with the strategy used in previous works [67,68].

The likelihood under nuisance parameters Eq. (14) is often solved by approximation methods, such as the Laplace approximation and sampling-based methods [69]. Within the scope of PF algorithms, the sampling method is a preferred choice. For the nuisance vector, certain representative samples can be obtained from the posterior distribution. Then, these samples yield likelihood values for each particle, and the maximum of these likelihoods is taken as the profile likelihood for one particle. This study employs the subsampling strategy for calculating the profile likelihood, where a subset of 200 samples is randomly selected from the samples representing the distribution of nuisance parameters.

Table A2
Algorithm hyperparameters for the aluminum fatigue crack dataset.

Model-based method	
Number of particles	40,000
Initial distribution for crack length	U (2.9, 3.1)
Initial distribution for polynomial coefficient	U ($0.8c_f$, $1.2c_f$)
Initial distribution for bias	U (−0.01, 0.01)
STD of process noise for crack length	0.05
STD of process noise for polynomial coefficient	$0.00015c_f$
STD of process noise for bias	0.001
STD for each measurement	20 % of RMS value of each training feature
Data-driven method	
Number of particles	40,000
Initial distribution for EOL	U (20000, 100000)
Initial distribution for bias	U (−0.01, 0.01)
STD of process noise for EOL	320
STD of process noise for HI	0.00015
STD of process noise for bias	0.001
STD for each measurement	20 % of RMS value of each training feature

(*continued on next page*)

Table A2 (*continued*)

Model-based method	
Fusion	
'Accurate' EOL distribution	99 % CB of EOL samples from data-driven method
Fusion parameter α	0.01
Fusion parameter β	0.01
Fusion parameter γ	0.01

Note: 'STD' 'CB', 'RMS', and 'c_f' denote 'standard deviation', 'confidence boundary', 'root-mean-square', and 'polynomial coefficient fitted from the crack propagation data [33]', respectively.

Table A3
Algorithm hyperparameters for the cutter wear dataset.

Model-based method	
Number of particles	20,000
Initial distribution for polynomial parameters	$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$
Initial distribution for bias	U (−0.01, 0.01)
STD of process noise for wear level	0.1
STD of process noise for polynomial parameters	0.005 $\boldsymbol{\mu}$
STD of process noise for bias	0.002
STD for each measurement	10 % of RMS value of each training feature
Data-driven method	
Number of particles	20,000
Initial distribution for EOL	U (0, 500)
Initial distribution for bias	U (−0.01, 0.01)
STD of process noise for EOL	2
STD of process noise for HI	0.0001
STD of process noise for bias	0.002
STD for each measurement	10 % of RMS value of each training feature
Fusion	
'Accurate' EOL distribution	99 % CB of EOL samples from data-driven method
Fusion parameter α	0.1
Fusion parameter β	0.1
Fusion parameter γ	0.01

Note: '$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$' denotes multivariate Gaussian distribution of the polynomial coefficients fitted from tool wear data of training cutters.

Data availability

Data will be made available on request.

References

[1] H. Li, Z. Zhang, T. Li, X. Si, A review on physics-informed data-driven remaining useful life prediction: challenges and opportunities, Mech. Syst. Sig. Process. 209 (2024) 111120.
[2] R. Gao, L. Wang, R. Teti, D. Dornfeld, S. Kumara, M. Mori, M. Helu, Cloud-enabled prognosis for manufacturing, CIRP Ann. 64 (2015) 749–772.
[3] B. Wang, Y. Lei, N. Li, N. Li, A Hybrid Prognostics Approach for estimating remaining Useful Life of Rolling Element Bearings, IEEE Trans. Reliab. 69 (2020) 401–412.
[4] J. Chen, S. Yuan, C. Sbarufatti, X. Jin, Dual crack growth prognosis by using a mixture proposal particle filter and on-line crack monitoring, Reliab. Eng. Syst. Saf. 215 (2021) 107758.
[5] F. Cadini, C. Sbarufatti, M. Corbetta, F. Cancelliere, M. Giglio, Particle filtering-based adaptive training of neural networks for real-time structural damage diagnosis and prognosis, Struct. Control Health Monit. 26 (2019).
[6] J. Chen, S. Yuan, H. Wang, On-line updating Gaussian process measurement model for crack prognosis using the particle filter, Mech. Syst. Sig. Process. 140 (2020) 106646.
[7] X. Zhou, S. He, L. Dong, S.N. Atluri, Real-Time Prediction of Probabilistic Crack Growth with a Helicopter Component Digital Twin, AIAA J. 60 (2022) 2555–2567.
[8] P. Kundu, A.K. Darpe, M.S. Kulkarni, A review on diagnostic and prognostic approaches for gears, Struct. Health Monit. 20 (2020) 2853–2893.
[9] X. Cai, N. Li, M. Xie, RUL prediction for two-phase degrading systems considering physical damage observations, Reliab. Eng. Syst. Saf. 244 (2024) 109926.
[10] M. Chiachío, J. Chiachío, S. Sankararaman, K. Goebel, J. Andrews, A new algorithm for prognostics using Subset simulation, Reliab. Eng. Syst. Saf. 168 (2017) 189–199.
[11] J. Chen, S. Yuan, X. Jin, On-line prognosis of fatigue cracking via a regularized particle filter and guided wave monitoring, Mech. Syst. Sig. Process. 131 (2019) 1–17.

[12] T. Li, C. Sbarufatti, F. Cadini, Multiple local particle filter for high-dimensional system identification, Mech. Syst. Sig. Process. 209 (2024) 111060.
[13] T. Yang, G. Li, K. Li, X. Li, Q. Han, The LPST-Net: a new deep interval health monitoring and prediction framework for bearing-rotor systems under complex operating conditions, Adv. Eng. Inf. 62 (2024) 102558.
[14] X. Liu, Y. Lei, N. Li, X. Si, X. Li, RUL prediction of machinery using convolutional-vector fusion network through multi-feature dynamic weighting, Mech. Syst. Sig. Process. 185 (2023) 109788.
[15] G. Galanopoulos, N. Eleftheroglou, D. Milanoski, A. Broer, D. Zarouchas, T. Loutas, A novel strain-based health indicator for the remaining useful life estimation of degrading composite structures, Compos. Struct. 306 (2023) 116579.
[16] M. Moradi, F.C. Gul, D. Zarouchas, A novel machine learning model to design historical-independent health indicators for composite structures, Compos. B Eng. 275 (2024) 111328.
[17] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T.H. Loutas, D. Zarouchas, Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and SHM data, Eng. Appl. Artif. Intel. 117 (2023) 105502.
[18] Q. Liu, J. Liu, X. Liu, C. Yue, J. Ma, B. Zhang, S.Y. Liang, L. Wang, A method for remaining useful life prediction of milling cutter using multi-scale spatial data feature visualization and domain separation prediction network, Mech. Syst. Sig. Process. 225 (2025) 112251.
[19] R. Zhong, B. Hu, Y. Feng, S. Lou, Z. Hong, F. Wang, G. Li, J. Tan, Lithium-ion battery remaining useful life prediction: a federated learning-based approach, Energy Ecol. Environ. 9 (2024) 549–562.
[20] X. Li, Q. Ding, J.-Q. Sun, Remaining useful life estimation in prognostics using deep convolution neural networks, Reliab. Eng. Syst. Saf. 172 (2018) 1–11.
[21] J. Yuan, Y. Lei, B. Yang, N. Li, X. Li, Z. Chen, W. Han, A hybrid maintenance strategy for equipment with competitive failure modes: Sudden failure and multi-stage degradation failure, Mech. Syst. Sig. Process. 234 (2025) 112846.
[22] M. Arias Chao, C. Kulkarni, K. Goebel, O. Fink, Fusing physics-based and deep learning models for prognostics, Reliab. Eng. Syst. Saf. 217 (2022) 107961.
[23] C. Yin, Y. Li, Y. Wang, Y. Dong, Physics-guided degradation trajectory modeling for remaining useful life prediction of rolling bearings, Mech. Syst. Sig. Process. 224 (2025) 112192.
[24] H. Cao, W. Xiao, J. Sun, M.-G. Gan, G. Wang, A hybrid data- and model-driven learning framework for remaining useful life prognostics, Eng. Appl. Artif. Intel. 135 (2024) 108557.
[25] Y. Zhu, J. Cheng, Z. Liu, X. Zou, Z. Wang, Q. Cheng, H. Xu, Y. Wang, F. Tao, Remaining Useful Life Prediction Approach based on Data Model Fusion: an Application in Rolling Bearings, IEEE Sens. J. 24 (2024) 42230–42244.
[26] Y. Wang, M. Li, L. Zheng, M. Shi, Z. Zheng, X. Pei, Phyformer: a degradation physics-informed self-data driven approach to machinery prognostics, Adv. Eng. Inf. 62 (2024) 102772.
[27] Y. Zang, W. Shangguan, B. Cai, H. Wang, M.G. Pecht, Hybrid remaining useful life prediction method, A Case Study on Railway D-Cables, Reliability Engineering & System Safety 213 (2021) 107746.
[28] T. Li, Particle filter-based fatigue damage prognosis using prognostic-aided model updating, Mech. Syst. Sig. Process. 211 (2024) 111244.
[29] K. Xue, J. Yang, M. Yang, D. Wang, An improved Generic Hybrid Prognostic Method for RUL Prediction based on PF-LSTM Learning, IEEE Trans. Instrum. Meas. 72 (2023) 1–21.
[30] M. Ahwiadi, W. Wang, An enhanced particle filter technology for battery system state estimation and RUL prediction, Measurement 191 (2022) 110817.
[31] Z. Li, D. Wu, C. Hu, J. Terpenny, An ensemble learning-based prognostic approach with degradation-dependent weights for remaining useful life prediction, Reliab. Eng. Syst. Saf. 184 (2019) 110–122.
[32] W. Hou, Y. Peng, Enhancing bearing life prediction: Sparse Gaussian process regression approach based on sequential ensemble and residual reduction for degradation prediction, Reliab. Eng. Syst. Saf. 256 (2025) 110788.
[33] T. Li, J. Chen, S. Yuan, D. Zarouchas, C. Sbarufatti, F. Cadini, Particle filter-based fatigue damage prognosis by fusing multiple degradation models, Struct. Health Monit. 14759217231216697 (2024).
[34] F. Zeng, Y. Li, Y. Jiang, G. Song, An online transfer learning-based remaining useful life prediction method of ball bearings, Measurement 176 (2021) 109201.
[35] J. Zhuang, Y. Cao, M. Jia, X. Zhao, Q. Peng, Remaining useful life prediction of bearings using multi-source adversarial online regression under online unknown conditions, Expert Syst. Appl. 227 (2023) 120276.
[36] W. Mao, K. Liu, Y. Zhang, X. Liang, Z. Wang, Self-Supervised Deep Tensor Domain-Adversarial Regression Adaptation for Online remaining Useful Life Prediction across Machines, IEEE Trans. Instrum. Meas. 72 (2023) 1–16.
[37] C. Liu, L. Zhang, Y. Zheng, Z. Jiang, J. Zheng, C. Wu, Online industrial fault prognosis in dynamic environments via task-free continual learning, Neurocomputing 598 (2024) 127930.
[38] T. Li, M. Moradi, M. Xiao, L. Wang, Online Inverse Solution for Deep Learning-based Prognostics, Materials Research Proceedings, 50 (2025) 119-126.
[39] X. Li, B. Lim, J. Zhou, S. Huang, S. Phua, K. Shaw, M. Er, Fuzzy neural network modelling for tool wear estimation in dry milling operation, Annual conference of the PHM society, 2009.
[40] M. Jouin, R. Gouriveau, D. Hissel, M.-C. Péra, N. Zerhouni, Particle filter-based prognostics: Review, discussion and perspectives, Mech. Syst. Sig. Process. 72–73 (2016) 2–31.
[41] D. Cristiani, C. Sbarufatti, F. Cadini, M. Giglio, Fatigue damage diagnosis and prognosis of an aeronautical structure based on surrogate modelling and particle filter, Struct. Health Monit. 20 (2020) 2726–2746.
[42] A. Doucet, S. Godsill, C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering, Statistics & Computing 10 (2000) 197–208.
[43] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2002) 174–188.
[44] Y. Wang, Y. Lei, N. Li, T. Yan, X. Si, Deep multisource parallel bilinear-fusion network for remaining useful life prediction of machinery, Reliab. Eng. Syst. Saf. 231 (2023) 109006.
[45] S. Garmaev, O. Fink, Deep Koopman Operator-based degradation modelling, Reliab. Eng. Syst. Saf. 251 (2024) 110351.
[46] L. Guo, Y. Lei, N. Li, T. Yan, N. Li, Machinery health indicator construction based on convolutional neural networks considering trend burr, Neurocomputing 292 (2018) 142–150.
[47] S. Yin, X. Zhu, Intelligent Particle Filter and its Application to Fault Detection of Nonlinear System, IEEE Trans. Ind. Electron. 62 (2015) 3852–3861.
[48] A. Aspeel, A. Gouverneur, R.M. Jungers, B. Macq, Optimal Intermittent Particle Filter, IEEE Trans. Signal Process. 70 (2022) 2814–2825.
[49] M.J. Simpson, S.A. Walker, E.N. Studerus, S.W. McCue, R.J. Murphy, O.J. Maclaren, Profile likelihood-based parameter and predictive interval analysis guides model choice for ecological population dynamics, Math. Biosci. 355 (2023) 108950.
[50] L. Qiu, S. Yuan, X. Shi, T. Huang, Design of piezoelectric transducer layer with electromagnetic shielding and high connection reliability, Smart Mater. Struct. 21 (2012) 075032.
[51] R. Zhong, Y. Feng, P. Li, X. Wu, A. Guo, A. Zhang, C. Li, Uncertainty-aware nuclear power turbine vibration fault diagnosis method integrating machine learning and heuristic algorithm, IET Collab. Intell. Manuf. 6 (2024) e12108.
[52] J. Tian, D. Han, H.R. Karimi, Y. Zhang, P. Shi, A universal multi-source domain adaptation method with unsupervised clustering for mechanical fault diagnosis under incomplete data, Neural Netw. 106167 (2024).
[53] J. Wu, Y. Yang, Z. Lin, Y. Lin, Y. Wang, W. Zhang, H. Ma, Weak ultrasonic guided wave signal recognition based on one-dimensional convolutional neural network denoising autoencoder and its application to small defect detection in pipelines, Measurement 242 (2025) 116234.
[54] P. Li, Y. Pei, J. Li, A comprehensive survey on design and application of autoencoder in deep learning, Appl. Soft Comput. 138 (2023) 110176.
[55] P. Dixit, S. Silakari, Deep Learning Algorithms for Cybersecurity applications: a Technological and Status Review, Computer Science Review 39 (2021) 100317.
[56] T. Li, C. Sbarufatti, F. Cadini, J. Chen, S. Yuan, Particle filter-based hybrid damage prognosis considering measurement bias, Struct. Control Health Monit. 29 (2022) e2914.
[57] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, M. Schwabacher, Metrics for evaluating performance of prognostic techniques, International Conference on Prognostics and Health Management 2008 (2008) 1–17.

[58] M. Corbetta, C. Sbarufatti, M. Giglio, M.D. Todd, Optimization of nonlinear, non-Gaussian Bayesian filtering for diagnosis and prognosis of monotonic degradation processes, Mech. Syst. Sig. Process. 104 (2018) 305–322.
[59] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, A.K. Nandi, Applications of machine learning to machine fault diagnosis: a review and roadmap, Mech. Syst. Sig. Process. 138 (2020) 106587.
[60] T. Li, J. Chen, S. Yuan, F. Cadini, C. Sbarufatti, Particle filter-based damage prognosis using online feature fusion and selection, Mech. Syst. Sig. Process. 203 (2023) 110713.
[61] X. Li, X. Yang, Y. Yang, I. Bennett, D. Mba, An intelligent diagnostic and prognostic framework for large-scale rotating machinery in the presence of scarce failure data, Struct. Health Monit. 1475921719884019 (2019).
[62] E.N. Chatzi, A.W. Smyth, The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing, Struct. Control Hlth. 16 (2010) 99–123.
[63] J. Chiachío, M.L. Jalón, M. Chiachío, A. Kolios, A Markov chains prognostics framework for complex degradation processes, Reliab. Eng. Syst. Saf. 195 (2020) 106621.
[64] F. Cadini, C. Sbarufatti, M. Corbetta, M. Giglio, A particle filter-based model selection algorithm for fatigue damage identification on aeronautical structures, Struct. Control Hlth. (2017) e2002.
[65] F. Cadini, E. Zio, G. Peloni, Particle Filtering for the Detection of Fault Onset Time in Hybrid Dynamic Systems with Autonomous Transitions, IEEE Trans. Reliab. 61 (2012) 130–139.
[66] M. Corbetta, C. Sbarufatti, M. Giglio, A. Saxena, K. Goebel, A Bayesian framework for fatigue life prediction of composite laminates under co-existing matrix cracks and delamination, Compos. Struct. 187 (2018) 58–70.
[67] T. Li, L. Lomazzi, F. Cadini, C. Sbarufatti, J. Chen, S. Yuan, Numerical simulation-aided particle filter-based damage prognosis using Lamb waves, Mech. Syst. Sig. Process. 178 (2022) 109326.
[68] T. Li, C. Sbarufatti, F. Cadini, J. Chen, S. Yuan, Particle filter-based hybrid damage prognosis considering measurement bias, Struct. Control Health Monit. 29 (2021) e2914.
[69] R. Xu, F. Vaida, D.P. Harrington, Using profile likelihood for semiparametric model selection with application to proportional hazards mixed models, Stat. Sin. 19 (2009) 819.